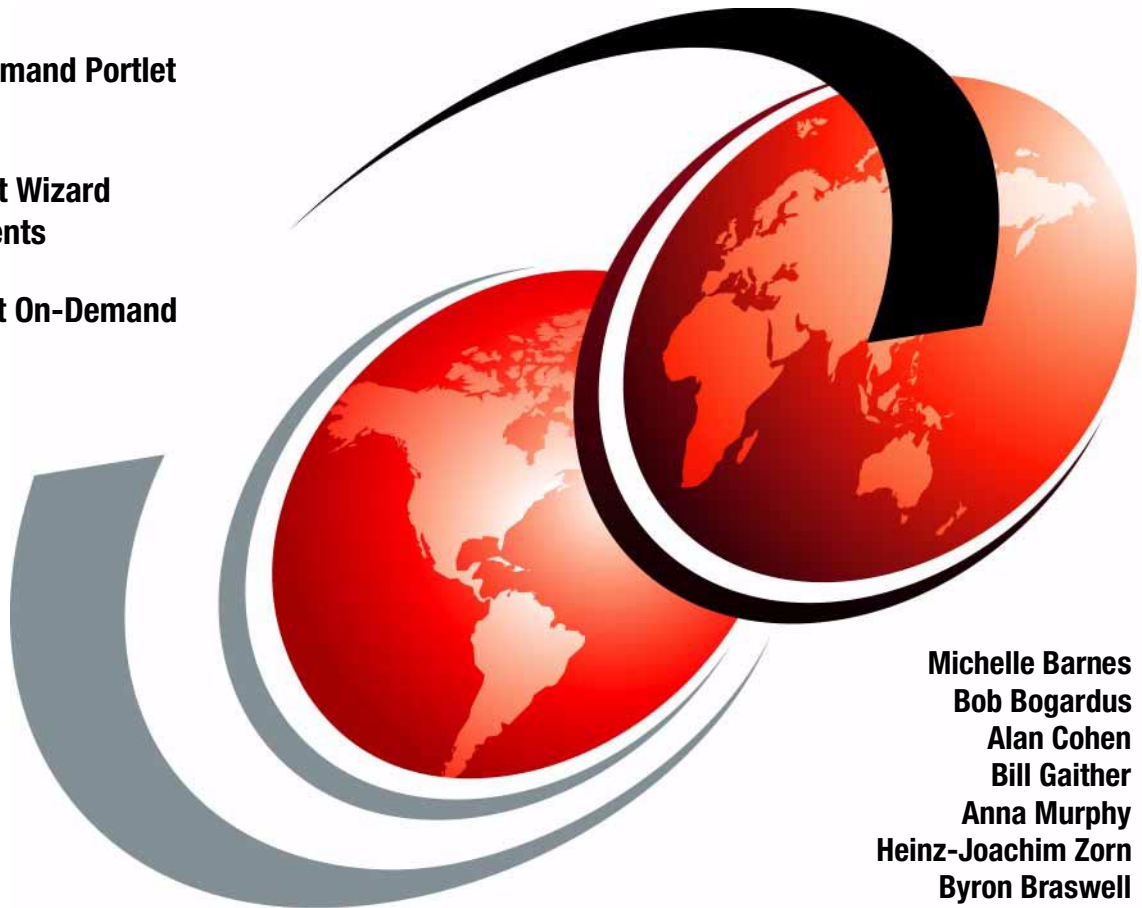


IBM Host Access Client Package Update

Host On-Demand Portlet

Deployment Wizard
enhancements

Enable Host On-Demand
for Java 2



Michelle Barnes
Bob Bogardus
Alan Cohen
Bill Gaither
Anna Murphy
Heinz-Joachim Zorn
Byron Braswell



International Technical Support Organization

IBM Host Access Client Package Update

December 2002

Take Note! Before using this information and the product it supports, be sure to read the general information in “Special notices” on page 1045.

Second Edition (December 2002)

This edition applies to Version 3 of the Host Access Client Package, which includes IBM WebSphere Host On-Demand Version 7, Screen Customizer Version 2.0.70, Personal Communications for Windows Version 5.6.

Comments may be addressed to:
IBM Corporation, International Technical Support Organization
Dept. HZ8 Building 662
P.O. Box 12195
Research Triangle Park, NC 27709-2195

When you send information to IBM, you grant IBM a non-exclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

© Copyright International Business Machines Corporation 2002. All rights reserved.

Note to U.S Government Users – Documentation related to restricted rights – Use, duplication or disclosure is subject to restrictions set forth in GSA ADP Schedule Contract with IBM Corp.

Contents

Preface	xix
The team that wrote this redbook	xix
Special notice	xxi
IBM trademarks	xxii
Comments welcome	xxii
 Chapter 1. Introduction to Host Access Client Package	1
1.1 Host On-Demand	3
1.1.1 Host On-Demand features	4
1.1.2 Host On-Demand components	8
1.1.3 Architecture and operations	12
1.2 Introduction to Personal Communications Version 5.6	15
1.3 Introduction to Screen Customizer	16
1.4 National Language Support	16
1.4.1 The need for National Language Support	17
1.4.2 Globalization de-mystified	18
1.4.3 Installation	19
 Part 1. IBM WebSphere Host On-Demand	21
 Chapter 2. Planning and installation	23
2.1 Supported platforms	24
2.2 Server requirements	24
2.2.1 zSeries platform	24
2.2.2 iSeries platform	24
2.2.3 Windows Platforms	25
2.2.4 AIX Platform	25
2.2.5 Solaris platform	26
2.2.6 HP-UX platform	26
2.2.7 Linux and other Unix platform	27
2.2.8 OS/2 platform	28
2.2.9 Novell Netware platform	28
2.2.10 Supported LDAP servers	29
2.2.11 Web servers	29
2.3 Client requirements	30
2.3.1 Supported operating systems	30
2.3.2 Supported browsers	31
2.4 Installing Host On-Demand	32
2.4.1 Installing on Windows NT, Windows 2000 or Windows XP Servers ..	32

2.4.2	Installing on OS/2	43
2.4.3	Installing on Novell NetWare	45
2.4.4	Installing on AIX	47
2.4.5	Installing on UNIX (Solaris, HP-UX, and Linux)	50
2.4.6	Installing on iSeries	52
2.4.7	Installing on OS/390 or z/OS	56
2.4.8	Changing the Configuration Server port	56
2.4.9	Installing the Configuration Servlet	60
2.4.10	Installing the locally installed client	62
2.4.11	Installing the Deployment Wizard	64
2.5	Migration considerations	66
2.5.1	Server considerations	67
2.5.2	Client considerations	73
2.5.3	Client Migration Problems	75
2.6	Removing Host On-Demand	75
2.6.1	zSeries	75
2.6.2	iSeries	75
2.6.3	Windows NT, Windows 2000, or Windows XP	76
2.6.4	AIX, Solaris, Linux, HP-UX	76
2.6.5	OS/2	76
2.6.6	Novell NetWare	76
2.7	Service updates	77
Chapter 3.	z/OS implementation	79
3.1	Planning	80
3.1.1	Software requirements	80
3.1.2	DASD storage requirements	81
3.1.3	Backing up the private directory	82
3.1.4	Upgrade considerations	82
3.2	Host On-Demand installation	83
3.2.1	Installation jobs	83
3.2.2	Installation instructions	84
3.3	Activating Host On-Demand Service Manager	85
3.3.1	UNIX System Services environment	86
3.3.2	Security Server (RACF) considerations	86
3.3.3	Web server environment	87
3.3.4	Modify the HOMSERVER sample job	89
3.3.5	Start the Host On-Demand Service Manager	89
3.3.6	Changing the configuration port	90
3.3.7	Stopping the Service Manager	92
3.3.8	Considerations when running multiple TCP/IP stacks	94
3.3.9	Miscellaneous information	94
3.4	Deployment Wizard considerations	96

3.4.1 Deployment Wizard files	96
3.5 Configuration Servlet setup	99
3.5.1 Configuration files	99
3.5.2 Testing the servlet.	102
3.5.3 Changing the Configuration Server port	103
3.5.4 Problem determination	104
3.6 Using SSL with Communications Server for z/OS.	105
3.6.1 Telnet Server and SSL support.	106
3.6.2 SSL encryption overview.	107
3.6.3 SSL Configuration using gskkyman	109
3.6.4 Certificate management using RACF	126
3.7 Express Logon Feature (ELF)	130
3.8 LDAP directory server	137
3.8.1 Schema installation.	138
3.8.2 Directory Tree	139
3.8.3 Performance considerations	139
3.9 Native Authentication	140
3.9.1 Installation of Native Authentication service	140
3.9.2 Starting Native Authentication service	141
3.9.3 Testing Native Authentication service.	143
Chapter 4. iSeries tips.	147
4.1 Upgrade JVM level to 1.3	148
4.2 Using IBM HTTP Server (Powered by Apache).	149
4.3 Using Lotus Domino HTTP Server	152
4.3.1 Restarting the Domino HTTP Server	154
4.3.2 Using the Domino HTTP Server and Host On-Demand	155
4.4 Using the Configuration Servlet	155
4.5 Screen Customizer and 5250 subfiles	156
4.6 Add Printer Definition Table entry.	157
4.7 Performance tips	158
4.7.1 Web page caching	158
4.7.2 Compile Host On-Demand for faster execution	159
4.8 iSeries as a target host.	160
4.8.1 5250 Workstation ID	160
4.8.2 5250 Telnet dropout	161
4.8.3 Tip for 5250 printing	161
4.8.4 Mapping a network drive to the iSeries.	162
4.8.5 Additional iSeries-related Web pages.	162
Chapter 5. Clients	163
5.1 Host On-Demand default clients.	164
5.1.1 Administration clients	166

5.1.2	Download clients	167
5.1.3	Cached clients	167
5.1.4	Emulator clients	171
5.1.5	Problem determination clients	171
5.1.6	Function On-Demand client	172
5.2	Componentization	172
5.3	Smart caching	174
5.4	Utility clients	174
5.4.1	New user client	175
5.4.2	Remove cached client	175
5.5	CICS Gateway client	176
5.6	Database On-Demand	177
5.7	The emulator session window	177
5.7.1	Operator information area	178
5.7.2	Customizing the toolbar	179
5.7.3	Color remapping	183
5.7.4	Keyboard remapping	185
5.8	Improvements to Java 2 support	188
5.8.1	Terms defined	188
5.8.2	Java 2 support before Host On-Demand 7.0	188
5.8.3	Java 2 support with Host On-Demand 7.0	189
5.8.4	Features that take advantage of Java 2	190
5.8.5	Look and feel with Java 2 version of Host On-Demand	190
5.9	Java 2 practical issues	193
5.9.1	Advantages of switching clients to Java 2 enabled browsers	194
5.9.2	Limitations and workarounds	194
5.9.3	Effects on system resources	195
5.9.4	Must I migrate my existing HOD 6.0 Deployment Wizard files?	196
5.9.5	What if I want to continue running Java 1 browsers only?	197
5.9.6	What if I am already running Java 2 enabled browsers?	197
5.9.7	What if I want to migrate my users to Java 2 enabled browsers?	198
5.10	Client Java type: Java 1, Java 2, or Auto Detect	199
5.10.1	Overview	199
5.10.2	Java 1	200
5.10.3	Java 2	201
5.10.4	Auto detect	201
5.11	Effect of client Java type at startup	202
5.11.1	Messages	202
5.11.2	Startup behavior for Java 1 download client	203
5.11.3	Startup behavior for Java 2 download client	204
5.11.4	Startup behavior for Java 1 cached client	205
5.11.5	Startup behavior for Java 2 cached client	206
5.12	Download client and cached client implementation	207

5.12.1	HostOnDemand applet and CachedAppletSupport applet	207
5.12.2	How Host On-Demand component modules are stored	208
5.13	The Java 2 cached client	210
5.13.1	Java 2 cache options	211
5.13.2	Downloading a Java 2 component not on the preload list.	212
5.13.3	Java 2 cached client does not interfere with download client	212
5.13.4	Java 2 cached client upgrades	213
5.13.5	Handling cached client components for Java 1 and Java 2	214
5.13.6	Increased flexibility with Java 2 cached clients	216
5.13.7	Removing the cached client	217
5.14	The Java 2 download client	220
5.15	Web browsers: Java 1 and Java 2 enabled	221
5.15.1	Web browsers supported	222
5.15.2	Netscape web browsers	222
5.15.3	Microsoft web browsers: Internet Explorer	223
5.16	The Java 2 plug-in	227
5.16.1	Java 2 plug-ins supported	227
5.16.2	Clients can download Java 2 runtime for Win32 platform	228
5.17	Additional information	234
5.17.1	More information on the Java 2 sticky cache	236
5.17.2	More information on the cached client	238
5.17.3	More information on the download client	239
5.17.4	More information on launching the Host On-Demand applets	241
Chapter 6.	Database On-Demand	247
6.1	Administering Database On-Demand	248
6.1.1	Creating Database On-Demand groups and users.	248
6.1.2	Configuring database options	248
6.1.3	Administering SQL statements	250
6.2	Using Database On-Demand	253
6.2.1	Creating a new SQL statement.	254
6.2.2	Running an SQL statement.	265
6.2.3	Changing an SQL statement.	265
6.2.4	Deleting an SQL statement.	266
6.2.5	Customizing user options	266
6.3	Installing and registering other JDBC drivers	266
6.3.1	Installing a driver	267
6.3.2	Registering a driver.	267
6.3.3	Using a new driver	268
6.3.4	Common access problems	269
Chapter 7.	Administration	273
7.1	Manage users and groups	274

7.1.1 Planning	276
7.1.2 Manage groups	276
7.1.3 Create a new user	277
7.1.4 Using Native Authentication	281
7.1.5 Administering groups, sessions and users	283
7.1.6 Filtering	284
7.1.7 Configuring sessions	285
7.1.8 Disabling functions	345
7.2 Services	348
7.3 Redirector service	350
7.3.1 Configuring the Redirector	351
7.3.2 Configuring emulator sessions to use the Redirector	354
7.4 Directory Service	357
7.4.1 Use Directory Service (LDAP)	359
7.4.2 Migrate configuration to LDAP directory	359
7.5 OS/400 Proxy Server	361
7.6 License Use Management	362
7.6.1 Enabling License-Use Count	364
7.6.2 Disabling License-Use Count	365
7.7 Directory Utility	367
7.7.1 Using Directory Utility	368
7.7.2 XML file syntax	369
7.7.3 Example	374
7.8 Java 2 considerations for iSeries	379
Chapter 8. LDAP directory server	381
8.1 LDAP overview	382
8.2 Host On-Demand and LDAP overview	383
8.3 Supported LDAP directory servers	383
8.4 Schema installation	384
8.4.1 Netscape Directory Server	384
8.4.2 IBM SecureWay LDAP Directory Server	384
8.5 Host On-Demand directory operations	386
8.5.1 Switching to an LDAP directory server	386
8.5.2 Unable to enable LDAP	389
8.5.3 LDAP migration implications	390
8.6 Operational issues	393
8.6.1 Startup sequence	393
8.6.2 Reverting to the private data store if a directory server fails	394
8.6.3 Debug Tracing of the Service Manager	394
8.6.4 LDAP logs	395
Chapter 9. Configuration Servlet	397

9.1	Installation	398
9.1.1	Manual installation	398
9.2	Configuring WebSphere Application Server 4.0	399
9.3	Configuring WebSphere Application Server 3.5	401
9.3.1	IBM WebSphere graphical configuration.	402
9.4	Enabling clients	411
9.5	Referencing the Configuration Servlet	412
9.5.1	Direct reference.	412
9.5.2	Indirect reference	412
9.6	XMLConfig utility	413
9.6.1	Add Configuration Servlet to default_app	413
9.6.2	Add Configuration Servlet to new application	415
9.7	Implementation scenarios.	417
9.7.1	Load balancing	417
9.7.2	Native Authentication	418
9.8	Problem determination	418
Chapter 10.	OS/400 Proxy	419
10.1	How to configure a simple session	420
10.2	Using the OS/400 Proxy	423
10.3	Enabling SSL	423
10.3.1	Prerequisites	423
10.3.2	Configure each target iSeries	424
10.3.3	Configuring the OS/400 Proxy keyring	425
10.4	Firewall rules for OS/400 Proxy	426
Chapter 11.	Security	427
11.1	Signed applet support	429
11.2	Host On-Demand SSL support	430
11.2.1	Java class files	430
11.2.2	Microsoft cryptographic service provider database.	431
11.3	Host On-Demand SSL implementations	440
11.3.1	Basic SSL	441
11.3.2	Server authentication	441
11.3.3	Client authentication	443
11.3.4	FTP client	444
11.3.5	TN3270 client	444
11.3.6	TN5250 client	444
11.3.7	VT client	445
11.3.8	AS/400 Database On-Demand client	446
11.4	Defining a secure Telnet session	447
11.4.1	Enable Security (SSL).	448
11.4.2	Telnet-negotiated session	448

11.4.3	Server authentication	448
11.4.4	Add MSIE browser's keyring	448
11.4.5	Client authentication	449
11.5	The Host On-Demand Redirector	452
11.5.1	Redirector certificates	453
11.5.2	Configuring the Host On-Demand Redirector	454
11.5.3	Certificate management	454
11.6	The OS/400 Proxy server	455
11.7	Configuration Servlet	455
11.8	Express Logon Feature	455
11.8.1	Host On-Demand session setup	456
11.8.2	Record the macro	457
11.8.3	ELF Design	461
11.9	LDAP directory considerations	465
11.10	Using Host On-Demand with a firewall	466
11.10.1	TCP/IP ports used by Host On-Demand	467
11.11	Native Authentication	468
11.11.1	Native platform authentication requirements	470
11.11.2	Installation and activation	471
11.11.3	Debug information	475
11.12	Integrated Windows domain logon	475
11.12.1	Activating Integrated Windows domain logon	476
11.12.2	Process flow	480
11.12.3	Configuration parameters	481
11.12.4	Trouble shooting	482
11.13	Telnet-negotiated security	483
11.13.1	Session configuration	483
11.13.2	Session negotiation	485
Chapter 12.	Certificate management	487
12.1	Files managed by Certificate Management	488
12.2	Using certificates	488
12.2.1	Using a site certificate from a well-known CA	489
12.2.2	Using a certificate from an unknown CA	489
12.2.3	Using a self-signed certificate	490
12.2.4	Making server certificates available to clients	490
12.3	Certificate management utility	492
12.3.1	Starting the certificate management utility	492
12.3.2	Create a request for an unknown CA	493
12.3.3	Receive the CA's certificate	497
12.3.4	Receive the certificate signed by the CA	498
12.3.5	Create a self-signed certificate	500
12.3.6	Make a certificate available for the clients	503

12.4 The Certificate Wizard	506
12.4.1 Using the Certificate Wizard	506
12.5 Keyring utility	509
Chapter 13. Deployment strategies	513
13.1 Host On-Demand configuration models	514
13.1.1 HTML-based model.	515
13.1.2 Configuration server-based model	516
13.1.3 Combined model.	518
13.1.4 User Preferences	520
13.2 Host On-Demand emulator clients	521
13.3 Security requirements	522
13.3.1 Firewall considerations	524
13.4 Host On-Demand Server Platform choices	525
13.4.1 User locations	527
13.5 Other considerations	528
Chapter 14. Deployment Wizard.	529
14.1 Planning	530
14.2 Starting the Deployment Wizard	530
14.3 Using the Deployment Wizard	532
14.3.1 HTML-based model example	534
14.3.2 Configuration Server-based model example.	549
14.3.3 Combined model example	555
14.3.4 User preferences stored on local machines	557
14.4 Distributing Deployment Wizard files	560
14.5 Files created by the Deployment Wizard	562
14.5.1 Files stored in publish directory.	562
14.5.2 Files stored in customized subdirectory	562
14.5.3 Files stored on local machine	563
14.6 Additional HTML parameters	564
Chapter 15. Custom HTML templates	567
15.1 Purpose of this feature	568
15.1.1 Note: the Host On-Demand logon page is also affected.	569
15.1.2 Supported configuration models	569
15.1.3 Client Java Types	570
15.2 Name of the main HTML output file	570
15.3 Specifying a custom HTML template file	570
15.3.1 File management	571
15.3.2 Do not modify Wizard.html	572
15.3.3 Custom HTML template vs. main Deployment Wizard output file.	572
15.3.4 When error checking is performed	572
15.4 UTF-8 encoding	573

15.4.1 Using an ASCII editor instead of a UTF-8 editor	573
15.5 Parts of the custom HTML template file	574
15.5.1 Default custom HTML template file	574
15.5.2 Text markers	575
15.5.3 DOCTYPE declaration	575
15.5.4 HEAD Element	576
15.5.5 BODY element	577
15.5.6 Within BODY before <!-- STARTAPPLETPARMS -->	577
15.5.7 Within BODY between <!-- STARTAPPLETPARMS --> and <!-- ENDAPPLETPARMS -->	579
15.5.8 Within BODY, between <!-- ENDAPPLETPARMS --> and <!-- APPLET -->	580
15.5.9 Within BODY, after <!-- APPLET -->	581

Chapter 16. Modifying session properties dynamically (HTML overrides) . 583

16.1 The Benefits of modifying session properties dynamically	584
16.2 The Need to dynamically modify session properties	584
16.3 List of session properties that can be overridden	585
16.4 Scenario 1: Overriding the LU name based on the client's IP address . 589	
16.4.1 Steps to modify the session properties	589
16.4.2 Troubleshooting Scenario 1	603
16.4.3 Completed HTML file after edits	604
16.5 Scenario 2: Specifying the host name using an HTML form	608
16.5.1 Steps to modify the session properties	608
16.5.2 Troubleshooting Scenario 2	613
16.5.3 Completed HTML file after edits	614

Chapter 17. Host On-Demand Portlets 617

17.1 Introduction to Portal Servers	618
17.1.1 What is a portlet	619
17.1.2 Information on IBM Portal Servers	619
17.2 How HOD portlet support works with Websphere Portal Server 4.1	620
17.3 Scenarios for When and where to use HOD portlet.	621
17.4 Installing HOD portlet in Websphere Portal Server 4.1	623
17.4.1 Deploying HOD portlet to Portal page.	629
17.4.2 Special Considerations when using HOD portlet	633
17.4.3 Migrating to HOD V7 portlet from previous HOD Versions	637
17.5 Potential Portlet problems	638
17.5.1 Location of Portal log Files	639

Chapter 18. Session Manager APIs 641

18.1 The four types of JavaScript-based APIs	642
--	-----

18.1.1 Session Manager APIs	642
18.1.2 Presentation Space APIs	642
18.1.3 Host On-Demand Function APIs	643
18.1.4 Error Reporting APIs	643
18.2 Host On-Demand components that support Session Manager APIs	643
18.2.1 Deployment Wizard	644
18.2.2 Cached Client	644
18.2.3 Tracing	644
18.3 Example customer scenario	645
18.3.1 Instructions for embedding host sessions	647
18.3.2 Explanation of Session Manager APIs in this scenario	655
18.4 Description of working demonstration	658
Chapter 19. Host printing	661
19.1 Overview	662
19.1.1 Types of printer LU	664
19.2 3270 printer session	666
19.2.1 Configuring a 3270 printer session	667
19.2.2 Using a 3270 printer session	678
19.3 3270 associated printer sessions	682
19.3.1 Configuring associated printer sessions	682
19.3.2 How an associated printer session works	685
19.4 5250 printer session	686
19.4.1 Host Print Transform	687
19.4.2 Configuring a 5250 printer session	688
19.4.3 Using the 5250 printer session	690
19.5 3270 Host printing for DBCS	691
19.5.1 Host code page	691
19.5.2 Printer definition tables	691
19.5.3 Font image file	692
19.6 VT Host printing	693
19.7 Adobe PDF printing	694
19.8 Host Print Java Beans	695
Chapter 20. Macro support and enhancements	697
20.1 Macros	698
20.2 Creating and Editing Macros	698
20.2.1 Recording a simple Macro	698
20.2.2 Adding advanced functions to the simple Macro	703
20.3 HOD Administrators and Macros	712
20.3.1 Controlling Access to HOD Macros	713
20.3.2 Automatically Starting Macros	715
20.4 Deploying Macros	715

20.4.1	Setup Macros for Configuration Server or Combined Model	716
20.4.2	Deploying Macros for Configuration Server Model	717
20.4.3	Setup Macros for HTML based server model	718
20.4.4	Deploying Macros for HTML Model and Combined Model	719
20.5	When Problems Occur with HOD Macros	720
20.5.1	Causes of Screen Mismatches or Non-Matches	720
20.5.2	Optional field	722
20.5.3	Adding pauses	722
20.5.4	Adding timeouts	723
20.5.5	Use of Inverse Descriptor	724
20.5.6	Unexpected Screens from the Host	725
20.5.7	Screen attributes	727
20.5.8	Screen scroll problems - adding looping to Macro	727
20.6	Problem determination tools and strategies	729
20.6.1	Adding trace statements to Macros	729
20.6.2	Errors with HOD Macro variables	730
20.6.3	Using problem determination trace	731
Chapter 21.	Host Access Toolkit	739
21.1	Introduction	740
21.2	Host Access Toolkit requirements	742
21.2.1	Operating systems requirements	742
21.2.2	Supported target environments	743
21.3	Host Access Toolkit vs. Personal Communications	744
21.4	Host Access Class Library for Java (HACLJ)	745
21.4.1	HACLJ programming strategy	745
21.4.2	HACLJ functionality	747
21.4.3	Automated host navigation	752
21.5	Host Access Beans for Java overview	754
21.5.1	Host Access Java Beans explained	757
21.5.2	Automated host navigation	766
21.6	Java 2	769
21.6.1	Security	770
21.6.2	Permissions programming examples	774
21.6.3	Debugging runtime failures	777
21.7	Host On-Demand EHLLAPI Bridge	780
21.7.1	Operational configuration	781
21.7.2	Supported interfaces	781
21.7.3	Supported Platforms and JVM environments	781
21.7.4	Installation	782
21.7.5	Operation	784
21.8	Programming notes	785
21.8.1	JARs and CABs	785

21.8.2	Swing components and Host Access Beans	787
21.8.3	Subclassing and additional notes on licensing issues	787
21.9	Using the Loadable Applet Interface	788
21.9.1	setApplet(Applet theRealApplet) Is Called First	790
21.9.2	Deploying in a Java 1 Environment.	792
21.9.3	Deploying in a Java 2 Environment.	794
21.10	Host On-Demand J2EE Connectors	795
21.10.1	Why use Host On-Demand J2EE Connectors?	796
21.10.2	Host On-Demand J2EE Connectors development cycle	797
21.10.3	Host On-Demand J2EE Connectors classes	797
21.10.4	A sample program.	798
21.10.5	Documentation and more information	798
21.11	Additional help	799
Chapter 22.	Problem determination.	801
Part 2.	Personal Communications Version 5.6	803
Chapter 23.	Enhancements	805
23.1	New enhancements of Personal Communications 5.6	806
23.1.1	Windows XP compatibility	806
23.1.2	Accessibility enhancements	806
23.1.3	Detect and repair feature	807
23.1.4	Feature selection.	809
23.1.5	Installing Personal Communications using an ini file and using system variables and UNC paths	811
23.1.6	Improved security features	823
23.1.7	Tivoli support.	823
23.1.8	Sound functionality enhancements	823
23.1.9	Enhanced run java applet facility	823
23.1.10	Bidirectional RTL print orientation	824
23.1.11	Enhanced macro conversion utility	825
23.1.12	Arabic code page conversion for macro	826
23.1.13	Map printer setup to key and add button to menu.	826
23.1.14	ASCII text pdf	827
23.1.15	Enhanced SNA link configuration dialogs	827
23.1.16	Manual adapter selection	831
23.1.17	Drivers for communication adapters	831
23.1.18	Enhanced Mouse Marking	833
23.2	Previous enhancements made to Personal Communications 5.0	834
23.2.1	Windows 2000 certification	834
23.2.2	Session Manager	834
23.2.3	CSD and APAR tool	838
23.2.4	Tivoli support.	847

23.2.5	Defining the session view from a batch file	852
23.2.6	Convert macro to XML	853
23.2.7	Telnet 3270E printer association.	854
23.2.8	Win32 cut, copy, and paste hotkeys	858
23.2.9	Windows 2000 Power Management	858
23.2.10	1390/1399 code page support	860
23.2.11	Hindi support.	860
23.2.12	Edit wrap pasted text.	861
23.2.13	Express Logon Feature	863
23.2.14	Smart card support	864
23.2.15	Scaling	864
23.2.16	Higher DPI printer support	865
23.2.17	Capture view	866
23.2.18	NWSAA and ActiveX Support	866
Chapter 24.	Migration	867
24.1	Migration during installation	868
24.2	Migration Utility	873
Chapter 25.	Security	877
25.1	Enhancements in certificate management	878
25.1.1	Example of certificate management	879
25.2	Smart card support.	890
25.2.1	Enabling smart card support.	890
25.3	Express Logon Feature	894
25.3.1	Client setup	894
Chapter 26.	Programming interfaces.	897
26.1	Macros	898
26.1.1	Converting macros to XML	898
26.1.2	Import macros into Host On-Demand	899
26.1.3	Hiding logon passwords	902
26.2	EHLLAPI	904
26.3	HACL	908
Chapter 27.	Problem determination.	909
27.1	Operator information area (OIA)	910
27.2	Status bar.	912
27.3	Tracing / bundling problem determination data	913
27.4	Tracing from the command line	914
27.4.1	Example TCP/IP trace from command line	917
Part 3.	Screen Customizer	921

Chapter 28. Screen Customizer	923
28.1 Screen Customizer overview	924
28.2 Screen Customizer features	928
28.3 Screen Customizer components	929
28.4 Planning for Screen Customizer	930
28.5 Installation of Screen Customizer	931
28.5.1 Administrator and Customization Studio Installation	931
28.5.2 Runtime installation	933
28.5.3 Screen Customizer considerations for OS/390 and z/OS	935
28.5.4 Silent installation	940
28.6 Migration	941
28.7 The Screen Customizer development cycle	942
28.7.1 Screen Customizer Administrator	942
28.7.2 Using the Customization Studio	953
28.7.3 Template development	973
28.7.4 Deployment	978
28.8 Service Bundler	981
28.8.1 Windows system	982
28.8.2 Command-line interface	983
28.9 Application programming interface	984
28.9.1 Custom Terminal Bean	985
28.9.2 Screen Customizer Component Interface (SCCI)	985
28.9.3 Application programming interface documentation	985
28.10 More information	986
Part 4. Appendixes	987
Appendix A. Introduction to TCP/IP security	989
Basic concepts of cryptography and digital certificates	990
Symmetric encryption algorithms	990
Asymmetric encryption algorithms	993
Performance issues of cryptosystems	994
Cryptosystems for data integrity	995
Message digest algorithms	995
Message digests for data integrity	996
Message authentication codes	997
Digital signatures	998
Public Key Infrastructure	1000
Digital certificates	1001
Firewall concepts	1004
General guidelines for implementing firewalls	1005
Firewall categories	1006
Hardening	1009

Virtual private network (VPN) and IPSec	1010
IPSec	1011
Alternative VPN solutions: Layer 2 Tunnel Protocol	1017
Secure Sockets Layer	1018
SSL overview	1019
Establishing secure communications with SSL	1020
SSL considerations	1022
Transport Layer Security Protocol (TLS)	1023
Telnet-negotiated sessions	1023
Session configuration	1025
Appendix B. Service Location Protocol	1027
Load balancing	1028
Warm standby	1029
Appendix C. An example of MacrolOProvider	1031
Appendix D. Additional material	1035
Locating the Web material	1035
Using the Web material	1035
Java 1 demo (JSDemoEJ1.zip)	1036
Java 2 demo (JSDemoEJ2.zip)	1037
System requirements for downloading the Web material	1038
How to use the Web material	1038
Related publications	1039
IBM Redbooks	1039
Other resources	1040
Referenced Web sites	1041
How to get IBM Redbooks	1044
IBM Redbooks collections	1044
Special notices	1045
Abbreviations and acronyms	1047
Index	1051

Preface

This redbook will help you install, configure, administer and use the products distributed in the IBM Host Access Client Package Version 3. The package consists of:

- ▶ IBM WebSphere Host On-Demand Version 7.0
- ▶ IBM Personal Communications for Windows Version 5.6
- ▶ IBM Screen Customizer Version 2.0.70

Many enhancements have been made in these products. This book covers the features and functions of Host On-Demand Version 7 and IBM Personal Communications Version 5.6 for Windows.

The team that wrote this redbook

This redbook was produced by a team of specialists from around the world working at the International Technical Support Organization, Raleigh Center.

Michelle Barnes manages an IT team at ANZ Banking Group Ltd in Melbourne, Australia. She has 13 years experience in the IT industry. She currently works in z/OS communications supporting TCP/IP and a 10,000 user Host On-Demand multi-national networking environment.

Bob Bogardus is a Software Engineer for the IBM Software Group at Research Triangle Park. He holds a degree in Computer Science from SUNY Potsdam. He has 18 years of experience in a variety of areas related to iSeries systems management, logistics/finance/manufacturing solutions, networking and I/T security. In 1999, Bob was part of the team that ported Host On-Demand and Screen Customizer to the AS/400.

Alan Cohen is a Software Engineer for the IBM Software Group at Research Triangle Park. He has an BSEE degree from Duke University. Alan has over 20 years experience with IBM in various areas of networking development including hardware design and protocol development. Alan is currently working in the host Access Client support group.

Bill Gaither is a Software Engineer for the IBM Software Group at Research Triangle Park. He has worked as a developer on Host On-Demand and IBM Personal Communications. He has 20 years experience working on a variety of software products. He has an MS degree in Computer Science from Georgia Tech and a JD degree from Georgia State University School of Law.

Anna Murphy is a software documentation writer for the IBM Software Group at Research Triangle Park. She currently writes Host On-Demand documentation and is working toward a Master of Science in Technical Communication. Her areas of interest include human factors, Web design, and user interfaces.

Heinz-Joachim Zorn is a software support specialist in IBM Germany. He has ten years of experience in supporting IBM host connection products. He holds a degree in Communication Electronics from Fachhochschule Ruesselsheim. His areas of expertise include Personal Communication, Communication Server, Host On-Demand and LANDP.

Byron Braswell is a networking professional at the International Technical Support Organization, Raleigh Center. He received a B.S. degree in Physics and an M.S. degree in Computer Sciences from Texas A&M University. He writes extensively in the areas of networking and host integration software. Before joining the ITSO 2 years ago, Byron worked in IBM Learning Services Development in networking education development.

Thanks to the following people for their contributions to this project:

Margaret Ticknor
Bob Haimowitz
Gail Christensen
Tamikia Barrow
International Technical Support Organization, Raleigh

Bryan Aupperle
Wendell (Jason) Bouknight
Gregory Brodsky
David L. Bucheger
Robert Bunn
Carol Carson
John Chambers
Casey Cooley
Charlotte Davis
Andrew Hans
Landon Kirk
Susan Kirkman
Cynthia Krauss
Doug Larson

Steve Modena
Laura Monteleone
Srinivasan Muralidharan
Jenny Nicholson

Ki Park
James Quigley
John Ruiz
Michael Rutherford
Chuck Tharp
Shuichi (Scott) Yoshizawa
Mike J. White
Lydia Woodson
Robin Yehle
IBM Research Triangle Park, NC

Linda Harrison
IBM Advanced Technical Support, Gaithersburg

Special notice

This publication is intended to help I/T Specialists to plan for, install and use the products distributed in the Host Access Client Package Version 3. The information in this publication is not intended as the specification of any programming interfaces that are provided by IBM WebSphere Host On-Demand, IBM Screen Customizer, and IBM Personal Communications. See the PUBLICATIONS section of the IBM Programming Announcement for Host Access Client Package Version 3 for more information about what publications are considered to be product documentation.

IBM trademarks

The following terms are trademarks of the International Business Machines Corporation in the United States and/or other countries:

AFP™	PBT®
AFS®	RACF®
AIX®	Redbooks Logo 
AnyNet®	Redbooks™
APPN®	SAA®
AS/400®	S/390®
CICS®	SecureWay®
CICS Connection®	SP™
DB2®	SP1®
DB2 Connect™	SP2®
DFTS™	Tivoli®
DPI®	Tivoli Enterprise™
DRDA®	VisualAge®
e (logo)® 	VTAM®
Home Director™	WebSphere®
IBM®	World Registry™
iSeries™	z/OS™
LANDP®	zSeries™
MVS™	Lotus®
Netfinity®	1-2-3®
OS/2®	Approach®
OS/390®	Lotus Notes®
OS/400®	Notes®
PAL®	Domino™

Comments welcome

Your comments are important to us!

We want our IBM Redbooks to be as helpful as possible. Send us your comments about this or other Redbooks in one of the following ways:

- ▶ Use the online **Contact us** review redbook form found at:
ibm.com/redbooks
- ▶ Send your comments in an Internet note to:
redbook@us.ibm.com
- ▶ Mail your comments to the address on page ii.



Introduction to Host Access Client Package

The IBM Host Access Client Package is a solution for all of your host connection needs. The package provides:

- ▶ Access to applications and data on IBM @server iSeries (5250), IBM @server zSeries (3270), and DEC/UNIX virtual terminal (VT) hosts for traditional Web users in SNA and intranet environments
- ▶ Thin client technology to distribute host access capability to remote users, as well as users in intranet and extranet environments
- ▶ The ability to create new graphical user interfaces to front-end host information without programming, using drag and drop technology

The IBM Host Access Client Package offering integrates WebSphere Host On-Demand, IBM Screen Customizer and IBM Personal Communications into a single packaged solution. This package can provide access to legacy applications to virtually any type of user regardless of their needs. Whether it is an office-based power user, a mobile employee, a systems programmer, or a business partner with no training on host applications, the Host Access Client Package fits virtually any user need. With the addition of relatively simple graphical rejuvenation tools, you can make your applications easy to use and available to virtually anyone, virtually anywhere.

IBM Host Access Client Package offers a migration path from traditional emulation to Web browser-based emulation, with the ability to put a simple graphical user interface on a host application. This bundle focuses on TN5250, TN3270E, and VT applications support.

The components of this package are:

- ▶ IBM WebSphere Host On-Demand Version 7.0

Host On-Demand provides Java applets that enable host connectivity and terminal emulation for 3270, 5250, and VT displays, plus FTP, and JDBC access, using industry-standard TCP/IP protocols for communications to the host. The Java applets are downloaded from a Web server via a Web browser.

- ▶ IBM Personal Communications Version 5.6

Personal Communications provides host connectivity and emulation for Telnet 3270, Telnet 5250, and VT displays; supports SNA applications and technologies and TCP/IP.

- ▶ IBM Screen Customizer Version 2.0.70

Screen Customizer provides the ability to present host screens as a graphical user interface.

1.1 Host On-Demand

WebSphere Host On-Demand is targeted to customers who wish to provide easy, and cost-effective host access with security to users in intranet-based and extranet-based environments. It enables businesses to extend the reach of their host applications and data to new users, including business partners, suppliers, and sales personnel.

Host On-Demand gives users secure browser access to host applications and data, with Web browser-based emulation. With support for TN3270E, TN5250, VT and CICS applications included in a single package, users need to learn only one interface to reach key host data. Because Host On-Demand is Java based, users in different operating environments get the same look and feel, and identical feature set.

Host On-Demand is installed only on the Web server, and the Host On-Demand applet is downloaded via a Web URL to the user's Web browser. Code maintenance, updates and configurations all occur on the Web server, and users are updated automatically. The Host On-Demand cached client reduces download and user idle time, enhances productivity, and helps save significant expense in product deployment and maintenance. Host On-Demand works with IBM Screen Customizer, providing customized graphical screens in place of "green screens" to help simplify the user experience for users who may be unfamiliar with or prefer not to use the traditional host "green screens".

Host On-Demand can be installed on nearly any server platform to fit nearly any size organization or branch office. Host On-Demand is supported by your choice of platforms, ranging from Windows NT, Windows 2000, AIX and Linux servers to iSeries and zSeries mainframes. And, this benefit extends to the desktop as well. Because the interface is Java based, it is the same in every workstation environment, whether it be Windows 95, Windows NT, Windows XP, OS/2, Linux or other supported workstation environments.

A rich Java tool set, including Host Access Beans for Java and the application programming interface (API), can enable customers to rapidly create custom e-business applications to achieve a competitive advantage. Because Host On-Demand is part of the WebSphere family, applications developed using the tool set can be incorporated into other WebSphere software projects, helping preserve your Host On-Demand investment and helping provide a quick start to the Web and e-business.

Host On-Demand is recommended for installations that require low-cost centralized deployment, easy, centralized administration, and support for a broad range of client and server platforms. IBM Personal Communications is recommended for full-function emulation, more extensive APIs and a wider range of protocols or connectivity, including APPN and other SNA technologies.

1.1.1 Host On-Demand features

The features and functions that Host On-Demand provides have been increasing steadily with each new version. It is now a very powerful terminal emulator, FTP client and database-access utility, all implemented in Java and downloadable through a standard Web browser.

Features summary

Here is a summary of the main features followed by a list of features by release that have been added.

- ▶ TN3270, TN5250 and VT (VT52/100/420) emulation and a CICS Gateway client
- ▶ 3270 host printer emulation
- ▶ Database On-Demand for database query
- ▶ 3270 file transfer with MVS, VM, CICS
- ▶ File transfer with OS/400
- ▶ FTP client
- ▶ Keyboard remap
- ▶ Color remap
- ▶ Copy, cut and paste
- ▶ Print screen
- ▶ Macro record/play, with prompts and waits and a powerful editor
- ▶ Session security, through the Secure Sockets Layer (SSL) protocol
- ▶ Support for firewalls
- ▶ Server-based management of user configurations
- ▶ Usage (license) management
- ▶ Telnet redirection
- ▶ Host Access Class Library for development of network-computing applications
- ▶ Host Access Beans for Java for application development
- ▶ Translation into 20 languages, with keyboard and code page support for 20 more, including Arabic, Hebrew and Thai
- ▶ Comprehensive problem determination capability
- ▶ 26 sessions allowed
- ▶ Ability to create customized clients
- ▶ Locally installed client on Windows machines

Host On-Demand Version 7

The main emphasis in Version 7 is on enhanced Java 2 support and HOD portlet support of WebSphere Portal Server.

- ▶ User productivity enhancements
 - Customizable toolbar buttons and icons for HOD sessions
 - Macro enhancement to support variables, conditional processing, arithmetic expression assignment to a variable, and starting a macro within a macro
 - Session inactivity timeout for 3270 or 5250 display/printer sessions or VT sessions
 - Initial support for accessibility features to help users who have physical disabilities. Currently, not all features are available for all screens.
 - Associated printer session improvements to reduce the risk of 3270 client associated printer sessions being inadvertently shared by different pooled LUs
 - Remap graphics colors for 3270 sessions
 - Arranging configured session icons on the desktop by name or type
 - Custom function editor to define and maintain new keyboard functions without having to edit HTML and JavaScript files
 - URL hot spots can be displayed as underlined links or as three-dimensional buttons
 - Confirmation dialog on exit from a Host On-Demand session
 - Print screen enhancements for Java 2-enabled browsers to specify page orientation and margins, add headers and footers, and suppress the print dialog box
 - Support for the Start PC Command (STRPCCMD) in 5250 sessions
 - Support for grid lines defined by Data Description Specifications for DBCS 5250 display sessions
 - Cached client improvements to specify a separate upgrade percentage for peak and off-peak demand periods. A download size and time estimate will be displayed.
- ▶ Technology improvements
 - Java 2 support for cached clients running the Java 2 plug-in. Some HOD V7 features require Java 2.
 - Creation of Adobe Portable Document Format (PDF) versions of host documents
 - Support for Auto Input Method Editor (IME) feature when configuring 3270 and 5250 display sessions
 - Session Manager JavaScript APIs for managing host sessions and text-based interactions with host sessions
 - Socks 5 and HTTP proxy server support to transparently access host systems that are behind a firewall
 - Support for Transport Layer Security (TLS) protocol version 1
- ▶ Support for IBM WebSphere Portal Server as a portlet

- ▶ Administrator improvements
 - Stand-alone Deployment Wizard installation
 - Distribution of Deployment Wizard files using the DWunzip tool
 - Size requirement display when creating download clients
 - Deployment Wizard files published to a location other than the Host On-Demand publish directory
 - Customized Host On-Demand Web pages using custom HTML templates
 - Copy and paste new sessions to the Users/Groups window in the administration client
 - Support for dynamic HTML overrides of session properties
- ▶ FTP enhancements
 - Support for transferring directories to and from the host
 - FTP client Transfer List Manger toolbar support to create file or directory transfer lists
 - Support for renaming files or directories before they are transferred to the receiving system
 - Support for viewing server directory information
 - Enhanced FTP client support switching between MVS Services and HFS Services without changing the host type or defining two FTP sessions
 - Support for UTF-8 transfer type
 - Language selection for greetings and error messages for UTF-8 transfers

For a list of all functions and enhancements, see the online *Getting Started* documentation:

- ▶ On the CD:
[http://\[cdrom\]/doc/\[language\]/doc/install/install.html](http://[cdrom]/doc/[language]/doc/install/install.html)
- ▶ On Windows:
Click **Start -> Programs -> IBM Host On-Demand -> Information Library -> Getting Started**
- ▶ On disk:
[http://\[published directory\]/\[language\]/install/install.html](http://[published directory]/[language]/install/install.html)

For the latest information about Host On-Demand, visit the Web site at:

<http://www.ibm.com/software/webserver/hostondemand>

To subscribe to the Software Support Bulletin, go to:

<http://www.ibm.com/software/network/support>

Host On-Demand Version 6

The major emphasis in Version 6 was on improving the deployment options and user management. The main features are:

- ▶ More flexibility and control over locally stored user preferences
- ▶ Deployment Wizard enhancements
- ▶ Native Windows print support
- ▶ Integrated Windows domain logon
- ▶ Cached client support across the Internet
- ▶ Support for Java 2-enabled browsers
- ▶ Netscape 6.0 Support
- ▶ Enhanced Local Preferences
 - No user IDs for the Administrator to create
 - No logons needed by the client
- ▶ Copy a session between users and groups rather than import/export
- ▶ Reset Insert Mode on Aid Key (for example Enter, PFx, PAX)
- ▶ Reset on Enter
 - Additional parameters
- ▶ Customizable toolbar buttons
- ▶ Tab to next word and delete word
- ▶ Multi-user Cached Client

The cached client will be machine specific, not user specific, allowing several users to work on one machine

- ▶ User-defined character (UDC) mapping editor for DBCS environments
- ▶ Greek National Language Support
- ▶ Basic GB18030 code page support
- ▶ Problem determination enhancements
- ▶ Dropped support for OHIO

Host On-Demand Version 5

The emphasis in Version 5 focused on several areas, from improving cached client operations to improving administration and security and the ability to work in an Internet environment. The main features were:

- ▶ Administrative improvements
 - Automatic installation on AIX
 - Customizable Service Manager port
 - Configuration Servlet
 - Deployment Wizard
 - Disable functions to end users
 - Java 2 platform support (server side)
 - Host On-Demand toolkit package
- ▶ Security enhancements
 - Express Logon Feature
 - Native Authentication
 - AS/400 file transfer and Database On-Demand proxy and enhanced SSL support
 - Configuration Servlet

- Telnet-negotiated security
- Smart card support
- ▶ Client operations
 - ENPTUI support
 - Improved cached client
 - Improved enhancements to keyboard remapping
 - Multiple session icon
 - Hindi enablement
 - Blink attribute support and color remap improvements

1.1.2 Host On-Demand components

Host On-Demand consists of the following components:

- ▶ One or more Web servers
- ▶ A Java environment, provided by a Java virtual machine (JVM)
- ▶ The Host On-Demand Server
- ▶ A Web application server (optional)
- ▶ Clients
- ▶ Deployment Wizard
- ▶ Certificate management utilities
- ▶ Host Access Class Library for Java
- ▶ Host Access Beans for Java
- ▶ Screen Customizer default graphical user interface

Web servers

Since the Host On-Demand applet must be downloaded from a server to the client, a Web server must be installed on the same machine as the Host On-Demand server. Any Web server will work; we have not yet found a Web server that does not work.

On Windows NT, Windows 2000, Windows XP and AIX, the Host On-Demand installation program will detect and configure most Web servers. Configuration consists of creating the required alias for the Host On-Demand publish directory so that the applet and other files are available to clients.

The client code is available for download once Host On-Demand is installed and the Web server configured. If the Configuration Server model is used (see 13.1, “Host On-Demand configuration models” on page 514), then the Host On-Demand Service Manager is utilized to configure and manage users and sessions.

Java virtual machine

Since Host On-Demand consists of a set of Java applets and applications, a Java environment is required on both the server and the client. This Java environment is provided by a Java virtual machine (JVM).

For Windows clients, the Java virtual machine is installed along with the Host On-Demand code. Host On-Demand V7 installs the IBM 1.3 JVM. For all other servers the JVM must be obtained and installed separately.

Clients are supported on Java 2-enabled Web browsers, such as Netscape 6.x and Mozilla. The Java 2 Plug-in with Netscape 4.x and Microsoft Internet Explorer is also supported. For more information refer to 5.15, “Web browsers: Java 1 and Java 2 enabled” on page 221.

Host On-Demand server

The Host On-Demand server consists of multiple sub-functions each of which is discussed in detail in Part 1, “IBM WebSphere Host On-Demand” on page 21.

- ▶ Configuration Server

The Configuration Server manages configuration data on a user and group level.

- ▶ Redirector service

The Redirector is a Telnet proxy that may be used to protect internal Telnet servers and ports from Internet users or to provide security for Telnet servers that do not support security natively.

- ▶ OS/400 Proxy

The OS/400 Proxy is a proxy that allows clients to connect to a back-end AS/400 system via a single port, instead of exposing the address and ports of the AS/400 to end users.

- ▶ Configuration Servlet

The Configuration Servlet is very useful in Internet environments. The servlet runs under control of a Web application server and is configured to relay communications between the client and Configuration Server over standard HTTP(S) connections.

- ▶ Native Authentication

This is a function that is used to authenticate Host On-Demand defined users with the native operating system of the server upon which the Host On-Demand server is running. This is supported only on Windows NT, Windows 2000, Windows XP, AIX, OS/390 and z/OS operating systems.

- ▶ License Use Management

This function keeps track of concurrent Host On-Demand usage.

Web application server

A Web application server, such as IBM WebSphere Application Server V3.5 or V4, Lotus Domino R5 or R6, Netscape IPlanet (JRun) Version 4.1, iPlanet Web Server Enterprise Edition V6, iPlanet Application Server V6 or HTTP Server V1.3.6.2, V1.3.6.4, V1.3.12.6, V1.3.19.2, or V2 is required to run the Configuration Servlet.

Clients

Host On-Demand provides several types of preconfigured clients. Note that an emulator client can support all of the listed terminal types from a single full-function client.

- ▶ Emulator clients
 - 3270 display
 - 3270 print
 - 5250 display
 - 5250 print
 - VT
- ▶ FTP client
- ▶ Database clients
- ▶ Administrative clients
- ▶ Utility clients

Host On-Demand provides these full-function clients as either a cached client or as a download client, or both, with normal or debug options. In addition, the Deployment Wizard utility allows for the creation of custom emulation clients.

All available clients are explained in detail in Chapter 5, “Clients” on page 163.

Deployment Wizard

The Deployment Wizard is a utility that allows the administrator to build an emulation client customized for their environment. By using the Deployment Wizard to build custom clients, the administrator can affect the size of the downloaded or cached client, restrict the client to a subset of functions that they may perform, and establish the security capabilities of the client. The list below is a partial list of the capabilities of the Deployment Wizard.

- ▶ Type(s) of emulation allowed:
 - 3270 display
 - 3270 print

- 5250 display
 - 5250 print
 - VT
- ▶ Capabilities of the emulator:
 - Keyboard remap
 - Color remap
 - File transfer
 - Macro play/record
- ▶ Cached or download client
- ▶ Normal or debug options
- ▶ User model
 - Configuration Server
 - HTML-based
- ▶ Portlet creation

For a complete discussion of the Deployment Wizard, refer to Chapter 14, “Deployment Wizard” on page 529.

Certificate management utilities

The certificate management utilities are installed on Windows and AIX platforms during installation. These applications provide the capability of managing digital certificates and keyring files used by the Host On-Demand clients and Redirector on supported platforms. Refer to Chapter 12, “Certificate management” on page 487 for additional information.

Host Access Class Library for Java

The Host Access Class Library (HACL) for Java, delivered on the Host On-Demand Toolkit CD, provides a set of classes and methods that allow the development of platform-independent applications that can access host information at the data stream level. HACL implements the host access function in a complete class model that is independent of any graphical display and requires only a Java-enabled browser or comparable Java environment.

With HACL, application developers can write Java applets that manipulate data from the data stream presentation space (such as 3270, 5250, and VT) without requiring that the applets reside on the client machines.

HACL is a significant improvement over traditional emulator programming interfaces, such as EHLLAPI, in several respects. It is an object-oriented API, with all the well-known benefits of the object-oriented programming paradigm and it requires far fewer statements to achieve the same result.

For more details, refer to Chapter 21, “Host Access Toolkit” on page 739.

Host Access Beans for Java

The Host Access Beans for Java provide emulator functions as a set of JavaBeans. JavaBeans are components that have configurable properties that use events to communicate between each other, and that can be manipulated in visual development environments. The Host Access Beans can be used by developers to rapidly develop custom applications that deliver the specific functions they want to include in their host-access applications.

For more details, refer to Chapter 21, “Host Access Toolkit” on page 739.

Screen Customizer default graphical interface

Host On-Demand has the ability to automatically render the classic “green screen” into a basic graphical user interface. A default graphical user interface, Screen Customizer/LE, is included in all the Host On-Demand clients and can be turned on in the session configuration panels, but this cannot be customized. In order to be able to customize this interface, you must install the full IBM Screen Customizer product that is distributed with the Host Access Client Package and then customize the screens using the customization capabilities of the product. For information on the full IBM Screen Customizer product, refer to Part 3, “Screen Customizer” on page 921.

1.1.3 Architecture and operations

Figure 1-1 illustrates the basic operations of Host On-Demand. We briefly describe the components and how Host On-Demand works, referring the reader to more specific chapters.

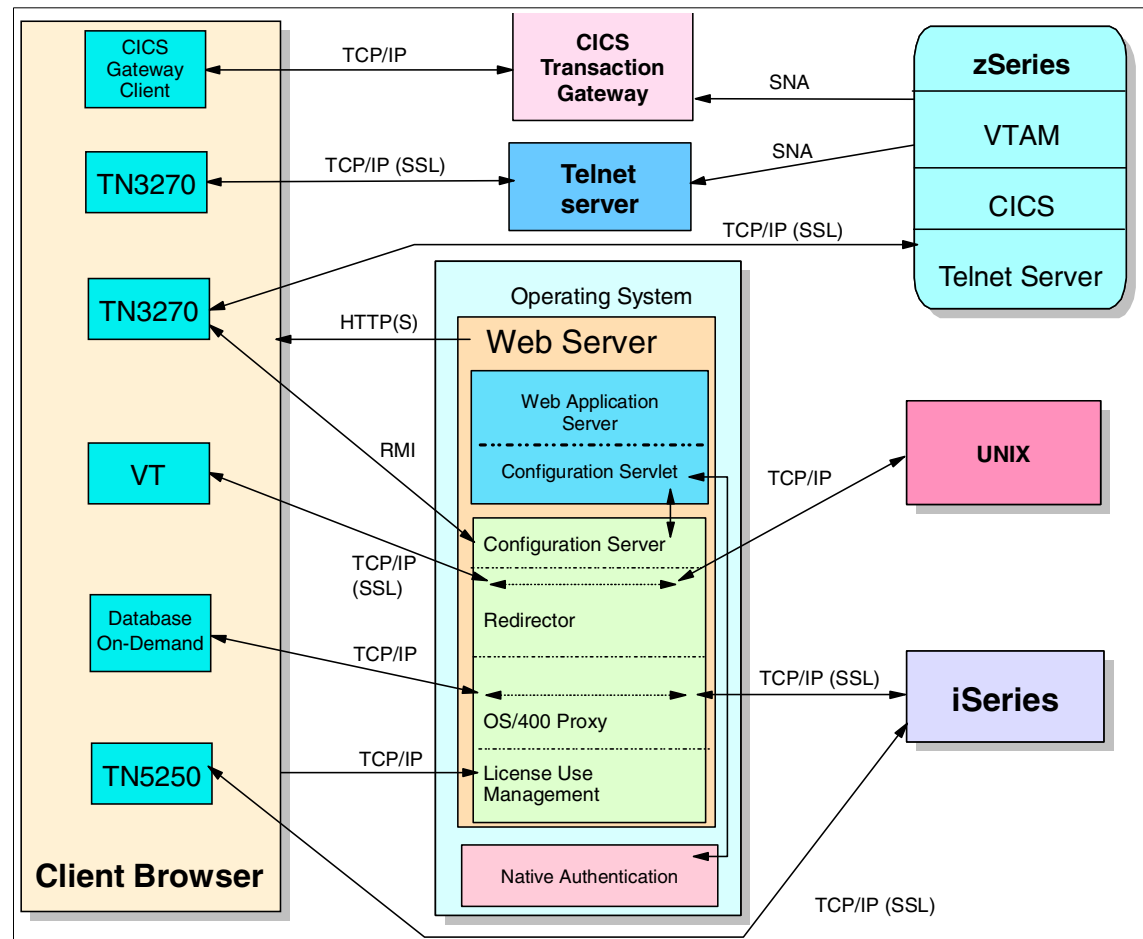


Figure 1-1 Host On-Demand basic operations

Host On-Demand is installed on a Web serving platform. This platform can be almost any platform that supports a Web server, and Java.

A client browser running a Java virtual machine contacts the Web server and requests an HTML page that has a Host On-Demand client embedded. This connection to the Web server may optionally be secured with HTTPS.

Configuration information for the client is either downloaded with the HTML page or is obtained from the Host On-Demand Configuration Server component via a non-secure TCP RMI connection. To secure this configuration information and to simplify administration in an Internet environment with firewalls, a Configuration Servlet running under a Web application server can be configured and HTTP(S) used to pass the configuration information as opposed to the non-secure RMI connection.

If configuration information is obtained from the Host On-Demand server and not via the HTML files, then the ability to store these preferences using Windows user IDs is supported. Additionally, the ability is provided to challenge the user for their user ID and password as known by the server upon which Host Access Client Package is running.

Once the configuration information is obtained, any of the clients shown in Figure 1-1 may be invoked by the user in the following ways:

- ▶ Connections are supported to the CICS Gateway using the CICS Gateway client using a non-secure TCP/IP connection
- ▶ Standard TN3270(E) and TN5250 connections are supported to any of the following:
 - Stand-alone Telnet servers via secure or non-secure connections
 - Host On-Demand Redirectors
 - Communications Server for AIX Redirectors via secure, non-secure, or passthrough connection
 - Direct to an iSeries or zSeries server via secure or non-secure Telnet protocols
- ▶ VT sessions are supported in the following ways:
 - Directly to a supported VT host via non-secure Telnet connections
 - To a Redirector via a secure Telnet connection that can then connect to the back-end non-secure VT host
- ▶ Database On-Demand connections to an iSeries host in the following ways:
 - Direct to the iSeries via secure or non-secure TCP/IP connections
 - Via the Host On-Demand OS/400 Proxy via a non-secure TCP/IP connection

In order to provide the administrator a mechanism for insuring that the company stays within its licensing agreement, the License Use Management function is provided on the Host On-Demand server. This function, when enabled, provides a count of the maximum number of users concurrently running a Host On-Demand client at any given point in time.

This architecture and operations is fully described in Part 1, “IBM WebSphere Host On-Demand” on page 21.

1.2 Introduction to Personal Communications Version 5.6

Host Access Client Package V3 ships with Personal Communications Version 5.6 for Windows. Personal Communications Version 5.6 is Windows XP certified and runs on Windows 95, Windows 98, Windows ME, Windows NT, Windows 2000, and Windows XP. Both products are market leaders for traditional client access and connectivity, providing connectivity and APIs for all environments.

Personal Communications brings the power of personal networking to your workstation by exploiting networking capabilities to provide a variety of connectivity options supporting local area network (LAN) and wide area network (WAN) environments. Whether it is for host terminal emulation, client/server applications, or connectivity, Personal Communications offers a robust set of communication, networking, and administrative features.

Personal Communications is a full-function emulator. In addition to host terminal emulation, it provides these useful features:

- ▶ File transfer
- ▶ Dynamic configuration
- ▶ An easy-to-use graphical interface
- ▶ APIs for SNA-based client applications
- ▶ An API allowing TCP/IP-based applications to communicate over an SNA-based network

A variety of SNA-based client application programming interfaces (APIs) are supported by Personal Communications. You can create applications that use the peer-to-peer client APIs, which are based on LU 6.2 and provided by Personal Communications. These APIs let you simultaneously access and process information on peer workstations.

Personal Communications supports Advanced-Peer-to-Peer Networking (APPN) as an end node, and uses the advanced network features: high-performance routing (HPR) and dependent LU requester (DLUR).

AnyNet SNA over TCP/IP is a feature of Personal Communications that allows emulator and client/server SNA applications to communicate over a TCP/IP network.

Refer to Part 2, “Personal Communications Version 5.6” on page 803 for more information.

1.3 Introduction to Screen Customizer

IBM Screen Customizer is a Java-based Web client for IBM WebSphere Host On-Demand and IBM Personal Communications. It transforms the traditional character user interface of mainframe green-screen applications to nice-looking, presentable and user-friendly graphical application. It supports 3270 and 5250 and CICS Gateway-based emulation sessions; however, Screen Customizer support is not available for VT display sessions.

It interprets the host data stream that comes through either Host On-Demand or Personal Communications Version 5.6 and transforms it to a default basic graphical screen. Provision is available for customization of these screens to generate more sophisticated graphical screens. The transformation happens in real-time thus ensuring there are no compromises on the speed and performance of the client system and the underlying legacy application.

Extensive customization is possible with the help of the Studio and Administrator tools, by which the green screens can be transformed to modern graphical application. Thus IBM Screen Customizer gives a new lease on life to those ageing character user interface-based legacy applications. All this is possible without the need to do any programming or modification of the host-based legacy applications.

Refer to Part 3, “Screen Customizer” on page 921 for more information.

1.4 National Language Support

IBM, as a true international leader of global information technology, has a global vision for all its products. As a part of its globalization strategy, IBM ensures all its products, communications and services are designed to meet the needs of the global market.

This means that they must support the language, culture, and character encoding elements of IBM's worldwide customers. The strategy is to do this in a consistent and comprehensive manner, and at the same time leveraging existing international standards where possible.

National Language Support (NLS) provides a standardized method of supporting multiple international locales, code pages, input methods, sort orders, and number/currency/time/date formats.

1.4.1 The need for National Language Support

As the global economy becomes more integrated, users require software compatibility across multicultural and multilingual barriers. They want to run applications using their own language and local conventions for time display, menu selections, and error messages.

A large corporation with several branches offices around the world may require a its applications to have interfaces for more than one language, may be a mixture of English, Japanese, and French software environments with, perhaps, multiple languages supported in a single site.

Enterprises with such complex requirements require a unified system-software architecture that can support global networks without the incompatibilities often found with different localized versions of software. Not only do they require unified system-administration models and policies, but they also need to be able to develop internal applications that operate without modification across all their operations.

For an application to be successful and to reach the global marketplace, it has to made in accordance to the local customs, conventions and other requirements. These could be culturally specific, such as the way date is represented and the way decimals are rounded.

In accordance with the IBM globalization strategy for its products, the Host Access Client Package is translated to several languages to ensure its global reach. The individual products such as Host On-Demand, Screen Customizer and Personal Communications Version 5.6 are provided in many languages. The session windows, configuration panels, help files, and the documentation have been translated. In addition, display, keyboard, and processing support is provided for Arabic, Hebrew, Thai, Hindi, and Greek.

The languages and code pages supported by Host On-Demand and Screen Customizer are listed in the National Language Support section of the online *Installation and Planning Guide*. There is also a list of the suffixes required if you want to load non-native versions of the client applets.

Support in terms of code pages, fonts, screens and keyboard functions has been included for a host of languages such as Arabic, Hebrew, Hindi, and Thai, as well as for the Euro, the new currency of the European Union. For a complete and comprehensive list of supported languages and their levels of support, please refer to the installation guide provided along with the software.

Note: National Language Support is operating-system dependent. It calls for the availability of the necessary font and keyboard support for the language you want to use and it should be installed in the operating system. For example, if you want to use French as the host-session language but do not have the French font and keyboard support installed, you may not be able to display the correct characters.

1.4.2 Globalization de-mystified

This section attempts to explain some of the technical terms/jargon used.

National Language Support Supporting multiple international locales, code pages and input methods to a software product using a standards based approach that makes the software product culturally correct and closer to the user.

Internationalization The process of designing software applications so that it can be adapted to various languages and regions without engineering changes. Sometimes the term internationalization is abbreviated as i18n, because there are 18 letters between the first “i” and the last “n.”

Localization The process of adapting software for a specific region or language by adding locale-specific components and translating text. The term localization is often abbreviated as l10n, because there are 10 letters between the “l” and the “n.” Usually, the most time-consuming portion of the localization phase is the translation of text. Other types of data, such as sounds and images, may require localization if they are culturally sensitive. Localizers also verify that the formatting of dates, numbers, and currencies conforms to local requirements.

DBCS Double-Byte Character Set, which is used in Asia Pacific countries, including Japan, Korea, Taiwan (Traditional Chinese), China (Simplified Chinese), and Hong Kong (Simplified Chinese). As the name suggests, it may be assumed that DBCS simply means a character-encoding scheme where each character occupies two bytes. However, depending on the scheme such as EBCDIC, ASCII, ISO2022, and Unicode,

	each character could occupy one or more bytes in the DBCS environment.
Code Page (CP)	A code page (CP) is the specification of code points (hexadecimal value) for each character in a character set. Each CP has its own ID number. For example, Japanese Kanji has a Code Page ID number 300.
Unicode	Provides a unique number for every character, irrespective of the platform, or program or language.

1.4.3 Installation

When you install any or all the products of the Host Access Client Package on a Windows server or on an iSeries, you can choose which languages you wish to be installed.

Note: For an exhaustive list of supported languages and code pages, please refer to the online *Getting Started Guide* for the product you are using.



Part 1

IBM WebSphere Host On-Demand

In this part, we review Host On-Demand Version 6.0. We address all aspects of the product from installation planning to deployment strategies, from administration to client operations, and migration issues from prior versions of Host On-Demand. Installation and deployment on zSeries are discussed in considerable detail.

2



Planning and installation

This chapter discusses the planning for an installation of Host On-Demand on all platforms except zSeries. Installation on a zSeries is significantly different from installation on distributed platforms; therefore, we will discuss zSeries planning and installation in Chapter 3, “z/OS implementation” on page 79.

2.1 Supported platforms

The following paragraphs describe the platforms that are supported by Host On-Demand and the server disk space requirements.

2.2 Server requirements

2.2.1 zSeries platform

For a complete list of OS/390 and z/OS requirements, see the Program Directory.

2.2.2 iSeries platform

Table 2-1 iSeries server requirements

Server operating system	OS/400 (R) V4R5, V5R1, and V5R2. Recent cumulative service is recommended. Refer to the OS/400 Fixes, Downloads and Updates Web page for service information.
Disk space	410 MB DASD
Memory	256 MB memory or more. Refer to the iSeries Performance Capabilities Reference Web page for additional information about the impact of additional memory and Java performance
Supported Web servers	<ul style="list-style-type: none">▶ Apache-based HTTP Server for iSeries▶ IBM HTTP Server for iSeries▶ Lotus Domino for iSeries
Java	IBM Java Toolbox Java Developer's Kit *BASE option and one of the following <ul style="list-style-type: none">▶ Option 4 - 1.1.8▶ Option 5 - 1.3
All other requirements	TCP/IP Connectivity Utilities for iSeries QShell Interpreter

2.2.3 Windows Platforms

Table 2-2 Windows server requirements.

Server operating systems	<ul style="list-style-type: none"> ▶ Windows NT 4.0 with SP5 or later ▶ Windows 2000 Professional, Server, and Advanced Server ▶ Windows XP Professional (32-bit) (Note: This should not be used for a large scale production server.)
Disk space	340 MB for an English-only installation. Add 4 to 8 MB for each additional national language to be installed.
Supported Web servers (automatically configured)	<ul style="list-style-type: none"> ▶ IBM HTTP Server ▶ IBM Internet Connection Server ▶ Lotus Go, Domino, and Domino Go ▶ Microsoft Internet Information Server 3 and 4 ▶ Microsoft Peer Web Services ▶ Microsoft Personal Web Server
Java	Installed with Host on Demand For XP see Note

Note: To get the latest JVM update for Windows XP install the latest Service Pack for XP via Windows Update. (SP1 contains Java2 update)

Alternate sites to get Java support:

<http://www.ibm.com/java>

<http://java.sun.com/getjava/download.html>

2.2.4 AIX Platform

Table 2-3 AIX server requirements.

Server operating system	AIX (R) Version 4.3.3 and 5L 5.1
-------------------------	----------------------------------

Disk space (installp image)	310 MB for an English-only installation. Add 4 to 8 MB for each additional national language to be installed (including the additional security files).
Supported Web servers	<ul style="list-style-type: none"> ▶ Apache Web Server ▶ IBM HTTP Server
Java	JVM 1.1.8 or 1.3

You can obtain the latest AIX JVM from one of the following Web sites:

<ftp://ftp.hursley.ibm.com/pub/java/>

<http://www.ibm.com/java>

2.2.5 Solaris platform

Table 2-4 Solaris server requirements.

Server operating system	Sun Solaris 2.6, 2.7, and 2.8
Disk space	278 MB for an English-only installation. Add 4 to 8 MB for each additional national language to be installed.
Supported Web servers	<ul style="list-style-type: none"> ▶ Apache Web Server ▶ IBM HTTP Server
Java	JVM 1.1.8 or 1.3

You can obtain the latest Solaris JVM from one of the following Web sites:

<ftp://ftp.hursley.ibm.com/pub/java/>

<http://www.ibm.com/java>

2.2.6 HP-UX platform

Table 2-5 HP-UX server requirements.

Server operating system	Sun Solaris 2.6, 2.7, and 2.8
-------------------------	-------------------------------

Disk space	278 MB for an English-only installation. Add 4 to 8 MB for each additional national language to be installed.
Supported Web servers	<ul style="list-style-type: none"> ▶ Apache Web Server ▶ IBM HTTP Server
Java	JVM 1.1.8 or 1.3

You can obtain the latest HP-UX JVM from one of the following Web sites:

<ftp://ftp.hursley.ibm.com/pub/java/>

<http://www.ibm.com/java>

2.2.7 Linux and other Unix platform

Table 2-6 Linux server requirements.

Server operating system	<ul style="list-style-type: none"> ▶ Red Hat Linux 6.2, 7.0, 7.1, 7.2, and 7.3 ▶ SuSE Linux 6.4, 7.0, 7.1, 7.2, 7.3, and 8.0 ▶ Caldera 2.3 and 3.1 ▶ TurboLinux 6.0, 6.5, and 7.0 ▶ Unixware 7
Disk space	278 MB for an English-only installation. Add 4 to 8 MB for each additional national language to be installed.
Supported Web servers	<ul style="list-style-type: none"> ▶ Apache Web Server ▶ IBM HTTP Server
Java	JVM 1.3 or 1.4

You can obtain the latest Linux JVM from the following Web site:

<http://www.ibm.com/java>

When using Redhat Linux Version 7.0, make sure that the glibc package is at least Version 2.2-12. In addition, make sure the IBM JDK is at least J2RE 1.3.0 IBM Build cx130-20010207.

2.2.8 OS/2 platform

Table 2-7 OS/2 server requirements.

Server operating system	<ul style="list-style-type: none"> ▶ OS/2 (R) Warp Server Version 4 ▶ OS/2 Warp Server for e-Business 4.5
Disk space	410 MB. The hard disk must be configured for HPFS.
Supported Web servers	Lotus Domino Go Web server for OS/2
Java	OS/2 JVM 1.1.8 or JVM 1.3.

You can obtain the latest OS/2 JVM from one of the following Web sites:

<ftp://ftp.hursley.ibm.com/pub/java/>

<http://www.ibm.com/java>

For JVM 1.1.8, make sure your classpath entry in config.sys is updated with the location of the JVM class files and that the current directory (.) is included. The classpath should include something like this:

c:\Java11\lib\classes.zip;

Note: When you have installed the JDK and set the classpath, reboot the workstations that the updated classpath takes effect.

2.2.9 Novell Netware platform

Table 2-8 OS/2 server requirements.

Server operating system	Novell NetWare Version 4.2, 5.1, and 6
Disk space	410 MB.
Supported Web servers	Novell Web Server
Java	Novell Java Development Kit 1.1.8.

You can obtain the latest Novell JDK at <http://www.developer.novell.com>. The JDK must be configured for long-filename support.

Note: For users to load the client HTML files from a Novell server, their browsers might need to be configured not to use a proxy server. In addition, if users have a browser with a Java 2 plug-in, the IBM plug-in must be 1.3.0 or later and the Sun plug-in must be version 1.3.1 or later. The client applets do not successfully load if the plug-in is an earlier version.

2.2.10 Supported LDAP servers

The Host On-Demand server can optionally use the lightweight directory access protocol (LDAP) as a data store for user and group information. The following LDAP servers are supported:

- ▶ IBM LDAP Directory Server V2.1, V3.1.1 V3.2.1, V3.2.2
- ▶ IBM LDAP Server running on OS/390 V2R9, V2R10
- ▶ IBM LDAP Server running on z/OS V1R1, V1R2, V1R3, V1R4
- ▶ Netscape Directory Server V3.1 and V4.0 (Windows NT and AIX)

For more information on IBM's LDAP Directory solution and to download a complimentary evaluation kit, go to

<http://www.software.ibm.com/network/directory/>

For instructions on using LDAP with Host On-Demand, see Chapter 18, "Configuring Host On-Demand Server to use LDAP" on page 129.

2.2.11 Web servers

The following Web servers are supported:

- ▶ WebSphere Application Server 3.5, 4.0
- ▶ Lotus Domino R5, R6
- ▶ Netscape iPlanet (JRun) V4.1
- ▶ iPlanet Web Server Enterprise Edition V6.0
- ▶ iPlanet Application Server V6.0
- ▶ IBM HTTP Server V1.3.6.2, V1.3.6.4, V1.3.12.6, V1.3.19.2, V2.0

2.3 Client requirements

For updates to client requirements, refer to the Readme file, [readme.html](#).

2.3.1 Supported operating systems

Host On-Demand clients are supported on the following operating systems:

- ▶ Windows 95
- ▶ Windows 98
- ▶ Windows Millennium Edition (ME)
- ▶ Windows NT 4.0 with SP5 or later
- ▶ Windows 2000 (Professional)
- ▶ Windows XP Professional and Home Edition (32-bit version)
- ▶ AIX 4.3.3, AIX 5L 5.1
- ▶ OS/2 Warp 4
- ▶ Sun Solaris 2.6, 7, and 8
- ▶ HP-UX 10.20, 11.00, and 11i
- ▶ Red Hat Linux 6.2, 7.0, 7.1, 7.2, and 7.3
- ▶ SuSE Linux 6.4, 7.0, 7.1, 7.3, and 8.0
- ▶ Caldera 2.3 and 3.1
- ▶ TurboLinux 6.0, 6.5, and 7.0
- ▶ Windows Terminal Server Version 4
- ▶ Windows Terminal Services for 2000
- ▶ Netstation V2R1M0
- ▶ Citrix Metaframe 1.8 for Windows Terminal Server 4.0 and 1.8 for Windows 2000
- ▶ Server
- ▶ Citrix Metaframe XP (Versions s,a,e) for Windows

Note: A local client is supported only on Windows XP, Windows NT, Windows 2000, Windows 95, Windows 98, and Windows Millennium (installed from win32 using setup.exe lc)

Certain newer versions of Windows do not ship with Java support. Host On-Demand does not have a way to detect whether Java exists unless Java is already present on the workstation; therefore, your clients will not be directed to the Web page to download and install the Java 2 plug-in. Thus, if you plan to roll out these versions of Windows on your client machines and want to use the Java 2 functions listed above, it is recommended that you install the Java 2 plug-in before rolling out the client machines.

Restricted users do not have the authority to install the Java 2 plug-in. Someone with administrative authority must load the Java 2 plug-in.

2.3.2 Supported browsers

The following browsers are supported for you to download the Host On-Demand clients from a remote Host On-Demand server or to run Host On-Demand on a locally installed client:

- ▶ Netscape Navigator 4.6, 4.7, 6.1, 6.2
- ▶ Netscape Navigator (OS/2) 4.61
- ▶ IBM Web Browser for OS/2 V1.2
- ▶ Microsoft Internet Explorer 4.01 with SP1, 5.0 or 5.1, 5.5, 6.0
- ▶ Sun and IBM Java plug-in 1.3, 1.3.1, and 1.4

If you are using a Java 2-enabled Web browser, such as Netscape 6.x or the IBM Web Browser for OS/2, several restrictions on Host On-Demand functions apply when using the predefined HTML pages (HOD.html). For more information on these limitations, the most up-to-date list of supported Web browsers, refer to the Readme and to the Host On-Demand Web site.

<http://www.ibm.com/software/webservers/hostondemand>

Note: Based on our experience during the tests we strongly suggest that you use the latest browser versions.

Microsoft Technical Document, Q163637, available on Microsoft's Web site, provides information for Internet Explorer users on how to obtain the latest JVM for Microsoft Windows 98 operating system, Microsoft Windows 98 Second Edition operating system, Microsoft Windows Millennium Edition operating system, Microsoft Windows NT(R) 4.0 operating system, and Microsoft Windows 2000 operating systems.

For Windows XP operating system, without a JVM installed, reference the Microsoft support news group for Windows XP operating system, located on Microsoft's Web site

2.4 Installing Host On-Demand

The Host On-Demand clients are served as Web pages, so you must install the Host On-Demand server on a system with a Web server. The installation steps are different for each of the following operating systems:

- ▶ Installing on Windows NT, Windows 2000 or Windows XP Servers and XP
- ▶ Installing on OS/2
- ▶ Installing on Novell NetWare
- ▶ Installing on AIX
- ▶ Installing on UNIX (Solaris, HP-UX, and Linux)
- ▶ Installing on iSeries
- ▶ Installing on OS/390 or z/OS

2.4.1 Installing on Windows NT, Windows 2000 or Windows XP Servers

A Web server is required to install Host On-Demand on Windows NT, Windows 2000 or Windows XP. The Web servers which are recognized and automatically configured during installation are listed in Table 2-2 on page 25.

You can install Host On-Demand with a graphical interface using the Windows InstallShield or with a response file using Windows InstallShield in silent mode. You have to have administrator rights on the machine to install HOD!

The installation (and de-installation) is logged in HODINSTALL.1log which is located in

`c:\Documents and Settings\userid\Local Settings\Temp`

Installation using InstallShield

To automatically install Host On-Demand on a Windows NT, Windows 2000 or Windows XP server using the InstallShield, follow the steps below.

1. Log in as Administrator or a user that is a member of the administrators group.
2. If CD autoplay is enabled on your Windows NT, Windows 2000, or Windows XP server, insert the CD and wait for the start window. Otherwise, insert the CD and run the setupwin.exe program in the root directory. Figure 2-1 shows the new Host On-Demand Version 7 Windows installation welcome window from which you can choose your next step.

Note: For directly installing Host On-Demand you can run from the win32 subdirectory the setup.exe. Without any parameter it will install the HOD server. With the 1c parameter it will install the local client.

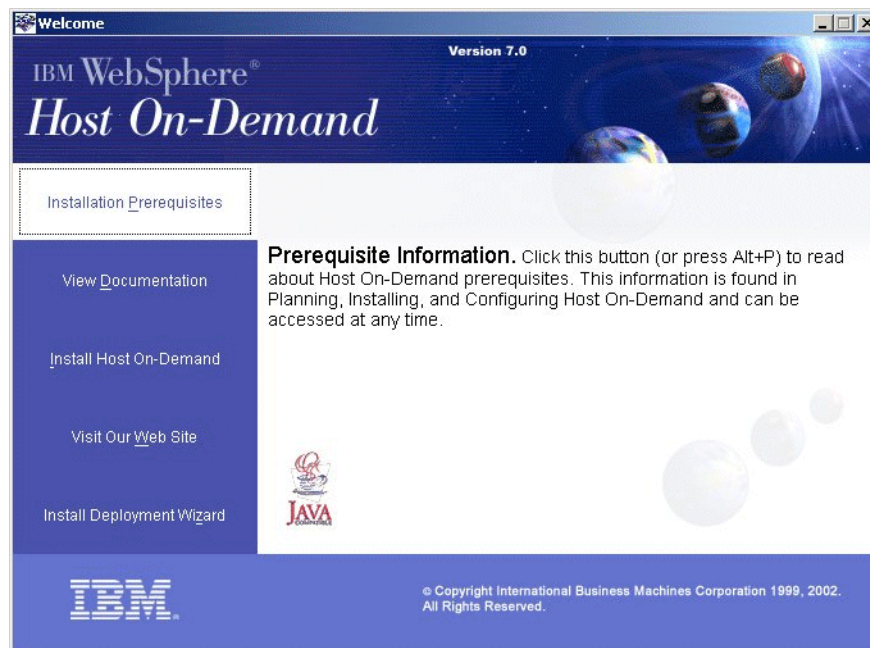


Figure 2-1 Windows installation welcome window

3. Click Install.

4. Follow the directions in the installation windows. You will reach the language selection panel as shown in Figure 2-2. The English language is always selected by default. Using the english language setup is very helpful when discussing problems with IBM support.

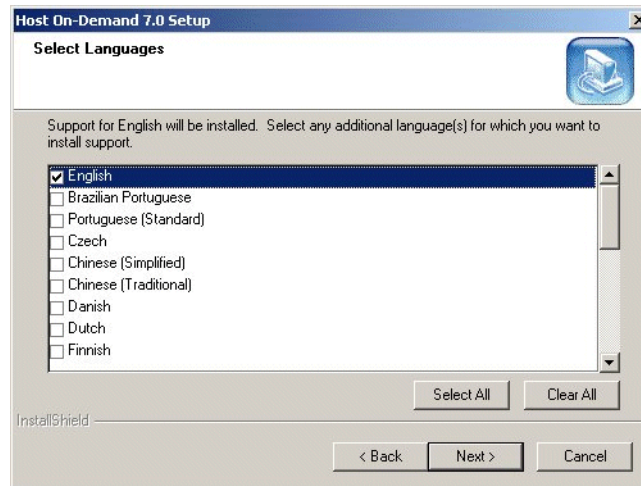


Figure 2-2 Language selection panel

Continue following the next selection panels until you reach the panel showing you what is selected for installation.

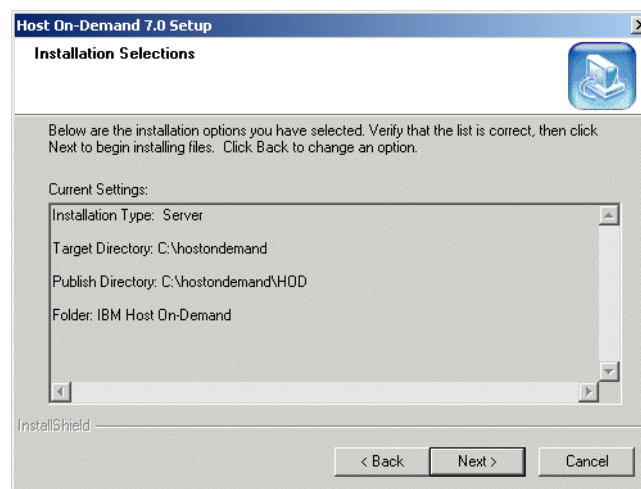


Figure 2-3 Panel showing the activated selections for installation

The installation will take place according to your selections. After that the installed Web Server is detected. Select your server and continue.

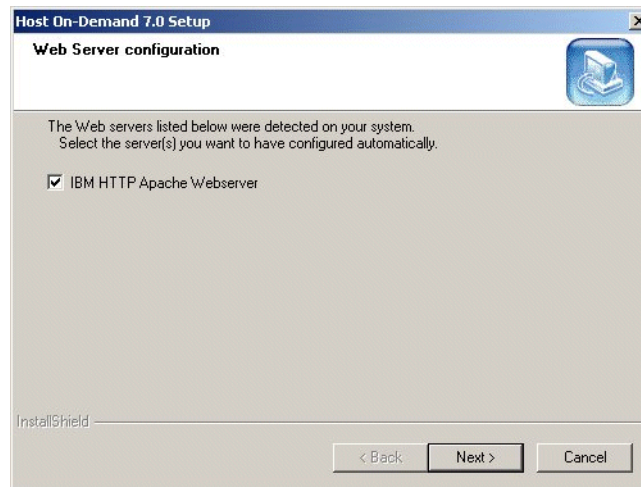


Figure 2-4 Detecting the installed web server

- ▶ The default server directory is \hostondemand. If you are upgrading, the installation program uses the same server directory as before. The server directory contains files used only by the server and must not be available to client workstations.
- ▶ The default publish directory is \hostondemand\H0D. The publish directory contains files that must be available to client users who access the server through a browser.

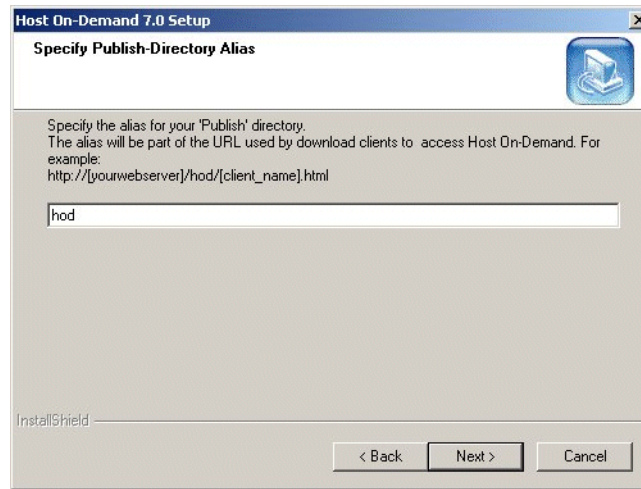


Figure 2-5 Published directory alias

- The default Service Manager port is 8999, and it is usually a safe port to select. Check your server documentation to see if this port is being used. If it is in use, you can change the port during installation, or later. For more information about changing the Service Manager port, see Changing the Service Manager's configuration port in the online help and Chapter 2.4.8, "Changing the Configuration Server port" on page 56.

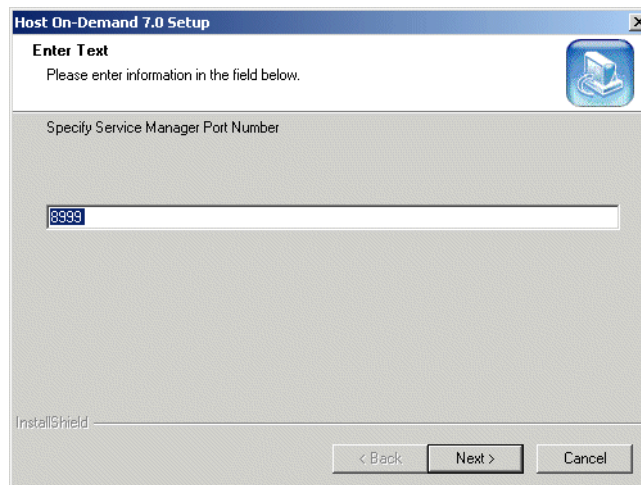


Figure 2-6 Selecting the Service Manager port

- If the installation program detects that IBM WebSphere Application Server is installed, you are asked if you want to use the configuration servlet to connect

to the configuration server for client configuration information. If you are running Host On-Demand through a firewall, this eliminates the need to open an extra port for the configuration server. Answering **Yes** automatically configures the clients to access the configuration server through the configuration servlet. Answering **No** configures the clients to access the configuration server directly on port 8999. See “Installing the Configuration Servlet” on page 60 for more information.

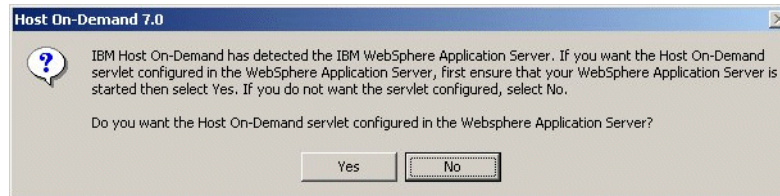


Figure 2-7 Select to use configuration servlet

Note: If you would like to set up your server so that some clients are using the Configuration Servlet and others do not, select **Yes** at this time and refer to Chapter 9, “Configuration Servlet” on page 397 for more information.

5. A window appears giving you the option to register your software and view the *Planning, Installing, and Configuring Host On-Demand* guide.
6. If a message tells you that your Web server is not recognized or was not configured, configure it. If you install a Web server later or your Web server is not recognized by Setup, you must publish the Publish directory to the Web. Refer to the Web server documentation for information on how to publish the directory.

For IBM HTTP Server for Windows, the entry in the `httpd.conf` file will probably look like this:

```
Alias /hod/ C:/hostondemand/HOD/
```

7. Restart the Web server to pick up the configuration changes.
8. Load the `HODMain.html`, located in the published directory into your browser as shown in Figure 2-8 on page 38. This page contains links to all the Host On-Demand clients and utilities, the readme file, and basic configuration steps for configuring the Host On-Demand server.

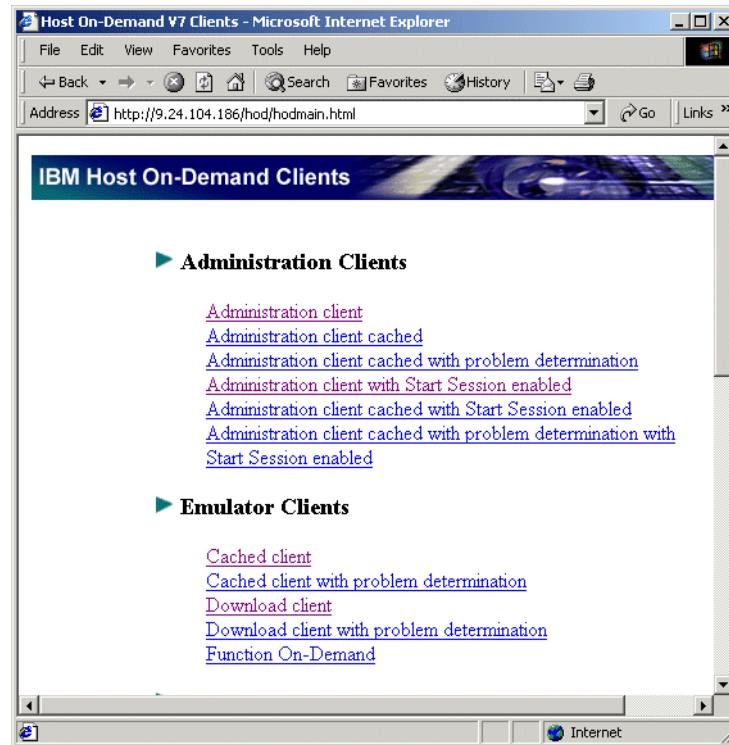


Figure 2-8 Started HODMain web page

At the end of installation, the Host On-Demand Service Manager is started automatically.

Installation in silent mode

A silent installation installs Host On-Demand without displaying any windows or asking for input. All of the input required during an installation is obtained from a text file called a response file. A response file is created by recording an installation. A sample response file is included in the `instmgr` directory of the product CD as `install.script`. The defaults are English, no WebServer configuration, no WebSphere configuration, and a port value of 8999. If these values are not correct and there is not a GUI-capable console available to record a new response file, the `install.script` file may be copied to a writable directory and manually edited based on the comments included in the `install.script` file.

A local client cannot be installed silently.

Note: When you install in silent mode, there is because of this mode no indication on the screen or taskbar that the installation is in progress or that it is complete. In task manager you will find the IKERNEL.EXE running as long as the install process is active.

The directory for the response file must be defined prior to the start of the installation.

Use the setup.exe from the WIN32 subdirectory. To record a response file:

setup.exe -r -f1c:\temp\server1.iss

The resulting server1.iss response file which was created using the default values and answers for the HOD7 server installation is shown in Example 2-1 on page 39.

Example 2-1 Recorded response file

```
[InstallShield Silent]
Version=v6.00.000
File=Response File
[File Transfer]
OverwrittenReadOnly=NoToAll
[{ABFAF859-1718-11D5-AF27-0060945545E0}-DlgOrder]
Dlg0={ABFAF859-1718-11D5-AF27-0060945545E0}-SdWelcome-0
Count=12
Dlg1={ABFAF859-1718-11D5-AF27-0060945545E0}-SdAskOptionsList-0
Dlg2={ABFAF859-1718-11D5-AF27-0060945545E0}-SdDlgDestinationPath-0
Dlg3={ABFAF859-1718-11D5-AF27-0060945545E0}-SdAskDestPath-0
Dlg4={ABFAF859-1718-11D5-AF27-0060945545E0}-SdSelectFolder-0
Dlg5={ABFAF859-1718-11D5-AF27-0060945545E0}-SdStartCopy-0
Dlg6={ABFAF859-1718-11D5-AF27-0060945545E0}-AskOptions-0
Dlg7={ABFAF859-1718-11D5-AF27-0060945545E0}-AskText-0
Dlg8={ABFAF859-1718-11D5-AF27-0060945545E0}-AskYesNo-0
Dlg9={ABFAF859-1718-11D5-AF27-0060945545E0}-MessageBox-0
Dlg10={ABFAF859-1718-11D5-AF27-0060945545E0}-AskText-1
Dlg11={ABFAF859-1718-11D5-AF27-0060945545E0}-AskYesNo-1
[{ABFAF859-1718-11D5-AF27-0060945545E0}-SdWelcome-0]
Result=1
[{ABFAF859-1718-11D5-AF27-0060945545E0}-SdAskOptionsList-0]
Component-type=string
Component-count=1
Component-0=English
Result=1
[{ABFAF859-1718-11D5-AF27-0060945545E0}-SdDlgDestinationPath-0]
szDir=C:\hostondemand
Result=1
[{ABFAF859-1718-11D5-AF27-0060945545E0}-SdAskDestPath-0]
```

```

szDir=C:\hostondemand\HOD
Result=1
[{ABFAF859-1718-11D5-AF27-0060945545E0}-SdSelectFolder-0]
szFolder=IBM Host On-Demand
Result=1
[{ABFAF859-1718-11D5-AF27-0060945545E0}-SdStartCopy-0]
Result=1
[Application]
Name=WebSphere Host On-Demand Server
Version=7.00.000
Company=IBM
Lang=0009
[{ABFAF859-1718-11D5-AF27-0060945545E0}-AskOptions-0]
Result=1
Sel-0=1
[{ABFAF859-1718-11D5-AF27-0060945545E0}-AskText-0]
szText=hod
Result=1
[{ABFAF859-1718-11D5-AF27-0060945545E0}-AskYesNo-0]
Result=1
[{ABFAF859-1718-11D5-AF27-0060945545E0}-MessageBox-0]
Result=1
[{ABFAF859-1718-11D5-AF27-0060945545E0}-AskText-1]
szText=8999
Result=1
[{ABFAF859-1718-11D5-AF27-0060945545E0}-AskYesNo-1]
Result=1
[{ABFAF859-1718-11D5-AF27-0060945545E0}-SdFinish-0]
Result=1
bOpt1=0
bOpt2=0

```

Again use the setup.exe from the win32 subdirectory to install in silent mode:

setup.exe -s -f1c:\temp\server1.iss -f2c:\temp\server1.log

The resulting server1.log looks after the successful installation using the above server1.iss is shown in Example 2-2 on page 40.

Example 2-2 Log file contents after successful silent installation

```

[InstallShield Silent]
Version=v6.00.000
File=Log File
[ResponseResult]
ResultCode=0
[Application]
Name=WebSphere Host On-Demand Server
Version=7.00.000

```

Company=IBM
Lang=0009

All options and their usage are listed in Table 2-9.

Table 2-9 Options supported in silent mode

Command Line Option	Description
-r	Records a response file.
-s	Runs a response file to install Host On-Demand.
-f1[path\response_file_name].iss	Defines the response file, in both record and run modes. The path and filename must be 43 characters or fewer. There must not be a space between parameter and value. The filename extension must be iss.
-f2[path\log_file_name]	Defines the log file and can be used in run mode to create a file that contains a history of an installation. The path and filename must be 43 characters or fewer. There must not be a space between parameter and value.

The system’s configuration of the target (to which HOD will be installed via silent mode) must be the same as that of the source system (the system on which the response file was created). For example, if the source system has a previous installation of Host On-Demand, the target system must have the same. If the source system installed Host On-Demand on the D drive, the target system must also have a D drive. The source and target systems must have the same number of Web servers, although they do not need to be the same types.

The playback of the response file relies on exactly the same windows, in the same sequence and with the same contents to appear during the installation as they appeared during recording. So both systems have to match. If during install e.g. a windows from another application would pop up that would interfere with the sequence and this silent install would not be successful. In such cases response code of -12 would be recorded in the log file.

If an installation is not successful, the log file may indicate the reason. The log file of the silent install does not contain exhaustive details for the cause of the problem. In some cases it might be needed to run the installation on the target machine manually with Install Shield for visual comparison of the sequence and contents of pop-up-windows with the machine where the response file was recorded.

The ResultCode entry indicates whether or not the installation was successful. Possible values are shown in Table 2-10.

Table 2-10 ResultCode values

Value	Description
-0	Successful
-1	General error
-2	Mode not valid
-3	Required data not found in the response file
-4	Not enough memory available
-5	File does not exist
-6	Cannot write to the response file
-7	Cannot write to the log file
-8	Path to the response file is not valid
-9	Not a valid list type (string or number)
-10	Data type is not valid
-11	Unknown error during setup
-12	Dialogs are out of order. Since the dialog order depends on what other related products were already installed on the workstation, the target system must have the same products.
-51	Cannot create the specified folder
-52	Cannot access the specified file or folder
-53	Selected option is not valid

Tips: The following are common problems that have appeared during tests and do not necessarily lead to an error message:

- ▶ The **setup.iss** file is not in the directory specified by the **-f1** option.
- ▶ You changed the name or location of the **setup.iss** file and did not specify the new name or location when you ran the **setup.exe** command to install the product.
- ▶ There is not enough space on the specified target drive to install the product.
- ▶ You are installing or uninstalling Host On-Demand and you are not logged on to the target machine with Administrator authority.
- ▶ There is an error in the syntax of the **setup.exe** command.
- ▶ The environment to which you are installing the HOD is not the same as when you did record the response file. So the panels and answers as recorded in iss file do not match as they occur (invisible) on the installing system

Note to Reviewer: I have no resources / skill to prove the contents for the following operating systems. Byron is taking over the AIX part, I had asked Bob Bogardus to check the AS400 part - Achim Zorn 08/28

2.4.2 Installing on OS/2

You can obtain the latest JVM level from:

<http://www-105.ibm.com/developerworks/tools.nsf/dw/java-all-byname>

Note: If you have previously installed Host On-Demand and have changed `/hostondemand/private/NSMprop` or changed or created `/hostondemand/hod/config.properties`, you must back up these files before installation and then restore them after installation. The files are overwritten during the unzip process.

The following steps assume that `hostondemand` is the server directory and HOD is the publish directory. To install the Host On-Demand server:

1. Insert the CD.

2. If this is a new installation, create a server directory, for example C:\hostondemand. The server directory contains files that are used only by the server and must not be available to client workstations

3. Change to the server directory.

4. Run the following command to extract the files:

```
unzip [cd_rom]:\zip\hod70srv.zip
```

Where:

- unzip is your unpacking program (such as UNZIP.EXE). It must support long filenames.
 - [cd_rom] is the CD-ROM drive letter.
 - zip is the directory on the CD.
5. If this is a new installation, create the publish directory, for example C:\hostondemand\HOD. The publish directory contains files that must be available to client users who access the server through a browser.
 6. Change to the publish directory.
 7. Run the following command to extract the files:

```
unzip [cd_rom]:\zip\hod70www.zip
```
 8. Make the publish directory available to clients on the network. Refer to the documentation of your Web server for information about how to publish a directory.
 9. Configure a local host by adding the following line to the setup.cmd file, which is usually found in the \mptn\bin directory:

```
ifconfig lo 127.0.0.1
```
 10. Start the Host On-Demand Service Manager, which provides support services for Host On-Demand and runs as a Java application:
 - a. At the command prompt, change directory to \hostondemand\lib.
 - b. Copy NCServiceManager-OS2.cmd from the \hostondemand\lib\samples\CommandFiles directory.
 - c. Edit NCServiceManager-OS2.cmd to reflect the directory paths appropriate for your workstation.
 - d. Run NCServiceManager-OS2.cmd. The Service Manager does not display a message indicating that it has started. Also, disregard the following message: *Native library failed to load, indicating this Redirector does not support SSL*. The failure to load this library simply indicates that the server does not support SSL sessions.

Note: For Host On-Demand to function, the Service Manager must be running. If you reboot the server, you must also restart the Service Manager. You might want to add the `NCSERVICEManager-OS2.cmd` command to your `startup.cmd` file so that the Service Manager starts automatically when the workstation boots. If you do, remember to specify the path to change directory to the `\hostondemand\lib` subdirectory before the command runs.

11. Restart the Web server to pick up the changes in the configuration
12. Load `HODMain.html`, located in the `\hostondemand \HOD` directory, into your browser.
 - Click **Readme** to see updated information.
 - Click **Basic configuration steps** to help you get started with configuring the Host On-Demand server.

To start the Service Manager automatically, you might want to:

- ▶ Add the command to your `startup.cmd` file so that the Service Manager starts automatically when the workstation boots.
- ▶ If you do, include logic to change directory to the `\hostondemand\lib` subdirectory before the command runs.

2.4.3 Installing on Novell NetWare

To obtain the Novell JDK, go to <http://www.developer.novell.com>. The JDK must be configured for long-filename support.

Note: If you have previously installed Host On-Demand and have changed `/hostondemand/lib/NSMprop` or changed or created `/hostondemand/hod/config.properties`, you must back up these files before installation, then restore them after installation. The files are overwritten during the unzip process.

These steps assume that `hostondemand` is the server directory and `HOD` is the publish directory. To install the Host On-Demand server:

1. Stop the Service Manager with the `Java -exit` command.
2. From a client workstation, map a drive to the `SYS:` volume of the Novell server.
3. Mount the `SYS:volume`.
4. Insert the CD.

5. If this is a new installation, create a server directory, for example `hostondemand`. The server directory contains files that are only used by the server and must not be available to client workstations.
6. Change to the server directory.
7. From the drive mapped to the SYS:volume, run the following command to extract the files:

```
unzip [cd_rom]:\zip\hod70srv.zip
```

Where:

- `unzip` is your unpacking program (such as UNZIP.EXE). It must support long filenames.
 - `[cd_rom]` is the CD-ROM drive letter.
 - `zip` is the directory on the CD.
8. Change to `SYS:\web\docs`. This directory is usually published (made available to client users who access the server through a browser) automatically. If the `\web\docs` directory does not exist, create a publish directory, for example `HOD`, change to that directory, and go to Step 10.
 9. Create a publish directory named `HOD` and change to that directory. The `HOD` directory contains files that must be available to client users who access the Host On-Demand server through a browser.
 10. Run the following command to extract the files:

```
unzip [cd_rom]:\zip\hod70www.zip10
```
 11. From the server console, run the command `load java` to start the Java NLM.
Start the Host On-Demand Service Manager, which provides support services for Host On-Demand and runs as a Java application, by following these steps from a client system mapped to the SYS volume of the server:
 - a. Copy `NCServiceManager-Novell.ncf` from the `\hostondemand\lib\samples\CommandFiles` directory to the `\system` directory on the Novell server. To run the command from the server console, you might have to change the file name to the eight-dot-three format.
 - b. Edit `NCServiceManager-Novell.ncf` (or the eight-dot-three format of the file) to reflect the directory paths that are correct for your workstation.
 - c. From the server, run `NCServiceManager-Novell.ncf` (or the eight-dot-three format of the file). The Service Manager does not display a message indicating that it has started.

Note: For Host On-Demand to function, the Service Manager must be running. If you reboot the server, you must also restart the Service Manager.

2.4.4 Installing on AIX

You can automatically install Host On-Demand through a graphical interface, or through an ASCII control file in silent mode.

The automatic installation verifies the presence and version of required products before installation occurs. If a prerequisite is missing, the action taken by the Install Manager will depend on the policy setting in the control file.

Installation using the graphical interface

To install the Host On-Demand server on an AIX workstation using the graphical interface, follow the steps below.

1. Insert the CD and mount the CD-ROM drive.
2. Start the installation program by changing to the root directory of the CD and run the `setupaix.sh` script. You may need to type `./setupaix.sh` if the current directory (`.`) is not set in your `PATH` variable. The AIX install program window appears.
3. You may click **View Documentation** to see the product information (including the installation instructions).

Note: Make sure you have configured Netscape such that it can be run by the installation program. Specifically, before running `setupaix.sh`, ensure that the Netscape executable is in your `PATH` (e.g. `/usr/local/netscape`), and that `MOZILLA_HOME` is set to the appropriate directory (e.g. `/usr/local/netscape`).

To check for the latest version of Netscape Communicator please check the following URL: <http://techsupport.services.ibm.com/aix/efixes/netscape/>

4. Click **Install Product**
5. Follow the directions in the installation windows.
 - ▶ The default server directory, determined by the installation program, is `/usr/opt/hostondemand`. The server directory contains files used only by the server and must not be available to client workstations.
 - ▶ The default publish directory, determined by the installation program, is `/usr/opt/hostondemand/HOD`. The publish directory contains files that must be available to client users who access the server through a browser.
 - ▶ The default Configuration Server port is 8999, and it is usually a safe port to select. Check your server documentation to see if this port is being used. If it is in use, you can change the port during installation or later. For more

information about changing the Configuration Server port, see 2.4.8, “Changing the Configuration Server port” on page 56.

- ▶ If the installation program detects IBM WebSphere Application Server, Lotus Domino Go Webserver or IBM Domino Go Webserver installed, you are asked if you want to use the Configuration Servlet to connect to the Configuration Server for client configuration information. If you are running Host On-Demand through a firewall, this eliminates the need to open an extra port for the Configuration Server. Selecting **Yes** automatically configures the clients to access the Configuration Server through the Configuration Servlet. Selecting **No** configures all clients to access the Configuration Server directly on its port (default=8999).

Note: If you would like to set up your server that some clients are using the Configuration Servlet and others do not, select **Yes** at this time and refer to Chapter 9, “Configuration Servlet” on page 397 for more information.

6. Click **Finish** to end the installation.
7. If a message tells you that your Web server was not recognized or was not configured, configure it. If you install a Web server later or your Web server was not recognized by the Install Manager, you must publish the Host On-Demand Publish directory to the Web. Refer to the Web server documentation for information on how to publish the directory.
8. Restart the Web server to pick up any configuration changes.

Note: For Host On-Demand to function, the Service Manager must be running. If you reboot the server, you must also restart the Service Manager. To arrange for this script to be run at boot time, refer to the documentation supplied with your operating system on how to add a boot service.

Installation in silent mode

A silent installation installs Host On-Demand without displaying any windows or asking for input. All of the input required during an installation is obtained from a text file called a response file. A response file is created by recording an installation.

When you install in silent mode, there is no indication that installation is in progress or that it is complete.

Below are sample command lines that will install Host On-Demand on an AIX workstation in silent mode. The silent mode installation installs Host On-Demand in the /usr/opt directory, creates hostondemand as the server directory and HOD as the publish directory. The examples assume that you mounted the CD-ROM drive as /cdrom.

Note: The following commands must be on one line. Before issuing any of the following commands, change into the instmgr directory, for example:

```
cd /cdrom/instmgr
```

To install in silent mode using the install.script from the CD and record a log file called HodInstall.log:

```
/cdrom/instmgr/installaix.sh -p /cdrom/instmgr/install.script  
>/tmp/HodInstall.log
```

To record a response file:

```
/cdrom/instmgr/instaix.sh -r /tmp/install.script
```

To playback the response:

```
/cdrom/instmgr/instaix.sh -p /tmp/install.script
```

Note: When you install in silent mode, there is no indication that installation is in progress or that it is complete.

Table 2-11 Options supported in silent mode

Command Line Option	Description
-r	Records a response file.
-p	Runs a response file to install Host On-Demand.
/path/response_file_name	Defines the name for the response file. The default is install.script, and a sample install.script file is provided in the /instmgr directory on the Host On-Demand CD. Any file name can be used if properly specified on the command line used to execute the installation process.

The target system's configuration must be the same as that of the source system (the system on which the response file was created). For example, if the source system has a previous installation of Host On-Demand Version 7, the target system must have the same. If the source system installed Host On-Demand to a /usr/opt/hostondemand directory, the target system must also have a /usr/opt/hostondemand directory. The source and target systems must have the same number of Web servers, though they do not need to be the same type.

2.4.5 Installing on UNIX (Solaris, HP-UX, and Linux)

Supported Java versions for UNIX may be obtained from:

<http://www.ibm.com/developerworks/tools.nsf/dw/java-devkits-byname>

Note: If you have previously installed Host On-Demand and have changed /hostondemand/lib/NSMprop or changed or created /hostondemand/HOD/config.properties, you must back up these files before installation and restore them after installation.

To install the Host On-Demand server on a UNIX workstation, follow the steps below. These examples assume that you are installing Host On-Demand in the /usr/opt directory and that hostondemand is the server directory and HOD is the publish directory. Adjust the statements to match your environment.

1. Insert the CD and mount it.
2. Change to the /usr/opt directory and create a server directory, for example /hostondemand. The server directory will contain files that are used only by the server and must not be available to client workstations.

```
cd /usr/opt
mkdir hostondemand
```
3. Change to the server directory, and untar the files from hod70srv.tar to the server directory. Tar files are located in the /cdrom/tar directory.

```
cd hostondemand
tar -xf /cdrom/tar/hod70srv.tar
```
4. Create the publish directory HOD and change to it. It will contain the files that must be available to client users who access the server through a browser.

```
mkdir HOD
cd HOD
```
5. Untar the files from hod70www.tar to the publish directory using the following command:

```
tar -xf /cdrom/tar/hod70www.tar
```

English language support is installed by default. If you want additional language support, untar the appropriate language file from the /cdrom/tar directory into the publish directory. For example, to install Spanish language support:

```
cd /usr/opt/hostondemand/HOD
tar -xf /cdrom/tar/hod_es.tar
```

6. Make the publish directory, /usr/opt/hostondemand/HOD available to clients on the network. Refer to the documentation of your Web server for information about how to publish a directory.
7. Start the Host On-Demand Service Manager, a Java application that provides support services for Host On-Demand and runs as a Java application:
 - a. Change directory to the /usr/opt/hostondemand/lib subdirectory.
 - b. Copy NCServiceManager-UNIX from the /usr/opt/hostondemand/lib/samples/CommandFiles directory.
 - c. Edit NCServiceManager-UNIX to reflect the directory paths that are correct for your system.

Note: Make sure the NCServiceManager-UNIX file has execute permission.

- d. Run NCServiceManager-UNIX. The Service Manager does not display a message indicating that it has started. To arrange for this script to be run at boot time, refer to the documentation supplied with your operating system to add a boot service. Also, disregard the following message: Native library failed to load, indicating this Redirector does not support SSL. The failure to load this library simply indicates that the server does not support SSL sessions.

Note: For Host On-Demand to function, the Service Manager must be running. If you reboot the server, you must also restart the Service Manager. To arrange for this script to be run at boot time, refer to the documentation supplied with your operating system on how to add a boot service.

8. Restart the Web server to pick up any configuration changes.
9. Load HODMain.html, located in the hostondemand/HOD directory, into a browser.
 - ▶ Click **Readme** to see the latest information.
 - ▶ Click **Basic configuration steps** to help you get started with configuring the Host On-Demand server.

2.4.6 Installing on iSeries

Installing Host On-Demand on iSeries is a two step process:

1. Install the Host On-Demand server software.
2. Configure the HTTP server.

For information about the impact of additional memory and Java performance, refer to the iSeries Performance Capabilities Reference Web page:

<http://www.ibm.com/servers/eserver/iseries/perfmgmt/resource.htm>

Recent cumulative service is recommended. Refer to the Fixes, Downloads and Updates Web page:

<http://techsupport.services.ibm.com/eserver/fixes>

Search for iSeries.

Install the software

1. Sign on to the iSeries with the QSECOFR user profile (or user profile with equivalent security authorities).
2. If Host On-Demand was previously installed, issue the following OS/400 command to shut down the Service Manager:

```
ENDHODSVM
```

3. If you previously installed Host On-Demand, do the following:
 - a. Type the following command to migrate the NSMprop file to the correct location for Host On-Demand 7:

```
MOV OBJ('/QIBM/ProdData/hostondemand/lib/NSMprop ')
TODIR('/QIBM/ProdData/hostondemand/private ')
```

- b. Type the following commands to back up the current settings:

```
CRTSAVF QGPL/HOD
CALL QCMD
```

- c. Press the F11 key and enter the following command on line 2 of the window:

```
SAV DEV('/qsys.lib/qgpl.lib/hod.file ')
OBJ('/qibm/proddata/hostondemand/private/*')
('/QIBM/ProdData/hostondemand/hod/*.html ')
('/QIBM/ProdData/hostondemand/hod/hoddata/*')
('/QIBM/ProdData/hostondemand/hod/custom/*')
('/QIBM/ProdData/hostondemand/hod/config.properties ')
('/QIBM/ProdData/hostondemand/hod/CustomizedCAs.class ')
('/QIBM/ProdData/hostondemand/lib/com/ibm/as400/access/keyring.class '))
```

Note: The line beginning with SAV and ending with keyring.class ')) should be one line on your command line. One or more .Object not found....(CPFA0A9) messages may appear if the config.properties or hodddata files are not on your system.

4. Place the Host On-Demand for OS/400 CD in the iSeries CD drive.
5. Type the following OS/400 command:

```
RSTLICPGM LICPGM(5733A59) DEV(OPT01)
```

This command will process for 10-45 minutes, depending upon the configuration of the iSeries.

6. For each additional OS/400 secondary language for which you would like to provide full help text support, type the following OS/400 command:

```
RSTLICPGM LICPGM(5733A59) DEV(OPT01) LNG(yyyy) RSTOBJ(*LNG)
```

Where yyyy is the language code from the list below. This step is optional and can be performed after installation.

Table 2-12 Language codes

Language	Language code
Belgian Dutch	2963
Belgian English	2909
Belgian French	2966
Brazilian Portuguese	2980
Canadian French	2981
Chinese (simplified) PRC	2989
Chinese (traditional) Taiwan	2987
Czech	2975
Danish	2926
Dutch Netherlands	2923
English	2924
English DBCS (uppercase)	2938
English (uppercase)	2950
English DBCS	2984

Language	Language code
Finnish	2925
French	2928
French Multinational	2940
German	2929
German Multinational	2939
Greek	2957
Hungarian	2976
Italian	2932
Italian Multinational	2942
Japanese Kanji DBCS	2962
Korean DBCS	2986
Norwegian	2933
Polish	2978
Portuguese	2922
Portuguese Multinational	2996
Russian	2979
Slovenian	2911
Spanish	2931
Swedish	2937
Turkish	2956

7. Install IBM Screen Customizer 2.0.70 for iSeries (if you are planning to use the full runtime features). If you have previously installed IBM Screen Customizer, you must install the new version at this time. Refer to the installation manual for Screen Customizer.
8. If you want the Host On-Demand Service Manager to automatically start after an IPL (when QSYSWRK is started), type the following OS/400 command:

```
CFGHODSVM AUTOSTART(*YES)
```

To view the status of the Host On-Demand Service Manager, type the following OS/400 command:

```
WRKJOB QHODSVM
```

9. To restore the former configuration settings (if Host On-Demand was previously installed), do the following:

- a. type the following command:

```
CALL QCMD
```

- b. Press the F11 key and enter the following command in line 2 of the window:

```
RST DEV('/qsys.lib/qgpl.lib/hod.file ')
OBJ(('/qibm/proddata/hostondemand/private/*')
('/QIBM/ProdData/hostondemand/hod/config.properties ')
('/QIBM/ProdData/hostondemand/hod/hoddata/*')
('/QIBM/ProdData/hostondemand/hod/custom/*')
('/QIBM/ProdData/hostondemand/hod/config.properties ')
('/QIBM/ProdData/hostondemand/hod/CustomizedCAs.class ')
('/QIBM/ProdData/hostondemand/lib/com/ibm/as400/access/keyring.class '))
OUTPUT(*PRINT)ALWOBJDIF(*ALL)
```

Note: The line beginning with RST and ending with ALWOBJDIF(*ALL) should be one line on your command line.

One or more messages may appear if config.properties is not on your system.

10. To restore custom built web pages in the Host On-Demand publish directory, enter the following command:

```
RST DEV('/qsys.lib/qgpl.lib/hod.file ')
OBJ(('/qibm/proddata/hostondemand/hod/*.html ')
OUTPUT(*PRINT)
```

Note: By not specifying ALWOBJDIF(*YES), this step avoids replacing *.html objects that are part of Host On-Demand

Configuring the iSeries HTTP server

The following commands assume that you are using the IBM HTTP server's DEFAULT HTTP configuration and CONFIG HTTP instance. These adjustments are necessary to grant the HTTP server permission to serve objects from the /qibm/proddata/hostondemand/hod directory. For more information, refer to the iSeries Webmaster's Guide at the following site:

<http://publib.boulder.ibm.com/html/as400/v5r1/ic2924/info/rzahl/rzahlusergoal.htm>

For additional information, refer to the *iSeries Webmaster's Guide* at:

<http://as400bks.rochester.ibm.com>

1. Stop the Web server using the following command:

```
ENDTCPSVR *HTTP HTTPSVR(DEFAULT)
```

2. Configure the Web server using the following command:

```
WRKHTTPCFG
```

3. Make sure that active Enable POST and Enable GET entries exist and are not commented out. Add the following entry (there must be one space before the first slash (/) and after the first asterisk (*)):

```
pass /hod/*/QIBM/ProdData/hostondemand/HOD/*
```

This entry creates an alias, hod, for the path to the Host On-Demand files. It is case sensitive, so you must type it exactly as you typed the original directory names.

4. Press F3 to exit the WRKHTTPCFG tool.
5. Start the Web server using the following command:

```
STRTCPSVR *HTTP HTTPSVR(DEFAULT)
```

6. If you want the Host On-Demand Service Manager to automatically start after an IPL (when QSYSWRK is started), type the following command:

```
CHGHTTPA AUTOSTART(*YES)
```

7. Load `http://server_name/hod_alias/hodmain.html` (where *server_name* is the name of your server and *hod_alias* is the directory you set in step 3. on page 56) to verify that the Web server can serve Host On-Demand HTML pages.

For advanced configuration information, see Configuring on iSeries in Appendix C of the online *Host On-Demand Getting Started* manual. You can find this manual on the installation CD.

2.4.7 Installing on OS/390 or z/OS

For instructions about installing Host On-Demand on OS/390 or z/OS, refer to 3.2, “Host On-Demand installation” on page 83.

2.4.8 Changing the Configuration Server port

The Host On-Demand Service Manager provides support for persistent user configuration, error logging, and the Redirector.

During Host On-Demand installation, you have the option of specifying which port the Configuration Server will use to communicate with clients. The default port is 8999.

Changes to Configuration Server

You change the Configuration Server port after Host On-Demand is installed by updating the NSMprop file. If the Configuration Server is running on z/OS, see 3.3.6, “Changing the configuration port” on page 90 for details. On all other platforms, follow the instructions below to make changes to the Configuration Server:

1. Stop the Configuration Server.

- On Windows System it runs at a service - stop it using the Services panel
- On OS/2 press **CTL-C** in the window from where it was started
- On Novell enter **java-exit** from console
- On a Unix machine use the following sequence:

Determine the process ID of the Service Manager by entering the following command:

```
ps -ef | grep NCServiceManager
```

The system responds with a line similar to the following:

```
root 20130 22944 0 Feb 16 pts/1 0:20 java
com.ibm.eNetwork.HODUtil.services.admin.NCServiceManager
/usr/local/hostondemand
```

The number following root is the process ID (20130 in the example above).

Enter **kill -9 20130** at the command prompt.

2. Change ConfigServerPort parameter.

This can be done in one of several ways. Choose the most convenient method. These are listed in order of precedence based on where the parameter is set. These parameters are case-sensitive.

- a. Add the command-line parameter **/ConfigServerPort=8999** to the end of the command you use to start the Configuration Server (or edit the script that is used to start the Configuration Server so that the ConfigServerPort parameter is passed to it).

On the iSeries, use CFGHODSM to add the ConfigServerPort parameter.

On Windows machines the Service Manager runs as a service so that this parameter could be entered in the registry or in the properties panel for the service - but we do not recommend that - please refer to option b) and c)

- b. Edit the NSMprop file to add **ConfigServerPort=8999** to the bottom of the file. The NSMprop file is found in the private directory of the Host On-Demand server. For example:

- On Windows: **D:\hostondemand\private**

- On z/OS: /usr/lpp/HOD/hostondemand/HOD/private
- c. Edit the NSMprop file to update CONFIGSERVER_PARMS:
CONFIGSERVER_PARMS = %INSTALL_PATH% 8999
This method provides compatibility with previous versions of Host On-Demand.
- 3. Restart the Configuration Server.

Because there are several ways to specify a different configuration port for the Service Manager, there is a precedence that takes place based on where the parameter is set:

- a. First is a command line parameter, such as /ConfigServerPort=12345.
 - b. Second is the ConfigServerPort entry in the NSMprop file
 - c. Third is the setting of the second parameter of CONFIGSERVER_PARMS in the NSMprop file
4. Next, change the port the Host On-Demand clients use.

Changes to Host On-Demand clients

When you elect to use a port other than the default of 8999, you must notify all clients of the port change.

There are several ways to make the necessary changes to each entity but it is important to note the order of precedence. There must be a match between the port specified for the Configuration Server, the clients, and the Configuration Servlet (if used) in order for the clients to successfully access the Configuration Server. If you have trouble accessing the Configuration Server, verify each possible point of change to make sure one parameter is not canceling out another. These are listed below in the order of precedence based on where the parameter is set.

1. Change the ConfigServerPort parameter in the client HTML. The preferred method is through the Deployment Wizard. See 14.3.2, "Configuration Server-based model example" on page 549 for an example. Only clients that use the customized HTML files will pick up the change. If you need to manually modify the HTML, use the PARAM tag inside the APPLET tag to set the ConfigServerPort parameter. For example:

```
<PARAM_NAME=ConfigServerPort VALUE=8999>
```
2. Change the ConfigServerPort parameter in the config.properties file. This change will be picked up by all clients.

Edit config.properties to add ConfigServerPort=8999 to the file. If a config.properties file does not exist, create it and add the parameter. The config.properties file is found in the publish directory of the Host On-Demand server. For example:

- On Windows: D:\hostondemand\HOD
- On z/OS: /usr/lpp/HOD/hostondemand/HOD

Example of Windows config.properties file:

```
#
#
# This file is used to set properties that apply to all Host On-Demand
# applets loaded from this directory. By default, config.properties is
# read by each applet and used to over-ride default values.
# Parameters set using HTML PARAM tags will take precedence over those
# set in this file.
# Examples:
# =====
# ConfigServer=hodserver.ibm.com
# ConfigServerPort=12345
#
# For Websphere Application Server 3.5x
# ConfigServerURL=http://hodserver.ibm.com/servlet/HODConfig/hod
# For Websphere Application Server 4.x
# ConfigServerURL=http://hodserver.ibm.com/HODConfig/HODConfig/hod
#
ConfigServerPort=8999
```

Force any active Host On-Demand clients to re-read the config.properties file by clearing your browser's cache and reloading the Host On-Demand applet.

Because there are several ways to specify a different configuration port for the clients, there is a precedence that takes place based on where the parameter is set:

- a. First is the ConfigServerPort parameter set in the client HTML.
- b. Second is the ConfigServerPort set in the config.properties file.

If you are using the configuration servlet and you change the Service Manager port, you will need to set the ConfigServerPort parameter for the configuration servlet. For example, if you change the Service Manager port to 12345 you need to pass the ConfigServerPort=12345 parameter to the configuration servlet so it can communicate with the Service Manager. Check your web server or servlet engine documentation for information about how to pass parameters to servlets.

2.4.9 Installing the Configuration Servlet

During the Host On-Demand installation, you can choose to have the configuration servlet installed and configured on the Windows NT, Windows 2000, and AIX platforms for recognized Web application servers. Recognized Web application servers include:

- ▶ IBM WebSphere Application Server Version 3.5 and 4.0
- ▶ Lotus Domino Go Web Server
- ▶ IBM Domino Go Web Server

All Web servers and servlet engines are configured differently. Check your Web server and servlet engine documentation for servlet configuration details on your operating system

Installing on Windows and AIX platforms

The following instructions assume a Web server is already installed. To manually install the configuration servlet:

1. Install Host On-Demand, without running the configuration servlet installation, to a directory such as `d:\hostondemand` or `/usr/opt/hostondemand`.
2. Add `cfgsrvlt.jar` from the Host On-Demand installation's `lib` directory to the servlet engine's classpath; for example `d:\hostondemand\lib\cfgsrvlt.jar` or `/usr/opt/hostondemand/lib/cfgsrvlt.jar`. Refer to your Web server or servlet engine documentation for information about how to do this. You can get a copy of `cfgsrvlt.jar` from the `/servlet` directory of the Host On-Demand CD or from the `install_dir/hostondemand/lib` directory on your server.
3. Add a servlet definition named `hodconfig` with a class name of `com.ibm.eNetwork.HODUtil.services.remote.HODCfgServlet`. Refer to your Web server or servlet engine documentation for information about how to add a servlet definition.
4. Configure the configuration servlet. If necessary, set the `ConfigServer` and `ConfigServerPort` parameters to the host name and port number of the Host On-Demand Service Manager. Refer to your Web server or servlet engine documentation for information about how to pass parameters to a servlet.

The port used by the clients, configuration servlet, and the Service Manager can be customized. For instructions on how to customize the port, see the topics [Configuring the configuration servlet](#) and [Changing the Service Manager port](#) in the online help.

Host On-Demand clients use the default port of 8999 to communicate with the Service Manager for configuration information. If any of your clients are outside the firewall, the firewall administrator must open this port internally and externally. Optionally, you can customize the clients to access the

configuration servlet through a firewall over either HTTP or HTTPS. The configuration servlet then communicates with the Service Manager on port 8999. If both the configuration servlet and the Service Manager are inside the firewall, port 8999 does not need to be opened for Host On-Demand.

5. Publish the `hodconfig` servlet using an alias. Refer to your Web server or servlet engine documentation for information on about how to make the configuration servlet known to the Web server. In general, you are associating the fully qualified name of the servlet with an alias.
6. Stop and restart the Web server and the servlet engine, or refer to your Web server or servlet engine documentation for information about saving the changes.

Once the configuration servlet is installed, you must configure your clients to use the configuration servlet instead of directly accessing the Service Manager. You can use the Deployment Wizard to build customized HTML client pages. The wizard sets the applet parameters in the HTML based on your input, so you don't have to learn the syntax and valid parameter values. We recommend that you use the Deployment Wizard to set the `ConfigServerURL` parameter in the client HTML to the name you assigned to the servlet through the servlet engine in the publish step above. For example, if you set the name of the servlet to be `/servlet/hodconfig`, set the Configuration Server URL to `/servlet/hodconfig/hod`

Note: The servlet alias and the value for the `ConfigServerURL` parameter are different.

If you find you need to manually modify the HTML, use the `<param tag>` inside the `<applet>` tag to set the `ConfigServerURL`. For example, to set the `ConfigServerURL` to `/servlet/hodconfig/hod` set `<param name=ConfigServerURL value=/servlet/hodconfig/hod>` in the `<applet>` tag in the HTML client.

For more information regarding configuration servlet parameters, configuration and examples, see *Configuring the configuration servlet* in the online help.

Installing on the iSeries platform

On the iSeries platform, you must manually run a separate utility in order to install and configure the configuration servlet. This utility is provided by Host On-Demand and supports WebSphere Application Server Version 3.5 and 4.0.

To install and configure the configuration servlet on iSeries, do the following:

1. Start a shell by entering the following at the command prompt:

```
qsh
```

2. Change directories to the Host On-Demand servlet directory:

```
cd /qibm/proddata/hostondemand/lib/samples/hodservlet
```

3. Enter the following command to run the configuration utility:

```
cfghodservlet-os400.sh
```

The installation is complete when the \$ symbol is shown.

4. Press the F3 key to exit the qsh program.
5. Edit the config.properties file by entering the following at the command prompt:

```
edtf '/qibm/proddata/hostondemand/hod/config.properties'
```

6. Add the following line:

```
ConfigServletURL=http://my400/HOD/HODConfig/hod
```

where *my400* is the name of the iSeries system. Be aware that this URL is case sensitive.

7. Press the Enter key, then press the F3 key to update the config.properties file.
8. Stop the Host On-Demand Service Manager by entering the following command at the command prompt:

```
ENDHODSVM
```

9. Restart the Host On-Demand Service Manager by entering the following command at the command prompt:

```
STRTHODSVM
```

10. Verify that the servlet is working. In a Web browser, enter the following URL:

```
http://my400/HOD/HODConfig/info
```

where *my400* is the name of the iSeries system. Be aware that this URL is case sensitive.

2.4.10 Installing the locally installed client

The locally installed client installs to a local disk. The client applet is loaded directly into the default system browser, so there is no download from a server. The most common reason to use a local client is for users who connect remotely over slow telephone lines, where download time can be an issue and connectivity is unpredictable. You can also use the locally installed client to test host access capabilities without installing the full Host On-Demand product.

For Java2 information on Windows XP please see hints on Chapter 2.3, "Client requirements" on page 30

To install the Host On-Demand local client on a Windows NT, Windows 2000, or Windows XP workstation, you must be a member of the Administrators group.

1. Insert the CD and run

`setup.exe /c`

from the \win32 directory of the CD. At the Welcome screen click Next. Follow the next screens.

2. You will reach the Client Installation screen

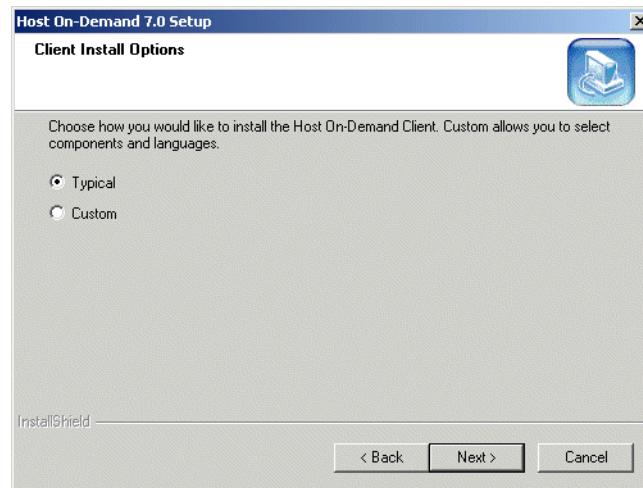


Figure 2-9 HOD Client Installation selection

3. Choose a Typical or Custom installation.
 - a. Typical installs the Host On-Demand Java applets and the information library in English plus the native language of your workstation according to your locale setting of the operating system.
 - b. Custom allows you to choose components to install: Host On-Demand Java applets, the information library and the Host Access Class Library. In addition to English, you can also select any of the other supported languages.
4. Proceed through the rest of the windows.
5. If you have not already done so, read the Readme available in the last window.

At the end of installation, the Host On-Demand Service Manager is configured and started automatically. On Windows NT, Windows 2000, and Windows XP, the Service Manager is installed as a Service; on Windows 95, Windows 98, and Windows Millennium (Me) it is added to the Start menu.

Starting the local client

To start Host On-Demand as a client, click

Start > Programs > IBM Host On-Demand > Host On-Demand

2.4.11 Installing the Deployment Wizard

The Deployment Wizard is automatically installed as part of the Windows Host On-Demand server installation. It is also available separately for those customers who do not wish to install the entire Windows Host On-Demand server. This separate Deployment Wizard can be installed in one of two ways:

- ▶ Using the Deployment Wizard install option on the Windows Host On-Demand server installation CD.
- ▶ Downloading it from the Host On-Demand server.

The following two sections describe the installation process for each, respectively.

Note: The Deployment Wizard installation image is approximately 85 MB. If you are planning to download this installation image, particularly over a modem, prepare for a large download

Installing the Deployment Wizard from the Windows CD

To install and run the Deployment Wizard, do the following:

1. Insert the Host On-Demand CD. If autorun is enabled, the CD Installer starts automatically. If autorun is not enabled, start the CD Installer by running the setupwin.exe file located on the Host On-Demand CD.
2. From the CD Installer window, select Install Deployment Wizard.
3. The InstallShield Wizard will guide you through the remaining installation steps.
4. Once installation is complete, you can launch the Deployment Wizard from the Start > Programs desktop menu.

Downloading the Deployment Wizard installation image from a Host On-Demand server

The Deployment Wizard image is shipped on all Host On-Demand server platforms, and can be downloaded from the server and installed on any Windows machine.

To download the Deployment Wizard from a Host On-Demand server, do the following:

1. From your Windows machine, start your browser and point to the HODMain_xx.html file on your Host On-Demand server, where xx is your two letter language suffix.
You can enter and use only those languages which had been installed on the HOD server according to Figure 2-2 on page 34. The Language suffixes are as shown in Table 2-13 on page 65.

Table 2-13 Language suffixes:

Language	Language suffix
Simplified Chinese	zh
Traditional Chinese	zh_TW
Czech	cs
Danish	da
Dutch	nl
English	en
Finnish	fi
French	fr
German	de
Greek	el
Hungarian	hu
Italian	it
Japanese	ja
Korean	ko
Norwegian	no
Polish	pl
Brazilian Portuguese	pt

Language	Language suffix
Portuguese	pt_PT
Russian	ru
Slovenian	sl
Spanish	es
Swedish	sv
Turkish	tr

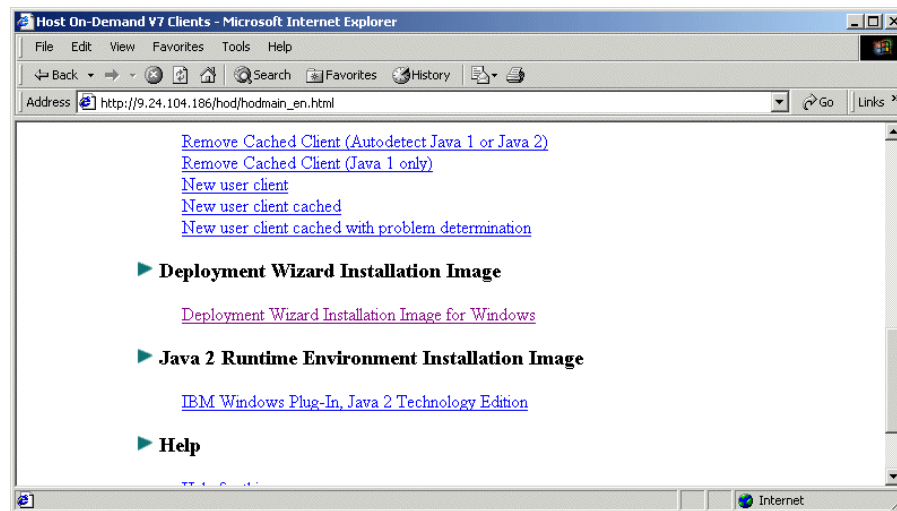


Figure 2-10 Downloading the Deployment Wizard

2. Click on the Deployment Wizard link. This will download the Deployment Wizard installation image to your Windows machine.
3. Run the Deployment Wizard installation from your Windows machine.
4. Once installation is complete, you can launch the Deployment Wizard from the Start > Programs desktop menu.

2.5 Migration considerations

With very few exceptions, data used in earlier versions of Host On-Demand will be automatically migrated when you begin using Host On-Demand Version 7.

You can upgrade the Host On-Demand server so that the upgrade is transparent to the clients. After the upgrade, the clients have their same sessions defined and all their customizations, for example, macros and keyboard remaps, continue to work as before (first introduced as Enhanced Local Preferences with HOD 6). On platforms where an uninstall program for Host On-Demand is not provided, the administrator must back up some files and directories before upgrading, and then restore them after the upgrade.

2.5.1 Server considerations

Any existing configuration data in your Configuration Server will be automatically available in Host On-Demand Version 7 once installation is complete. Even if most of the installation processes preserve the data in the `\private` directory, it is recommended that you back up all custom HTML files and the `\HODData` directory.

Note: The method for backing up files might vary depending on what server platform you use.

If you need to make changes to the `NSMprop` file (for example, to change the default port), or need to migrate `NSMprop` from a previous version of Host On-Demand, put this file in the `/private` directory.

Backing up files and directories

You can put custom HTML files (files generated from the Deployment Wizard), `config.properties`, and `CustomizedCAs.class` files in a directory other than the Host On-Demand publish directory. Creating a user publish directory makes it easier to apply future Host On-Demand upgrades because installing a new version of Host On-Demand will not affect the new directory. It also keeps the Host On-Demand publish directory read-only and provides a separate writable location for deploying Deployment Wizard pages. Additionally, creating a separate user publish directory isolates new files from those provided by Host On-Demand. Note that other user-modified files (such as customer applets and HACL programs) still need to run from the Host On-Demand publish directory.

To set up a separate user publish directory, do the following:

1. Specify the codebase (the URL of your Host On-Demand publish directory) as follows:
 - a. Using the Deployment Wizard, on the Additional Options page, click the Advanced Options button.
 - b. Select the Other tab.

- c. Enter the codebase. You can enter a fully qualified URL including the hostname (for example, `http://your_HOD_server/hod_publish_dir_alias/`) or a relative path (for example, `/hod_publish_dir_alias/`).

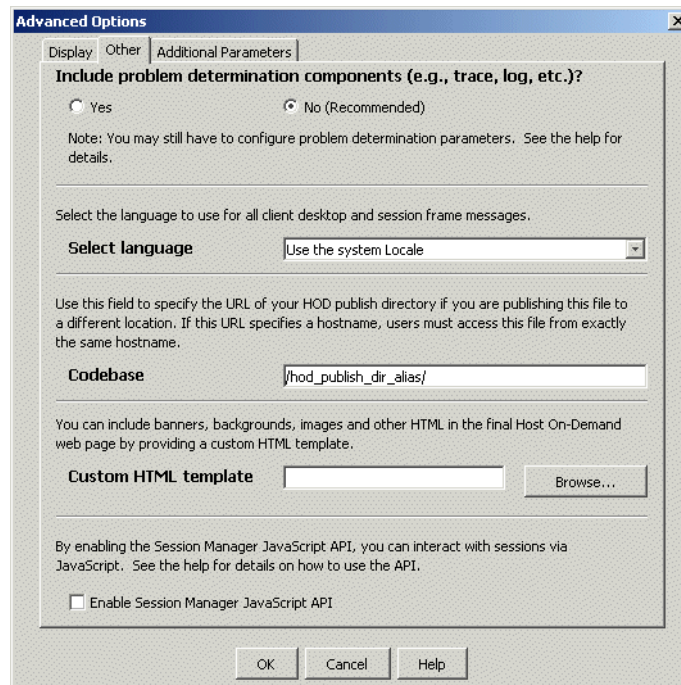


Figure 2-11 Advanced Options of Deployment Wizard

2. Select Output Zip to save the files generated from the Deployment Wizard in a Zip file.

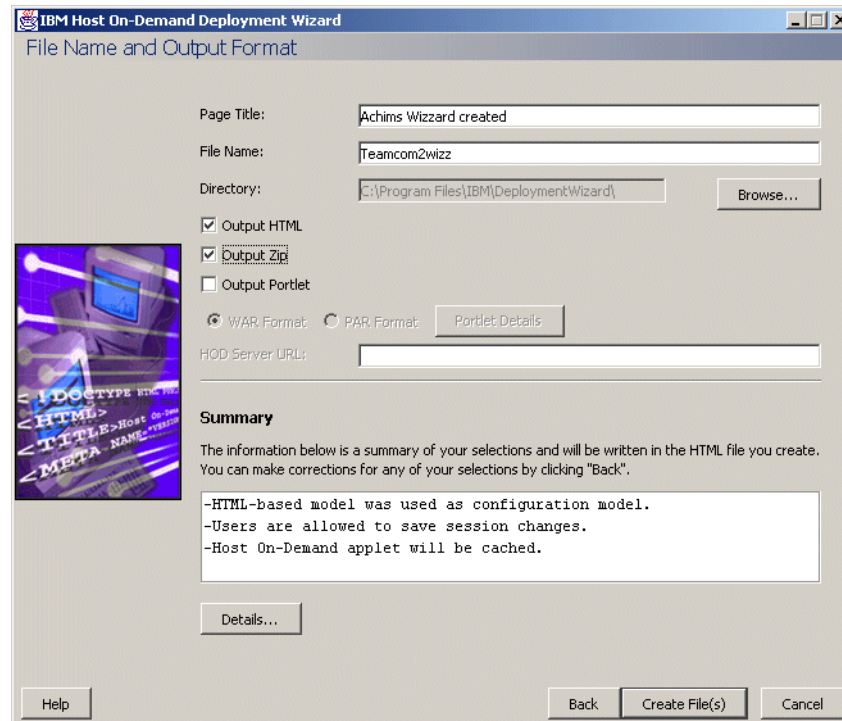


Figure 2-12 Name and Output Format Panel

3. Click Create Files.
4. If you are not running the Deployment Wizard on your Host On-Demand server, FTP the output Zip file to your server platform.
5. Create a separate user publish directory, /user_publish_dir/.
6. Use the DWunzip tool to install the Deployment Wizard generated files into the /user_publish_dir/ directory. You must edit the Dwunzip command file on your server to specify the correct MY_PUBLISHED_DIRECTORY value. You will find the sample DWunzip.cmd on Windows machines in \hostondemand\lib\samples\DWunzip. See the online help topic Using Dwunzip for more information on how to use this tool. Example 2-3 on page 69 shows a simple customized section of DWunzip.cmd.

Example 2-3 Example of customized DWunzip.cmd

```

REM #####
REM   If you do not use Host On-Demand's default web-published directory, then
REM   set the following variable to be your web-published directory.
REM   Note: This is also the directory where your zip file should be.
REM   Example: set MY_PUBLISHED_DIRECTORY=c:\hostondemand\HOD

```

```

set MY_PUBLISHED_DIRECTORY=c:\hostondemand\HOD

REM #####
REM   If you run DWUnzip from anywhere other than the directory where
REM   Host On-Demand was installed, then set the following variable
REM   to your Host On-Demand installation directory.
REM   EXAMPLE:  set MY_HOD_DIRECTORY=c:\hostondemand

set MY_HOD_DIRECTORY=c:\hostondemand

```

The execution of the DWunzip results in:

Example 2-4 Execution of DWunzip

```

C:\Program Files\IBM\DeploymentWizard>dwunzip teamcom2wizz

Extracting teamcom2wizz.zip to and from directory: c:\hostondemand\HOD
Allocating ZIP comments array
Added ZIP comment "HODCDLABEL"
Added ZIP comment "HODCDLABEL"
Added ZIP comment "HODCDLABEL"
Added ZIP comment "HODCDLABEL"
Added ZIP comment "HODCDLABEL"
Added ZIP comment "HODCDLABEL"
File being extracted from teamcom2wizz.zip: Teamcom2wizz.html
File being extracted from teamcom2wizz.zip: HODData\Teamcom2wizz\cfg0.cf
File being extracted from teamcom2wizz.zip: HODData\Teamcom2wizz\params.txt
File being extracted from teamcom2wizz.zip: HODData\Teamcom2wizz\policy.obj
File being extracted from teamcom2wizz.zip: HODData\Teamcom2wizz\preloads.obj
File being extracted from teamcom2wizz.zip: HODData\Teamcom2wizz\wInfo.txt

File extraction was a success.

```

The Deployment Wizard HTML files are installed in the directory /user_publish_dir/. Additional files like cfg0.cf, params.txt, and so forth, are installed in the /user_publish_dir/HODData/your_html directory as shown in Example 2-4 on page 70.

7. Add a pass rule (also known as an alias on some platforms) in your Web server configuration file, /etc/httpd.conf, to point to this new user publish directory. For example:

```
Pass /user_alias/ * /user_publish_dir/ *
```

8. If changes are required in the Host On-Demand config.properties file (for example, to change the default port or enable the Host On-Demand configuration servlet), do the following:

- a. Update the config.properties file. If your server platform does not support the ASCII character set, update this file on a machine that does support ASCII.
- b. If the config.properties file was updated on a different platform than your server, FTP the file to your server platform in binary format.
- c. Place the file in the user publish directory, /user_publish_dir/.
- d. Add the following pass rule (also known as an alias on some platforms) in the Web server configuration file /etc/httpd.conf:

```
/hod_publish_dir_alias/config.properties
```

```
/user_publish_dir/config.properties
```

Note: On the zSeries platform, append the ascii extension, /user_publish_dir/config.properties.ascii.

9. If you are using SSL and need to change the CustomizedCAs.class file, do the following:

Place the updated file in the user publish directory
/user_publish_dir/CustomizedCAs.class.

- e. Add the following pass rule (also known as an alias on some platforms) in the Web server configuration file /etc/httpd.conf:

```
/hod_publish_dir_alias/CustomizedCAs.class  
/user_publish_dir/CustomizedCAs.class
```

10. Restart the Web server.

11. From a Web browser, specify the URL:

```
http://your_HOD_server/user_alias/your_html.html.
```

Migrating on server platforms with an uninstall program

On server platforms that have an uninstall program, for example, Windows and AIX, the uninstall program assists in the upgrade process. The uninstall program does not uninstall any files that the installation program did not install initially, for example, CustomizedCAs.class or customized HTML files. Also, there are no changes to the private directory during the uninstall of the previous release. Any customized files that you added for the previous release of Host On-Demand remain unchanged when you install Host On-Demand 7. Run the uninstall program to remove the old version and then install Host On-Demand 7.

Migrating on server operating systems without an uninstall program

On server platforms without an uninstall program, you should delete the Host On-Demand 6 installation directory. Before you delete the installation directory, copy the private directory, any files added to the publish directory, such as CustomizedCAs.class or customized HTML files, and the HODData directory to a temporary location. After you install Host On-Demand 7, move these files and directories back to their original location.

Moving a Host On-Demand server installation to a new server

You can move your Host On-Demand server configuration from one server to another. If you install Host On-Demand in a test environment before deploying to your production environment, complete the following steps to migrate Host On-Demand from one server to another (or from one HFS to a different HFS in an OS/390 or z/OS environment). For example, if you wanted to move your Host On-Demand server configuration from server1 to server2, you must:

1. Install Host On-Demand on server2. You must install the same release level of Host On-Demand on server2 that is installed on server1. If server2 is using a web server that is not recognized by the Host On-Demand installation program, you must manually configure the web server for Host On-Demand. Refer to Planning, Installing, and Configuring Host On-Demand for more information.
2. Stop the Host On-Demand Service Manager on server2.
3. Replace the necessary files and directories on server2 with those from server1. This step overwrites any configuration changes made to Host On-Demand on server2 since step 1.
4. Copy the \private\ directory in the Host On-Demand root directory, which is \hostondemand\ by default, on server1 to the \private\ directory on server2 to move user and group configuration information to server2.
5. Copy all files and directories you have created with the Deployment Wizard from their server1 location to server2. For example, if you created test.HTML using the Deployment Wizard, copy test.HTML and Autotest.HTML in the publish directory from server1 to server2. Also copy the test directory in \HODData in the publish directory from server1 to server2.
6. Start the Host On-Demand Service Manager on server2

Note: If your current environment is not OS/390 or z/OS and you want to move to an OS/390 or z/OS environment, this migration requires some additional steps. You can copy the private directory and CustomizeCAs.class file over to the new server directly. However, you should use the DWUnzip utility to correctly install the customized HTML files and the HODData directory.

2.5.2 Client considerations

There is no need to actually migrate a client, except the locally installed client. For the locally installed client the InstallShield for Windows XP, Windows 2000, Windows NT, Windows 98 and Windows 95 will detect any earlier version of Host On-Demand, uninstall it, and install Host On-Demand Version 7.

When using a cached client, the user will be notified about the change at the server when accessing it for the first time and any new files will be downloaded.

Upgrading Host On-Demand Version 4.x cached clients to Host On-Demand 7

If you upgrade your Host On-Demand server from Version 4.x to Version 7, your clients will no longer be able to communicate with the server without upgrading.

If you need to manage network demand while upgrading cached clients, you can gradually move all of your Host On-Demand Version 4.x cached clients to Host On-Demand 7 by setting up two servers. One would be a Host On-Demand Version 4.x server and the other would be a Host On-Demand 7 server. Configure all clients to access the Host On-Demand 7 server, and then add the HTML parameter HODServer to HODCached.html, or any of your customized cached client HTML files that are on the Host On-Demand 7 server. There are two sets of applet parameters defined in the HTML. Add the HODServer parameter to the set defined by the array cHod_AppletParams. You can do all of this using the Deployment Wizard on the Additional Parameters window; however, if you want to manually modify the HTML, the format for the parameter is:

```
cHod_AppletParams[7] =<PARAM NAME=HODServer  
VALUE=http://yourhostname/alias/HODCached.html>
```

(this is actually contained in one single line)

where *yourhostname* and *alias* are your Host On-Demand Version 4.x server's hostname and alias, or Publish, directory. Make sure that the index of the new cHod_AppletParams array element is in the correct sequence with the existing array elements.

The *HODServer* parameter works with the *UpgradePercent* and *UpgradeURL* parameters to manage client upgrades. If the cached client won't be upgraded on this connection attempt, it is redirected automatically to the Host On-Demand Version 4.x server specified in the *HODServerHTML* parameter. If a cached client will be upgraded, the Host On-Demand Version 4.x cached client is removed and the Host On-Demand 7 cached client is installed. Once the client is upgraded to Host On-Demand 7, the *HTML* parameter is ignored and the client is no longer redirected to the Host On-Demand Version 4.x server. After you have gradually upgraded all your cached clients, you no longer need the Host On-Demand Version 4.x server.

Note: Be aware of the following when you are upgrading cached clients from Version 4.x to Version 7:

- ▶ Cached clients are upgraded in the foreground. The upgrade in background option is ignored.
- ▶ If you have customized Host On-Demand Version 4.x *HODCached.html* and have called it something different, like *OurHTML.html*, do the following:
 - a. Copy the Host On-Demand 7 version of *HODCached.html* to the file *OurHTML.html*;
 - b. Add the *HODServer* parameter to *OurHTML.html*. The *HODServer* parameter should specify `http://yourhostname/alias/OurHTML.html` as the Host On-Demand Version 4.x server.
- ▶ You can copy the new *HODCached.html*, that includes the *HODServer* parameter, to *AutoHODCached.html* and *AutoHODLaunch.html*, in case these pages are bookmarked by the clients. The *HODServer* parameter in *AutoHODCached.html* should specify the *AutoHODCached.html* page on the Host On-Demand Version 4.x server. The *HODServer* parameter in *AutoHODLaunch.html* should specify the *AutoHODLaunch.html* page on the Host On-Demand Version 4.x server
- ▶ If you are using language specific HTML files (such as *HODCached_es.html*, *AutoHODCached_es.html*, *AutoHODLaunch_es.html*, etc.) you can also add the *HODServer* to these pages.

Upgrading custom HTML files

Java 1

If your users have Java 1 browsers and you have customized HTML files from previous versions of the Deployment Wizard, you do not have to regenerate the custom files with the Host On-Demand 7 Deployment Wizard. The users can take advantage of all the new features (except Java 2-specific features) once the client code gets upgraded to Host On-Demand 7.

Java 2

If your users have Java 2 browsers, we strongly encourage you to regenerate the HTML files with the Host On-Demand 7 Deployment Wizard to receive the improved support for Java 2 environments. Additionally, if you want to take advantage of the new features built in to the Host On-Demand 7 Deployment Wizard, such as the customized template, separate codebase, or upgrade based on time of day, you should regenerate your custom HTML files.

2.5.3 Client Migration Problems

It is not common, but does happen that when a user upgrades a workstation from one HOD client version to another that it does not go well. For these situations you should be prepared to try the following procedures:

1. Have user use the HodRemove.html utility to remove the existing cache client
2. Clear browser temporary cache
3. Clear Java 2 cache
4. Reload the new cache client

If this still fails to resolve the problem then use the HOD InfoCenter and refer directly to the Troubleshooting section of the Online Help. The section *client troubleshooting checklist* under the list of *Troubleshooting topics* should be very helpful.

2.6 Removing Host On-Demand

To uninstall the Host On-Demand server follow the appropriate steps for your platform:

2.6.1 zSeries

Follow the instructions in the Program Directory for uninstalling the Host On-Demand server on zSeries.

2.6.2 iSeries

You will need *JOBCTL, *SPLCTL, *SERVICE and *ALLOBJ authority to use this command. Logon to the iSeries with a security officer user profile, such as QSECOFR.

1. Shutdown the Service Manager by typing ENDHODSVM at the command line.

2. Delete the licensed Host On-Demand product by typing DLTLICPGM LICPGM(5733A59) at the command line.
3. Remove any directories containing user data manually after the program has completed. You will also need to remove the QUSRSYS/QHODCFGD *DTAARA object.

2.6.3 Windows NT, Windows 2000, or Windows XP

Use Add/Remove Programs from the Windows control panel.

Notes:

- ▶ On Windows 2000, if you plan to reinstall Host On-Demand, you should reboot first.
- ▶ If you install the standalone Deployment Wizard on a Windows NT or Windows 2000 workstation that already has Host On-Demand server installed, you should uninstall the Deployment Wizard before you uninstall the Host On-Demand server.

If you uninstall Host On-Demand server first, you might not be able to uninstall the Deployment Wizard because the Deployment Wizard uninstallation attempts to use the Host On-Demand JVM.

2.6.4 AIX, Solaris, Linux, HP-UX

Stop the Host On-Demand Service Manager. Get the process ID, kill the process, then delete the Host On-Demand directories (except ./private).

2.6.5 OS/2

Stop the Host On-Demand Service Manager by pressing Ctrl+C in the OS/2 window in which you started it, close the window, then delete the Host On-Demand directories (except \private).

2.6.6 Novell NetWare

From the console, enter java -exit to stop the Java NLM, then delete the Host On-Demand directories (except \private).

2.7 Service updates

You can access the latest service updates from the Host On-Demand Support Web site. The URL is:

<http://www.ibm.com/software/webservers/hostondemand/support.html>

Click **Service Updates**.

To download the latest service updates for Host On-Demand, Screen Customizer, and Personal Communications, you will need to register with the IBM Software Internet Service Delivery site. The URL is:

<http://www6.software.ibm.com/enetwork/isd/home.html>

This site entitles you to download service updates directly from the Internet. You must first register, then add your service key to your registration. The service key is the 10-digit number on the service key card that is provided for each product. Then you can download the product that matches the key you have entered. We recommend registering every key that comes in the package.

The service key must be treated with the same care given to the base product in terms of export and import regulations. It provides access through the Web to product code that contains encryption technologies. Care should be taken to read and comply with the text presented on the Authorization to Download Web page. You must agree to these terms prior to being allowed to download the product code.

Note: The service key can be used by only one person. Once registered to an individual, it cannot be registered to another individual.

When attempting to download Corrective Service Distributions (CSDs) or Program Temporary Fixes (PTFs) from the site, you must make sure that you select the product version, the language and the encryption correctly.

Please also note that this key will provide you with access to the product as long as service is generally provided for this level of code.

3

z/OS implementation

In this chapter we discuss Host On-Demand on the z/OS platform. The base operating system referred to in this chapter is assumed to be one of the following: OS/390 V2R9 or higher; z/OS V1.1 or higher. Information relevant to a particular operating system is listed. Information pertinent to both OS/390 and z/OS is referred to as z/OS. Although the z/OS environment is a UNIX environment, it is installed and maintained differently from a normal UNIX distributed environment. Therefore, this chapter will cover the following areas for z/OS, emphasizing the z/OS unique aspects.

- ▶ Planning
- ▶ Host On-Demand installation
- ▶ Activating Host On-Demand Service Manager, including the configuration of the Web server and RACF
- ▶ Deployment Wizard considerations for uploading customized HTML pages, including a sample created and uploaded to a z/OS server
- ▶ Configuration Servlet setup
- ▶ Using SSL with Communications Server for z/OS, including general information about SSL on z/OS, samples of using the gskkyman certificate management utility, and RACF certificate management, TCP/IP profile for server and client authentication
- ▶ Express Logon Feature (ELF)
- ▶ Native Authentication
- ▶ LDAP directory server configuration on z/OS

3.1 Planning

Host On-Demand consists of only one FMID: HHOJ700.

The distribution medium for Host On-Demand Version 7 is magnetic tape or downloadable files. A program directory is supplied with the package, providing the information necessary to install Host On-Demand using SMP/E, activate Host On-Demand server, start the Native Authentication service, and set up the LDAP directory server.

Installing Host On-Demand by SMP/E will result in the previous Host On-Demand FMID being deleted, for example:

- ▶ HHOF600 - Version 6 FMID
- ▶ HHOH500 - Version 5 FMID

Host On-Demand on z/OS can only be installed using SMP/E. You cannot copy the .tar file from another platform and run hod70mvs.sh shell script to install it.

It is recommended that you consult the program directory and the following Web sites prior to installation. Support, product information, and hints and tips can be found on the following Web sites:

- ▶ Product information site:
<http://www.ibm.com/software/webservers/hostondemand/>
- ▶ Support site (hints and tips, service updates, newsletters):
<http://www.ibm.com/software/webservers/hostondemand/support/>
- ▶ Program directory softcopy:
<http://www.ibm.com/software/webservers/hostondemand/library/>

Maintenance for Host On-Demand V7 can be ordered in one of two ways, both of which are SMP/E installed:

- ▶ Go to the support Web site and select **Support Downloads**. You must be registered with the IBM Software Internet Service Delivery site. Refer to 2.7, “Service updates” on page 77 for additional details.
- ▶ Order the PTF tapes via IBM support.

3.1.1 Software requirements

The following requisites are required for the Host On-Demand V7 product to install or function:

- ▶ OS/390 V2R9 or higher or z/OS V1R1 and above.

- If running with OS/390 V2R10 and using the LDAP directory server, maintenance (APAR OW45791/PTF UW73147) is needed to properly display LDAP sessions.
- The z/OS Communications Server TCP/IP Services included with z/OS are required at runtime.
- Java for z/OS, required at runtime
- Java V1R3 or higher
- Java V1R1M8
- If running with OS/390 V2R10, you must install Java APAR OW45575/PTF UW78944(PTF6)
- ▶ A Web server
 - IBM HTTP Server V5.2 or higher
- ▶ WebSphere Application Server if using the Configuration Servlet
 - WebSphere Application Server for OS/390 V3.5
 - Websphere Application Server for z/OS and OS/390 V4.0.1
- ▶ For SSL encryption, one of the following elements is needed. Refer to Table 3-1 on page 108 for the correct FMID:
 - Communications Server for OS/390: IP Security SSL DES (56-bit Export)
 - Communications Server for OS/390: IP Security SSL Triple-DES (168-bit US)
- ▶ LDAP directory server (optional)
 - IBM LDAP directory server for z/OS

3.1.2 DASD storage requirements

DASD storage is required for the target and distribution libraries and for the HFS (Hierarchical File System). Work space is also needed during the SMP/E installation. The program directory outlines the storage requirements for Host On-Demand and for SMP/E.

Space requirements for the distribution library and HFS have increased from previous versions of Host On-Demand. The recommended allocation is 940 MB for the distribution library and 3500 MB for the HFS.

3.1.3 Backing up the private directory

The private directory can be backed up using either the **pax** command or the **tar** command. During our upgrade, we used the **pax** command. The private directory was on a different system running Host On-Demand V6.

1. From the Host On-Demand V6 HFS, change directory into the private directory.

```
cd /usr/lpp/HOD/hostondemand/private
```

2. Archive the private directory in a /tmp directory. The -z option compressed the file; the -v provided a list of files and subdirectories being archived (optional).

```
pax -wzvf /tmp/private.pax.Z *
```

3. The private.tar.Z file was then transferred in binary to the /tmp directory on the system for Host On-Demand V7.

4. On the Host On-Demand V7 HFS, change directory into the private directory where the file will be extracted.

```
cd /usr/lpp/HOD/hostondemand/private
```

5. Issue the **pax** command to extract the private.pax.Z file. The -z option specifies a compressed file; the -v provides a list of files and subdirectories being extracted (optional).

```
pax -rzvf /tmp/private.pax.Z
```

3.1.4 Upgrade considerations

When upgrading from a previous level of Host On-Demand, you will probably want to take into consideration previous customizations. The following are three processes of allocating an HFS and restoring the previous private directory. We found the third option to be the easiest.

1. Allocate a new HFS and copy the existing HFS into the new HFS. Then follow the installation procedure. The customization in the private directory will be intact. Any customizations made in any directory other than the private directory will be overwritten, so they must be backed up prior to running the hod70mvs.sh shell script.
2. Install into the existing HFS. With this process, you need to take into consideration space available in the HFS. Host On-Demand V7 will require more space. Any customization other than what is in the private directory will be lost as in option 1.
3. Allocate a new HFS, then follow the installation procedure. Copy your existing private directory into the new HFS using the **pax** or **tar** command. Refer to 3.1.3, "Backing up the private directory" on page 82.

In Host On-Demand V7 the NSMprop file is now located in the /private directory. If you have customized this file in a previous version you can copy the file from the /lib directory to the Host On-Demand V7 /private directory.

Customized files not in the private directory can also be copied to the new HFS, for example, CustomizedCAs.class, custom HTML pages, /HODData directory and config.properties.ascii found in the publish directory.

Publish Directory

With Host On-Demand V7, the administrator can choose to publish files created by the Deployment Wizard to a directory other than the Host On-Demand publish directory. If you wish to move your custom HTML pages to a separate user publish directory you will need to re-edit these files via the Deployment Wizard to add the Codebase parameter. See Chapter 14, “Deployment Wizard” on page 529 for more information on using the Deployment Wizard.

Mounting a separate user publish directory allows the administrator to mount the Host On-Demand HFS as read only. However, when starting the ServiceManager for the first time Host On-Demand will require write access to the /lib directory. Also, if you use the DWunzip utility you will need to either edit DWunzip-S390 prior to mounting the HFS as read only or copy the file to a directory with write permissions. After initialization of the ServiceManager for the first time, the Host On-Demand HFS can be changed from read/ write to read only mode. For more information on setting up a separate user publish directory see Chapter 8, *IBM WebSphere Host On-Demand Version 7.0 Planning, Installing, and Configuring Host On-Demand*, SC31-6301-00.

3.2 Host On-Demand installation

In this section we detail the installation of Host On-Demand on z/OS. We discuss the installation jobs and instructions, activating and stopping the Service Manager.

3.2.1 Installation jobs

Host On-Demand can be installed into its own SMP/E environment, but sample jobs to create and initialize the environment are not provided. Sample jobs are provided to do the basic RECEIVE, APPLY, and ACCEPT functions, as well as defining the DDDEF entries.

The program directory provides JCL that can be used to copy the sample jobs from the product tape. Once the RECEIVE is completed, the samples can be found in the IBM.HHOJ700.F1 data set.

The sample jobs provided to install Host On-Demand V7 are:

HOMRECVE	Sample RECEIVE job
HOMALLOC	Sample job to allocate target and distribution libraries
HOMDDCLN	Sample job for deleting V2 DDDEFs (only for V2 migrations)
HOMDDDEF	Sample job to define SMP/E DDDEFs
HOMCOPY	Sample job to copy V2 to the current version (only for V2 migrations)
HOMHFS	Sample job to define Host On-Demand HFS data set (optional)
HOMISMKD	Sample job to invoke the supplied HOMMKDIR EXEC to allocate HFS paths
HOMAPPLY	Sample APPLY job
HOMACCPY	Sample ACCEPT job
HOMSEVR	Sample job for starting Host On-Demand

The sample jobs should be updated to reflect the CSI, target zone and distribution zone names used in the installation.

3.2.2 Installation instructions

The program directory contains the steps for the SMP/E installation; therefore they are not included in this book. The support Web site contains the latest program directory.

If upgrading from a previous level of Host On-Demand, you need to decide which process to follow to migrate your customization. Refer back to 3.1.4, “Upgrade considerations” on page 82. If allocating a new HFS, you may want to consider increasing the space allocation in the HOMHFS job to accommodate future service updates. Refer to 3.1.2, “DASD storage requirements” on page 81.

Create the mount point and make sure it has permissions of 755. For example:

```
TSO MKDIR '[PATHPREFIX]/usr/lpp/HOD' MODE (7,5,5)
```

where [PATHPREFIX] is the appropriate high-level directory name. For users installing in the default path, this would be null. For others, the high-level directory may be something like /service/ or some meaningful name for your installation.

Note: In the UNIX System Services environment, everything is case-sensitive.

Mount the HFS to the system; it must be mounted with read and write access. This is the default if omitted on the MOUNT command. The command should be on one line.

```
TSO MOUNT FILESYSTEM('hfsprfx.hom.hfs') MOUNTPOINT('[PATHPREFIX]/usr/lpp/HOD')  
TYPE(HFS)
```

where hfsprfx is the name of the qualifier used in the HOMHFS installation job.

Regardless of whether a new or existing HFS is used, you must run the HOMISMKD job to create the directory structure for the Host On-Demand product.

If you obtained Host On-Demand as part of a CBPDO, follow the installation instructions found in the CBPDO RIMLIB data set to receive the Host On-Demand FMID, HHOJ700.

Important: Depending on when the product was ordered, you may receive PTF tapes in addition to the base product tapes. You must install the base product before installing the PTF tapes. The z/OS installation requires the base to be fully installed. The hod70mvs.sh shell script performs tasks such as symbolic links that the PTF shell script does not.

3.3 Activating Host On-Demand Service Manager

Before you can use a Host On-Demand server, there are several steps to complete:

- ▶ Set up the UNIX System Services environment.
- ▶ Set up the Security Server (RACF) if you plan to run the HOMSERVER as a started task.
- ▶ Set up the Web server environment.
- ▶ Modify the HOMSERVER sample job to suit your environment.
- ▶ Start the Host On-Demand Service Manager.

Note: The basic installation assumes the Host On-Demand port will be 8999, the recommended port. To change the port, refer to 3.3.6, “Changing the configuration port” on page 90.

3.3.1 UNIX System Services environment

Make sure the CLASSPATH, LIBPATH and PATH statements are all set correctly in the UNIX System Services (USS) environment. The USS file is commonly stored as /etc/profile. Since Host On-Demand supports both Java 1.1.8 (for all supported Host On-Demand OS/390 and z/OS releases) and Java 1.3.X (for OS/390 V2R9 and above), you need to point to your installed level of Java. You should also make sure that the CLASSPATH, LIBPATH, and PATH statements are included in the WebSphere product statements, for example /usr/lpp/internet/bin.

Note: The actual paths for Java may vary depending on your installation.

- ▶ The CLASSPATH environment variable should point to your z/OS Java library directory base class. The statement below is included for Java 1.1.8 support. For Java 1.3.X support, CLASSPATH is not required.

```
CLASSPATH=/usr/lpp/java/J1.1/lib/classes.zip
```

- ▶ The LIBPATH environment variable should point to the correct Java library. Include one of the following paths; the first is for Java 1.1.8 support, the second for Java 1.3.X support.

```
LIBPATH=/usr/lpp/java/J1.1/lib
LIBPATH=/usr/lpp/java/IBM/J1.3/lib
```

- ▶ The PATH environment variable should point to the correct Java library. Include one of the following paths; the first is for Java 1.1.8 support, the second for Java 1.3.X support.

```
PATH=/usr/lpp/java/J1.1/bin
PATH=/usr/lpp/java/IBM/J1.3/bin
```

- ▶ The JAVA_HOME environment variable should point to the correct Java library for Java 1.1.8 only. JAVA_HOME is not used for Java 1.3. The double quotes are necessary.

```
JAVA_HOME="/usr/lpp/java/J1.1"
```

3.3.2 Security Server (RACF) considerations

The HOMSEVR sample job is supplied to start Host On-Demand. It runs the ServiceManager.sh shell script, which starts the Host On-Demand Service Manager (NCServiceManager).

The HOMSERVER procedure must be started from a user ID with root authority in z/OS UNIX System Services. Our examples are for z/OS Secureway Security Server RACF. Here are the basic instructions to enable Host On-Demand to be assigned to the appropriate user ID:

1. Create a user ID for the HOMSERVER procedure (for example, HOMSRV).
 - a. Choose a default group that is defined to z/OS UNIX (for example, it has an z/OS UNIX segment with a group identification number (GID) defined. You might have a group called OMVSGRP that includes all z/OS UNIX users).
 - b. Add a z/OS UNIX segment, giving the user ID root authority by assigning it a user identification number (UID) of 0.
2. Create a started class entry (or update ICHRIN03) for the HOMSERVER procedure.
 - a. Make the entry name procname.* (for example, HOMSRV.*).
 - b. Assign =MEMBER to the user, thereby making the user ID and the procname the same (for example, HOMSRV.*).
 - c. Assign the procname to the z/OS UNIX group that is the default group for the user (for example, OMVSGRP).

Here is an example:

Associate the Host On-Demand Started Task with a RACF user ID that has an OMVS segment defined:

```
RDEFINE STARTED HOMSRV STDATA(USER(TCPIPOE))
SETROPTS RACLIST(STARTED) REFRESH
```

Where TCPIPOE is the user name with which the started task is associated.

If you want to create a different user ID to be used with Host On-Demand, issue the following commands:

```
ADDUSER HODSRV OMVS(HOME('/') UID(777))
      DFLTGRP(OMVSGRP) AUTHORITY(CREATE) UACC(ALTER)
RDEFINE STARTED HOMSRV STDATA(USER(HODSRV))
SETROPTS RACLIST(STARTED) REFRESH
```

Remember that the GROUP also has to have an OMVS segment defined.

3.3.3 Web server environment

To activate the Host On-Demand functions, the Web server must be configured to allow the HTML pages, class files, and JavaScript files to be downloaded to the user's machine. The name of the Web server configuration file is /etc/httpd.conf and it contains the configuration statements called directives.

The Pass directive specifies a template for requests that you want to be passed from the server. These Pass rules must be in the order listed and be placed prior to the ending Pass rule, Pass /*. The rules assume an alias of /hod/. Keep in mind that the alias is also case-sensitive. Consult the Web Server documentation for details.

Example 3-1 z/OS Web server Pass statements

```
Pass /hod/*.html /usr/lpp/HOD/hostondemand/HOD/*.html.ascii
Pass /hod/*.HTML /usr/lpp/HOD/hostondemand/HOD/*.HTML.ascii
Pass /hod/*.js /usr/lpp/HOD/hostondemand/HOD/*.js.ascii
Pass /hod/*.properties /usr/lpp/HOD/hostondemand/HOD/*.properties.ascii
Pass /hod/*.props /usr/lpp/HOD/hostondemand/HOD/*.props.ascii
Pass /hod/* /usr/lpp/HOD/hostondemand/HOD/*
```

If you are using a directory path other than /usr/lpp/HOD, change the directory path to the correct path, for example /service/usr/lpp/HOD.

Data type directives must be added among the other rules in the AddType section of the file:

```
AddType .cab          application/octet-stream binary 1.0
AddType .jar           multipart/x-zip binary 1.0
```

If you are using a directory path other than /usr/lpp/HOD, change the directory path to the correct path, for example /service/usr/lpp/HOD.

If you wish to publish your custom HTML pages from a separate user publish directory you will need to add a pass rule pointing the alias to your separate publish directory:

```
Pass /user_alias/* /user_publish_dir/*
```

For example:

```
Pass /hodpages/* /var/hod/customHTML/*
```

In this example the user_alias is hodpages and our customized files will be stored in the directory /var/hod/customHTML.

3.3.4 Modify the HOMSERVER sample job

To start the Service Manager with a started task, copy HOMSERVER sample procedure to a PROCLIB known to the system, and make the necessary changes to your installation. In this example we renamed the sample HOMSERVER procedure to HODSRV. If you have a different path structure for the Host On-Demand HFS, make the change in the PARM field. If you need to direct the STDOUT and STDERR to a file, you can modify the JCL like the following example to redirect the output to a temporary file.

Example 3-2 Sample Host On-Demand started procedure

```
//HODSRV  PROC
//*
//*  Function: IBM WebSphere Host On-Demand Server JCL
//*
//HODSRV   EXEC PGM=BPXBATCH,REGION=OK,TIME=NOLIMIT,
//      PARM='sh /usr/lpp/HOD/hostondemand/lib/ServiceManager.sh'
//SYSPRINT DD SYSOUT=*
//SYSIN    DD DUMMY
//SYSERR   DD SYSOUT=*
//STDOUT   DD PATH='/tmp/homserver-stdout',
//          PATHOPTS=(OWRONLY,OCREAT,OTRUNC),
//          PATHMODE=SIRWXU
//STDERR    DD PATH='/tmp/homserver-stderr',
//          PATHOPTS=(OWRONLY,OCREAT,OTRUNC),
//          PATHMODE=SIRWXU
//SYSOUT    DD SYSOUT=*
```

3.3.5 Start the Host On-Demand Service Manager

The Service Manager can be started one of two ways, either as a started task using the HODSRV job or by entering the shell script from the OMVS shell. You cannot start it from the ISHELL because the environment variables set in /etc/profile will not be used.

To start from the z/OS console using the sample procedure defined in Example 3-2 on page 89, enter:

```
s hodsrv
```

In this example the started task name, HODSRV, is less than 8 characters hence job names HODSRVx and HODSRVy will also be created, where x and y is a numeric between 1 and 9. In this scenario HODSRV1 and HODSRV2 were created as shown in Figure 3-1. If the started task name is 8 characters all three job names will be the same.

SDSF	DA	SC48	SC48	PAG	0	SIO	0	CPU	9/	7	LINE 1-3 (3)	
COMMAND	INPUT	==>									SCROLL ==>	PAGE
NP	JOBNAME	StepName	ProcStep	JobID	Owner	C	Pos	DP	Real	Paging	SIO	
	HODSRV	HODSRV	*OMVSEX	STC29265	STC		LO	FF	269	0.00	0.00	
	HODSRV1	STEP1		STC29264	STC		LO	FF	281	0.00	0.00	
	HODSRV2	STEP1		STC29263	STC		IN	F9	3568	0.00	0.00	

Figure 3-1 Job names

From the OMVS shell, there are a couple of ways to start the Service Manager:

```
/usr/lpp/HOD/hostondemand/lib/ServiceManager.sh &
```

or

```
cd /usr/lpp/HOD/hostondemand/lib
ServiceManager.sh &
```

Remember to include the '&' in order to run the shell script in the background; otherwise the ID will be unusable.

Once the Service Manager is started you will get the following message in STDOUT:

```
RDR0008: Native library failed to load, indicating this Redirector does not
support SSL.
```

The message is self-explanatory and can be ignored. If you would like to eliminate the message, edit the NSMprop file in the private library, /usr/lpp/HOD/hostondemand/private. Change the following parameter from YES to NO:

```
REDIRECTOR_AUTOSTART = NO
```

The Service Manager will need to be stopped and restarted to pick up the change.

3.3.6 Changing the configuration port

To change the configuration port for Host On-Demand on z/OS, the port must be specified in two places:

1. The NSMprop file, found in the /hostondemand/private directory, sets properties for the server. This file is in EBCDIC and does not need to be downloaded to be edited. Edit the NSMprop file and change the port number in the following line found in the CONFIGSERVER section. For our example we changed the port from 8999 to 8900:

```
CONFIGSERVER_PARMS = %INSTALL_PATH% 8900
```

2. The config.properties.ascii file sets the port for the clients. The file resides in the publish directory. Host On-Demand on z/OS ships with a config.properties.ascii file located in a subdirectory in the publish directory, /usr/lpp/HOD/hostondemand/HOD/hod. You need to edit the file on an ASCII based system. Add the following to set the port to 8900:

ConfigServerPort=8900

Then transfer the file to the z/OS server in binary. The reason for this is that the client applets expect ASCII text in config.properties.ascii, but files created or edited on z/OS are stored in EBCDIC.

After these changes have been made, the Service Manager can be started. To verify the correct port is listening, issue the TCP/IP **netstat conn** command:

To verify the client is making requests on the new port, access HODAdmin.html.

If you receive the message window as shown in Figure 3-2, verify you have transferred the file in binary mode from the workstation and the filename is config.properties.ascii.

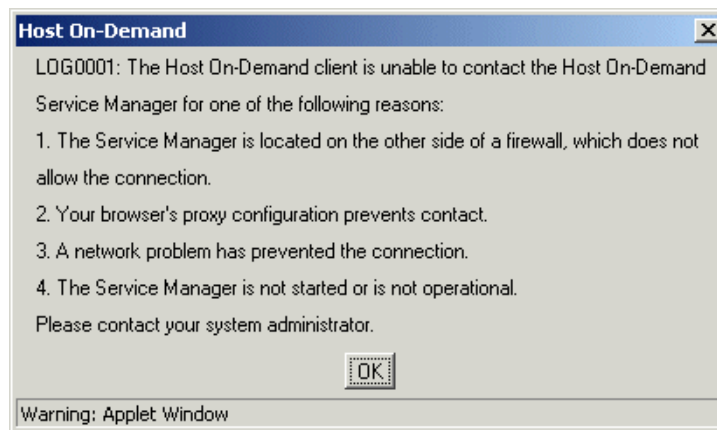


Figure 3-2 LOG0001 error window

Migration concern

If your existing Host On-Demand configuration was using a port other than 8999, you must be careful during migration. During the install process, the files in the private directory are unchanged, including the config.properties.ascii file. Also, in Host On-Demand V7 the NSMprop file is now installed in the directory, /usr/lpp/HOD/hostondemand/private. Therefore, you must edit the file to change the ConfigServerPort parameter to indicate the port you wish to use. For more migration concerns refer to 2.5, "Migration considerations" on page 66.

3.3.7 Stopping the Service Manager

If the Service Manager was started as a started task, the task cannot be stopped with the **purge** command as it will not work; the **cancel** command must be used. If the task name is HODSRV, stop it with the z/OS command:

```
c HODSRV
```

Note, you do not need to cancel the additional two jobs created by the system as these will be stopped when the main task is cancelled. It is recommended to use a started task name less than 8 characters. If the started task name is 8 characters, to cancel the Service Manager you will need to specify the ASID of the main task on the cancel command.

Important: If the level of Java on the z/OS is J1.3.1, when the cancel command is issued, CEEDUMP and HPITRACE files are created in the Host On-Demand /lib directory. If these files are not removed eventually the HFS will become full. You can manually remove these files or add the **rm** command to the ServiceManager.sh script as shown in Example 3-3. Note, if you stop Host On-Demand by killing the USS process these files will not be created.

Example 3-3 ServiceManager.sh with rm command

```
rm /usr/lpp/HOD/hostondemand/lib/CEEDUMP*
rm /usr/lpp/HOD/hostondemand/lib/HPITRACE*
```

If the Service Manager was started from the OMVS shell, you need to determine the PID of the Service Manager by issuing either, as super user:

```
ps -ef
```

from the OMVS shell, or:

```
d omvs,a=all
```

from the z/OS console. Two processes will be running:

```
WEBSRV 67895616 51118403 - 11:09:59 ttyp0000 0:03 java -Djava.compiler=off -classpath .:sm.zip:ibmjndi.jar:jndi.jar:jsdk.jar:ods
WEBSRV 51118403 84672827 - 11:09:59 ttyp0000 0:00 /bin/sh ./ServiceManager.sh
```

You must issue the **kill** command with the PID of the Java -classpath process even though the ServiceManager.sh process is the parent. For example:

```
kill -9 67895616
```

The ServiceManager.sh shell script issues the Java command that actually starts the server. If you kill the ServiceManager.sh process, the Java -classpath process remains. Then, if you try to restart the Service Manager, you will get the following error:

```
remote.Server. : ServerSocket Constructor Failed: EDC8115I Address already
in use.
*** Error - Failed to start Service Manager on port 8999
```

You must kill the Java -classpath process so the ServiceManager.sh process will also be killed and then the server can be restarted. You can create a shell script to kill Host On-Demand as shown in Example 3-4. In this example, we created a shell script, stophodsrv.sh, in the /private directory.

Example 3-4 Shell script to kill Host On-Demand

```
#!/bin/sh
#
HODTMP1=/tmp/tmp1
HODTMP2=/tmp/tmp2
HODTMP3=/tmp/tmp3
#
echo "Shell script looking for HODSRV group process id and kill it"
#
ps -e -o pgid,pid,ppid,args >$HODTMP1
egrep "ServiceManager.sh" $HODTMP1 >$HODTMP2
if test ! $? -eq 0
then
    echo "Could not find the Host On-Demand ServiceManager"
    echo
    exit 1
fi
i=0
while test $i -lt 1
do
    read gprocid cprocid pprocid junk
    echo $gprocid $cprocid $pprocid
    let i=$i+1
done <$HODTMP2 >$HODTMP3
echo "killing HODSRV group process " $gprocid
kill -- -$gprocid
rm $HODTMP1
rm $HODTMP2
rm $HODTMP3
```

The shell script can also be executed from an MVS procedure. A sample procedure is shown in Example 3-5, stophodsrv.sh is the name of the script we created in Example 3-4.

Example 3-5 Procedure to kill Host On-Demand

```
//STOPHOD PROC
//*****
//STOPHOD EXEC PGM=BPXBATCH,REGION=OK,TIME=NOLIMIT,
// PARM='sh /usr/lpp/HOD/hostondemand/private/stophodsrv.sh'
//SYSPRINT DD SYSOUT=T
//SYSIN DD DUMMY
//SYSERR DD SYSOUT=T
//STDOUT DD SYSOUT=T
//STDERR DD SYSOUT=T
//SYSOUT DD SYSOUT=T
//***STDENV DD PATH='/etc/leopt'
```

3.3.8 Considerations when running multiple TCP/IP stacks

If you are running multiple TCP/IP stacks, you may need to establish affinity to a specific stack by setting the `_BPXK_SETIBMOPT_TRANSPORT` environment variable. The variable can be set in one of two ways: add the variable to the `ServiceManager.sh` shell script or create a data set or PDS member that contains the environment variable and is pointed to by the `SYSENV DD` name in the started task. We recommend having a data set with the environment variable and not editing the `ServiceManager.sh` shell script. We do not recommend adding the environment variable to the `/etc/profile` in the event other processes will be establishing affinity to other stacks.

```
export _BPXK_SETIBMOPT_TRANSPORT=xxxxxx
```

Where `xxxxxx` is the name of the TCP/IP stack with which you want to establish affinity.

If you have multiple stacks and it is not necessary to establish affinity to a specific stack, then do not set the `_BPXK_SETIBMOPT_TRANSPORT` environment variable. The Host On-Demand server will bind to each stack that is active and can be accessed by the host name of each stack as long as a Web server is active on each stack.

3.3.9 Miscellaneous information

The following are additional functions that a z/OS user should be concerned with.

OS/400 Proxy server port

By default the OS/400 Proxy server is automatically started and it listens on port 3470. If the OS/400 Proxy server is not required, it may be disabled by following these instructions.

1. Start the Service Manager and log on to the administration applet from a workstation.
2. Select **OS/400 Proxy Server** in the left frame.
3. Select the **No** radio button, then click **Apply**.
4. Log off the administration applet.
5. Stop and restart the Service Manager and the port 3470 will no longer be open.

For more information on the OS/400 proxy server, refer to Chapter 10, “OS/400 Proxy” on page 419.

Web server timeout directives

During the download of the cached client from a z/OS Host On-Demand server, you may encounter what appears to be a hang. The default values of the Web server timeout values may not be sufficient for Host On-Demand clients, especially for dial-up connections. We recommend the following start values for the timeout directives in the httpd.conf file if you are supporting dial-up users.

InputTimeout	10 min
OutputTimeout	20 min (may need to be increased for slow connections)
ScriptTimeout	10 min
PersistTimeout	20 sec

You may need to adjust the values based on your environment. Refer to *IBM HTTP Server Planning, Installing, and Using*, SC34-4826.

Removing Host On-Demand on z/OS

To remove Host On-Demand on the z/OS platform, you must use SMP/E to delete the product from the SMP/E environment. Refer to the *SMP/E User's Guide*, SC28-1740 and *SMP/E Reference*, SC28-1806 to delete the product.

Error starting Service Manager, address already in use

If you receive an error message indicating the port address is already in use when starting the Service Manager, you need to check the IPL parameters in the BPXPRMxx member of SYS1.PARMLIB. In the FILESYSTYPE for the transport type of CINET, the parameters INADDRANYPORT and INADDRANYCOUNT reserve ports that the system will use. If the port you have specified for Host On-Demand is in the range specified by these parameters, Host On-Demand cannot use the port. For details on these parameters, refer to *MVS Initialization and Tuning Reference*, SA22-7592. The port can also be reserved via the z/OS Communications Server TCP/IP stack.

3.4 Deployment Wizard considerations

The Deployment Wizard does not run on z/OS, but runs only on a Windows platform. With Host On-Demand V7 the Deployment Wizard can be installed on a Windows platform from either a Host On-Demand Windows CD or by downloading setupDW.exe from HODMain.html. For further instructions see “Starting the Deployment Wizard” on page 530.

3.4.1 Deployment Wizard files

The administrator can select the type of output to be generated by the Deployment Wizard. If Output HTML is selected a number of files will be generated which must be transferred to the z/OS server. If Output Zip is checked the Deployment Wizard will create a zip file. In Host On-Demand V7 after the zip file has been transferred to the z/OS server it can be unzipped using the DWunzip utility. For detailed information about using the Deployment Wizard refer to Chapter 14, “Deployment Wizard” on page 529.

All Deployment Wizard output must be transferred to the z/OS server in binary mode and the names are case-sensitive. We recommend using FTP to transfer the file to the z/OS server.

Tip: We recommend using the DWunzip utility as the tool creates files in the appropriate directory, appends the necessary file extensions and sets file permissions.

Transferring Deployment Wizard Output Zip files

For our example, using the Deployment Wizard, we selected the HTML-based model to configure two 3270 sessions. The file name created is called RaleighITSO. After checking Output Zip, RaleighITSO.zip was created by the Deployment Wizard. In this example, the HTML file will be published from a separate user directory: /var/hod/customHTML.

Once a zip file is created, follow these steps to deploy the files to the z/OS system.

1. Start an FTP session with your z/OS server system from the workstation.
2. Use the binary command to ensure the transfer mode is Image.
3. Change directory on the target system to your publish directory. This will either be the Host On-Demand publish directory or your user publish directory. In this scenario we will use our user publish directory:

```
cd /var/hod/customHTML
```
4. Change directory on the local client to the directory where the zip file resides. For example:

```
1cd c:\DWizard\ZIPfiles
```

5. Transfer the zip file:

```
put RaleighITS0.zip RaleighITS0.zip
bye
```

6. Logon to your z/OS server and change directory to where the DWunzip file is located. The z/OS DWunzip file is called DWunzip-S390 and located in the Host On-Demand /lib/samples/DWunzipCommandFiles directory. For example:

```
cd /usr/lpp/HOD/hostondemand/lib/sample/DWunzipCommandFiles
```

Edit DWunzip-S390 modifying the parameters to your installation directories as shown in Figure 3-3. If you have created a separate user publish directory modify MY_PUBLISHED_DIRECTORY to reflect this directory. Verify DWunzip has execute permissions. Use the **chmod** command to set the permissions if required. If you have mounted the Host On-Demand HFS as read only you will need to copy DWunzip-S390 to a directory with write permissions in order to update the file with your installation variables.

```
#####
# Modify the following to be your web-published directory.
# Note: This is also the directory where your zip file should be.
#####
MY_PUBLISHED_DIRECTORY=/var/hod/customHTML

#####
# Modify the following to be your Host On-Demand install directory.
#####
MY_HOD_DIRECTORY=/usr/lpp/HOD/hostondemand

#####
# Modify the following to specify your java engine
#####
JAVA_ENGINE=/usr/lpp/java/IBM/J1.3/bin/java

#####
# Modify the following line to specify the path of your java class library
#####
JAVA_LIB_CLASSES=/usr/lpp/java/IBM/J1.3/lib
```

Figure 3-3 DWunzip-S390

Transferring Deployment Wizard Output HTML files

If Output HTML is selected you will need to transfer all the Deployment Wizard files, in binary, to the z/OS server. You will also need to append .ascii to the .html and .txt files.

In this example, using the Deployment Wizard and the HTML-based model we created two 3270 sessions. The file name entered was Raleigh2ITS0 and Output HTML was checked. See Example 3-6 for a list of files that was created by the Deployment Wizard. In this example the HTML file will be published from the Host On-Demand publish directory.

Example 3-6 Sample Output HTML Deployment Wizard files

```
hostondemand\HOD\Raleigh2ITS0.html
hostondemand\HOD\z_Raleigh2ITS0.html
hostondemand\HOD\HODData\Raleigh2ITS0\cfg0.cf
hostondemand\HOD\HODData\Raleigh2ITS0\cfg1.cf
hostondemand\HOD\HODData\Raleigh2ITS0\params.txt
hostondemand\HOD\HODData\Raleigh2ITS0\policy.obj
hostondemand\HOD\HODData\Raleigh2ITS0\preloads.obj
hostondemand\HOD\HODData\Raleigh2ITS0\udparams.txt
hostondemand\HOD\HODData\Raleigh2ITS0\wInfo.txt
```

Follow the following steps to copy the files to the z/OS server.

1. Start an FTP session with your z/OS server system from the workstation.
2. Use the binary command to make sure the transfer mode is Image.
3. Change directory on the target system to the Host On-Demand publish directory or your user publish directory. In this example we have used the Host On-Demand publish directory:

```
cd /usr/lpp/HOD/hostondemand/HOD
```

4. Change directory on the local client to the directory where the files reside. For example:

```
lcd c:\DWizard\HTMLfiles
```

5. Transfer the files, renaming the .html and .txt files to append .ascii:

Example 3-7 FTP Deployment Wizard generated files to z/OS

```
put Raleigh2ITS0.html Raleigh2ITS0.html.ascii
put z_Raleigh2ITS0.html z_Raleigh2ITS0.html.ascii
mkdir HODData
cd HODData
mkdir Raleigh2ITS0
cd Raleigh2ITS0
lcd HODData\Raleigh2ITS0
mput cfg*.*
put params.txt params.txt.ascii
mput p*.obj
put udparams.txt udparams.txt.ascii
put wInfo.txt wInfo.txt.ascii
bye
```

6. Verify the permissions are 755 of all the files, including the subdirectories HODData and Raleigh2ITSO. If they are not, change them using the **chmod** command, either through the FTP session or by logging on to the z/OS system on TSO.

The customized client is now ready to be downloaded. If the Host On-Demand server and Web server were active before you uploaded, you do not need to recycle them to pick up the new HTML files.

If you are updating an existing customized HTML file, you should stop the Web server prior to uploading, since the FTP may fail if the custom page is in use.

3.5 Configuration Servlet setup

In this section we discuss how to configure the WebSphere Application Server for z/OS for the Host On-Demand Configuration Servlet using WebSphere Application Server. Host On-Demand V7 Configuration Servlet can be deployed using WebSphere Application Server V3.5 or V4.01. For further details about using the Configuration Servlet see Chapter 9, "Configuration Servlet" on page 397.

Our samples are based on WebSphere Application Server V3.5 for OS/390. See Chapter 15, *IBM WebSphere Host On-Demand Version 7.0 Planning, Installing, and Configuring Host On-Demand*, SC31-6301-00 for information on using WebSphere Application Server V4.01.

Unlike the Windows NT platform, z/OS does not provide a graphical interface to configure the Configuration Servlet. Several configuration files must be modified for the Configuration Servlet:

- ▶ was.conf
- ▶ httpd.conf
- ▶ httpd.envvars
- ▶ config.properties.ascii

3.5.1 Configuration files

The sample configuration files are based on our installation. The directory paths for Java, Web server, and WebSphere Application Server for your installation may be different. For our configuration, make the following assumptions:

- ▶ Java is located in /usr/lpp/java/IBM/J1.3
- ▶ Configuration files are located in /web/hod70, may normally be found in the /etc directory

- ▶ WebSphere Application Server is located in /usr/lpp/was35
- ▶ TCP/IP stack name is TCPIPOE

was.conf

The following statements were added to /web/hod70/was.conf. Each deployedwebapp and webapp statement is on a single line, but due to space constraints are shown on multiple lines in the example. Note the third deployedwebapp statement spans four lines.

Example 3-8 was.conf example

```

deployedwebapp.HOD.host=default_host
# rooturi must match pathname on Service statement in httpd.conf:
deployedwebapp.HOD.rooturi=/servlet
deployedwebapp.HOD.classpath=/usr/lpp/HOD/hostondemand/HOD:/usr/lpp/HOD/hostondemand/lib/cfgsrvlt.jar:/usr/lpp/java/IBM/J1.3/bin:/usr/lpp/HOD/hostondemand/HOD/com/ibm/eNetwork/HODUtil/services/remote:/usr/lpp/was35/servlet:/usr/lpp/HOD/hostondemand/HOD/com/ibm/eNetwork/HOD:/usr/lpp/java/IBM/J1.3/lib/rt.jar
deployedwebapp.HOD.documentroot=/usr/lpp/HOD/hostondemand/lib
deployedwebapp.HOD.autoreloadinterval=100000
webapp.HOD.jspmapping=*.jsp
webapp.HOD.jspmapping=*.jhtml
webapp.HOD.jsplevel=1.0
webapp.HOD.filemapping=/
# URL to servlet by code name or servletmapping alias listed below:
webapp.HOD.servlet.HODConfigServlet.code=com.ibm.eNetwork.HODUtil.services.remote.HODCfgServlet
webapp.HOD.servlet.HODConfigServlet.servletmapping=/HODConfig
webapp.HOD.servlet.HODConfigServlet.initargs=ConfigServer=9.12.6.126,ConfigServerPort=8900,ShowStats=true,Trace=true
webapp.HOD.servlet.HODConfigServlet.autostart=true

```

The ConfigServer parameter is the IP address of the TCP/IP stack with which the Sample Web server establishes affinity. Note that ConfigServerPort is 8900 instead of 8999. The ShowStats parameter allows you to list information about WebSphere Application Server to help verify it is configured correctly. The Trace parameter allows you to get trace information for problem determination.

httpd.conf

The following statements were added to our /web/hod70/httpd.conf. Note the ServerInit statement and the first Service statement are shown on multiple lines due to space constraints, but each must be on one line.

Example 3-9 httpd.conf example

```

# =====
# *** WAS directives ***

```

```
# =====
ServerInit /usr/lpp/was35/AppServer/bin/was350plugin.so:init_exit
/usr/lpp/was35,/web/hod70/was.conf
Service /webapp/examples/*
/usr/lpp/was35/AppServer/bin/was350plugin.so:service_exit
Service /examples/* /usr/lpp/was35/AppServer/bin/was350plugin.so:service_exit
Service /servlet/* /usr/lpp/was35/AppServer/bin/was350plugin.so:service_exit
Service /*.jsp /usr/lpp/was35/AppServer/bin/was350plugin.so:service_exit
ServerTerm /usr/lpp/was35/AppServer/bin/was350plugin.so:term_exit
```

httpd.envvars

The following is the entire contents of our /webhod70/httpd.envvars. note that PATH, NLSPATH, and LIBPATH are shown on multiple lines, but each must be on one line. Other variables may be different in your installation such as the timezone variable, TZ.

Example 3-10 httpd.envvars example

```
PATH=/bin:./usr/sbin:/usr/lpp/internet/bin:/usr/lpp/internet/sbin:/usr/lpp/ldap/bin:/usr/lpp/java/IBM/J1.3/bin
SHELL=/bin/sh
TZ=EST5EDT
LANG=C
LC_ALL=en_US.UTF-8
NLSPATH=/usr/lib/nls/msg/%L/%N:/usr/lpp/internet/%L/%N:/usr/lpp/ldap/lib/nls/msg/%L/%N:/usr/lpp/was35/AppServer/msg/%L/%N
LIBPATH=/usr/lpp/internet/bin:/usr/lpp/internet/sbin:/usr/lpp/ldap/lib:/usr/lpp/java/IBM/J1.3/lib
JAVA_HOME=/usr/lpp/java/IBM/J1.3
_BPXK_SETIBMOPT_TRANSPORT=TCPIPOE
STEPLIB=CURRENT
```

The JAVA_HOME variable is necessary for WebSphere Application Server even though the variable is not needed for the Host On-Demand server.

config.properties.ascii

The clients can be enabled to use the Configuration Servlet in one of two ways. It depends on how you choose to deploy the Configuration Servlet; either all clients or only some clients will use the servlet. If all clients will be enabled to use the Configuration Servlet, then the config.properties.ascii will need to be modified. Set the ConfigServerURL parameter in the config.properties.ascii file. It must be edited on an ASCII machine then transferred in binary to the publish directory. A sample config.properties.ascii file is shipped in the /usr/lpp/HOD/hostondemand/HOD/hod directory.

Example 3-11 config.properties.ascii example

```
ConfigServerPort=8900  
ConfigServerURL=servlet/HODConfig/hod
```

If only some clients will be enabled to use the Configuration Servlet, change the HTML page for the clients that use the servlet. For example if your clients will be using the download client, modify or make a copy of the HOD.html.ascii file and add the following parameter in the HTML page:

```
<PARAM NAME=ConfigServerURL VALUE=/servlet/HODConfig/hod>
```

Since HTML pages on the z/OS server are ASCII, you must transfer the file in binary to an ASCII machine. Edit the file, then transfer back to the z/OS server. You could also add ConfigServerURL to a client HTML page created by the Deployment Wizard.

3.5.2 Testing the servlet

Restart the Web server and Host On-Demand Service Manager if the config.properties.ascii file was modified. You can test the Configuration Servlet by invoking the ShowStats function. The ShowStats argument in the was.conf file must be set to true in order for the function to work. Specify the following URL from a browser:

```
http://server_name/servlet_location/HODConfig/info
```

Using our configuration the URL would look like the following:

```
http://server_name/servlet/HODConfig/info
```

When successful your browser will return a window similar to that shown in Figure 3-4.

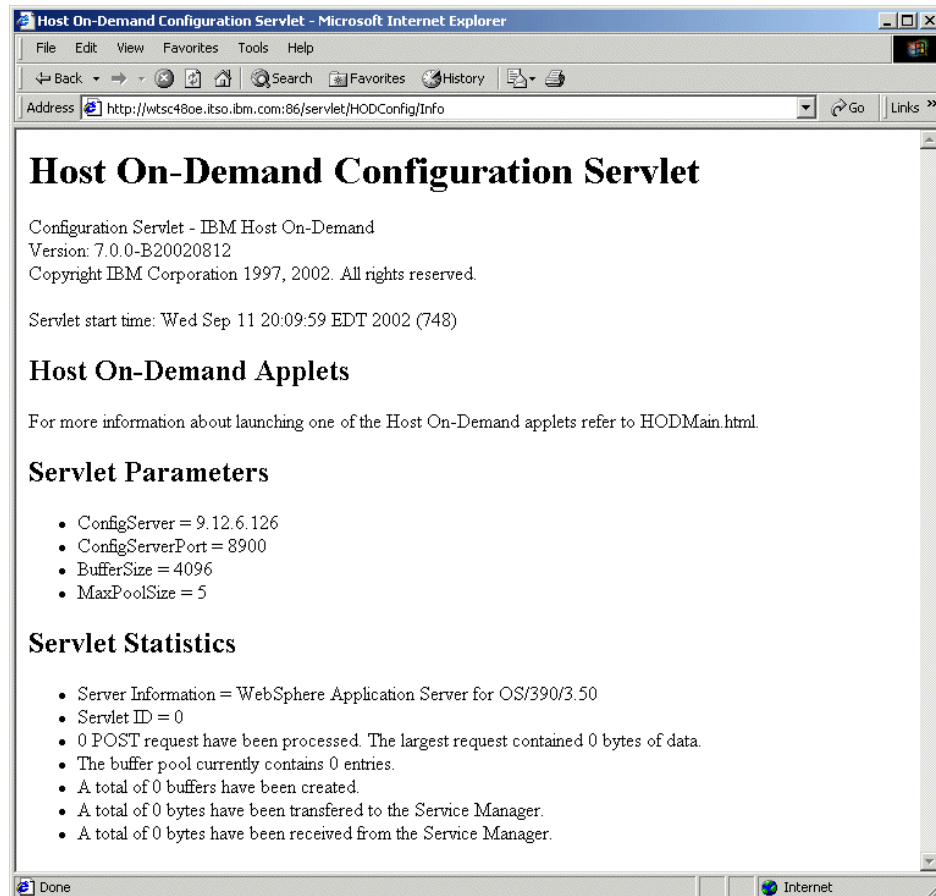


Figure 3-4 Configuration Servlet testing information

After invoking the ShowStats function, we accessed the HODCServ.html page that was modified. We then logged on to a Host On-Demand user. If it had failed you would have received the error window shown in Figure 3-5 on page 104.

3.5.3 Changing the Configuration Server port

When you change the port for Host On-Demand as described in 3.3.6, “Changing the configuration port” on page 90, you must also change the following webapp statement in the was.conf file:

```
webapp.HOD.servlet.HODConfigServlet.initargs=ConfigServer=9.12.6.126,ConfigServerPort=8900,ShowStats=true,Trace=true
```

Restart the Web server and Host On-Demand server to use the new port.

3.5.4 Problem determination

If anything is incorrect in the configuration you may receive the following error when trying to access an HTML page.

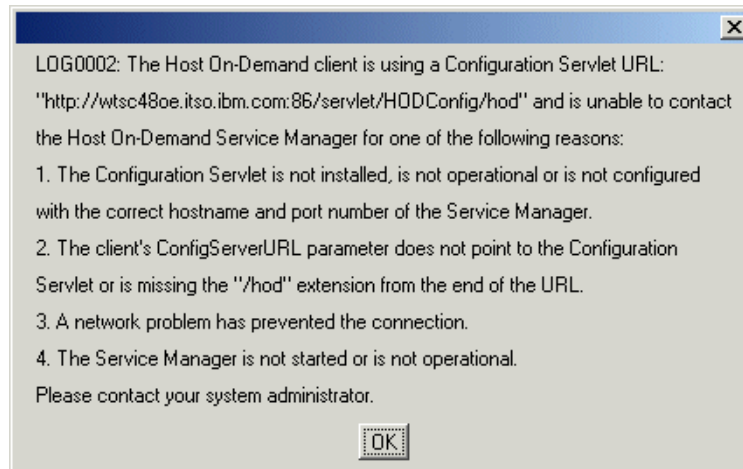


Figure 3-5 LOG0002 error window

If you receive the LOG0002 error, check the following items:

- ▶ If using a port other than 8999:
 - Verify the Configuration Servlet is using the correct port by using the ShowStats function described in 3.5.2, “Testing the servlet” on page 102.
 - Verify the correct port is displayed.
 - Verify Host On-Demand server is listening on the correct port by issuing the following TSO command:

```
netstat conn
```
 - Verify the correct port is defined in the config.properties.ascii file. If you need to edit the file, transfer it in binary to an ASCII machine. Make any corrections, then transfer the file back to the host and restart the Web server and Host On-Demand server.
- ▶ Verify all the configuration files for any incorrect paths in was.conf, httpd.conf, and httpd.envvars. If any changes are made to these files, restart the Web server.
- ▶ When entering the URL, remember the URL is case-sensitive. If you have the following in the was.conf file:

```
webapp.HOD.servlet.HODConfigServlet.servletmapping=/HODConfig
```

You also need to specify the ConfigServerURL as `servlet/HODConfig/hod` in either the HTML or the `config.properties.ascii` file, depending on how the Configuration Servlet is enabled to the clients. In this case the URL to access the Configuration Servlet ShowStats would be:

```
http://server_name/servlet_alias/hodconfig/info
```

- ▶ If you had specified `Trace=true` in the `was.conf` file on the `webapp.HOD.servlet.HODConfigServlet.initargs`, then you can access the trace, configuration and statistic information from the Configuration Servlet for debugging purposes. To view the trace, load the following URL into your browser:

```
http://server_name/servlet_alias/HODConfig/Trace
```

The Configuration Servlet's trace information will be displayed in the browser. This information may help in determining what is in error. For example, if you see an exception like the following:

```
java.net.ConnectException: EDC8128I Connection refused
```

It indicates the Configuration Servlet is unable to connect to the Host On-Demand server. The port definitions may be incorrect. The Host On-Demand server may not be operational.

- ▶ Verify the ConfigServer parameter is pointing to the correct TCP/IP stack if you have multiple stacks.

3.6 Using SSL with Communications Server for z/OS

Communications Server for z/OS supports data encryption through the Secure Sockets Layer (SSL) protocol. Beginning with Communications Server for OS/390 V2R10 the use of RACF as a repository for the server's keyring is supported. OS/390 V2R10 also introduced the `TELNETPARMS CONNTYPE` statement that allows a client to connect to a Telnet port either as secured or basic, which allows security negotiation on a single port.

There are three main scenarios:

1. A Host On-Demand client can be configured to make an SSL-secured connection directly to a Communications Server for z/OS server, having been loaded from a separate Host On-Demand server.
2. A Host On-Demand Redirector on Windows NT or AIX can be configured to make an outgoing SSL connection to a Communications Server for z/OS.
3. When a Host On-Demand server and Communications Server for z/OS are installed on the same system, a client downloaded from z/OS can make an SSL connection to the z/OS Telnet server without the use of the Redirector.

The z/OS set up required for scenarios 1 and 2 are the same. Communications Server for z/OS uses gskkyman for its key management and the keyring database is of the kdb type. Prior to OS/390 V2R8, mkkf was the key management utility. This book will cover the usage of gskkyman and RACF. The procedure for putting the server's site certificate into the CustomizedCAs.class file is as follows:

1. On the z/OS, create the keyring file and a certificate request using gskkyman.
2. Store the unknown CA's certificate into the key database.
3. Receive the signed certificate into the key database.
4. Update the CustomizedCAs.class.

Scenario 3 is different because the key management utility on z/OS is not able to add the certificate to the class file database, CustomizedCAs.class. However, a Java program named keyrng.class is provided by Host On-Demand to add the certificate to the CustomizedCAs.class file. This is demonstrated in "Make certificates available to clients" on page 119.

3.6.1 Telnet Server and SSL support

The z/OS TN3270 Telnet server supports the Secure Sockets Layer (SSL) protocol. This provides secure data transmission between a secure port and an SSL-enabled client. In Communications Server for OS/390 V2R8, SSL support was enhanced to support three levels of client authentication (allowing additional authentication and access control by means of a certificate that must be presented to the server by a client):

- ▶ Client authentication defined in the SSL specification, Level 1
- ▶ Client authentication against the certificate stored in RACF, Level 2
- ▶ Client authentication with the SERVAUTH RACF class, Level 3

In the TCP/IP profile, three keywords can be used for the CLIENTAUTH parameter in the TELNETPARMS block:

NONE	Indicates that no client authentication is required during the SSL handshake.
SSLCERT	Specifies that the SSL handshake process authenticates the client certificate as well as the server certificate. This is Level 1 security support.
SAFCERT	Indicates the additional validation associated with Levels 2 and 3. Level 2 requires the certificate to be stored in RACF, and Level 3 requires the SERVAUTH RACF class is in effect.

In Communications Server for OS/390 V2R10, the Security Server (RACF) provided common keyring support, so no key database is required. All certificates can be managed through the RACF database.

Telnet-negotiated session

A Telnet-negotiated session determines if the security negotiations between the client and the Telnet server are done on the established Telnet connection or on an SSL connection prior to the Telnet negotiation. For the client to use this feature, the Telnet server must support Telnet-negotiated security. The other SSL options are valid regardless of whether Telnet-negotiated is set to Yes or No.

In OS/390 V2R10 or above, the `CONNTYPE ANY` keyword in the `TELNETPARMS` block signifies that the Telnet server can support both SSL clients and non-SSL clients over a single port. The Telnet server first establishes a Telnet session then negotiates security. If the client wishes to enter into a secure connection, SSL protocols will be used for all subsequent communication. If the client is not willing to enter a secure connection, a non-SSL or basic connection is used. For a complete discussion of Telnet-negotiated sessions, refer to “Telnet-negotiated sessions” on page 1023.

Note: Do not use `CONNTYPE ANY` if you are going through a firewall, because this will allow a non-SSL connection through the firewall. For details about the `CONNTYPE` keyword, refer to the *IBM Communications Server IP Configuration Reference* manual for your operating system release.

3.6.2 SSL encryption overview

Host On-Demand V7 has one FMID providing all the encryption support, HHOJ700.

In an SSL-encrypted session, any data on a secure port is encrypted by means of the SSL protocol before it is sent to the client. Data received from the client is decrypted before the data is sent to other processes, such as VTAM. The flows between Telnet and VTAM are unchanged.

Secure connections are made through a secure port. When running with base TCP/IP, Telnet connections across ports defined as secure are protected by way of MD5 or SHA hashing algorithms and support SSL V3 clients, but do not provide data encryption. SSL Encryption support by way of RC2, RC4, DES, or triple DES requires one of the optional features shown in the tables below.

The following table describes the FMIDs for the respective levels of OS/390 and z/OS.

Table 3-1 Encryption FMIDs

Encryption Feature	Base	Level 1	Level 2	Level 3
V2R8	HTCP380	JTCP383	JTCP382	JTCP38K
V2R10	HTCP50A	HTCP53A	HTCP52A	JTCP5KA
zOS V1R1	HTCP50A	HTCP53A	HTCP52A	JTCP5KA
zOS V1R2	HIP6120	N/A	N/A	JIP612K
zOS V1R3	HIP6120	N/A	N/A	JIP612K

The following table provides the level of security that each level provides.

Table 3-2 Encryption features for OS/390

Level	SSL V3 Clients	SSL V2 Clients
Base	NULL SHA NULL MD5 NULL NULL	Not supported
Level 1	RC4 MD5 Export RC2 MD5 Export NULL SHA NULL MD5 NULL NULL	RC4 Export RC2 Export
Level 2	DES SHA RC4 MD5 Export RC2 MD5 Export NULL SHA NULL MD5 NULL NULL	RC4 Export RC2 Export
Level 3	Triple DES SHA US DES SHA RC4 MD5 Export RC4 SHA US RC4 MD5 US RC2 MD5 Export NULL SHA NULL MD5 NULL NULL	Triple DES US DES US RC4 Export RC4 US RC2 Export RC2 US

Refer to “Basic concepts of cryptography and digital certificates” on page 990 for descriptions of the encryption elements.

For more information about the security levels, please refer to:

- ▶ *IBM Communications Server IP Configuration Reference* manual for your operating system release.

You can find additional information on the following Web sites:

About SSL protocol:

<http://home.netscape.com/eng/ss13/ssl-toc.html>

- ▶ About the encryption methodology:

<http://www.verisign.com/repository/crptintr.html>

3.6.3 SSL Configuration using gskkyman

In this section we discuss the use of the gskkyman utility to do the following:

- ▶ Create the key database
- ▶ List all trusted CAs
- ▶ Create key pair and certificate request
- ▶ Store a CA certificate
- ▶ Receive a certificate issued for the request
- ▶ Create a self-signed certificate
- ▶ Make certificates available to clients
- ▶ Server Authentication using a certificate from an unknown Certificate Authority
- ▶ Client authentication
- ▶ Transport Layer Security-based security (OS/390 V2R10 and higher)

Create the key database

If you do not have a key database, you must create it using either gskkyman utility, shipped as part of the Cryptographic Services, or the Security Server (RACF) if on OS/390 V2R10 or higher.

Note: When using gskkyman, do not create your key database in any of the Host On-Demand directories, for security and migration reasons.

If using gskkyman, go to the OMVS shell and follow the steps shown in Example 3-12. Before using the utility, you might need to make gskkyman known to the UNIX System Services environment:

```
export STEPLIB=GSK.SGSKLOAD (verify name)
```

Example 3-12 Creating the key database

```
CASEY @ SC48:/u/casey>gskkyman
```

IBM Key Management Utility

Choose one of the following options to proceed.

- 1 - Create new key database
- 2 - Open key database
- 3 - Change database password

- 0 - Exit program

Enter your option number: **1**

Enter key database name or press ENTER for "key.kdb": **itso.kdb**

Enter password for the key database.....>

Enter password again for verification.....>

Should the password expire? (1 = yes, 0 = no) 1": **0**

The database has been successfully created, do you want to continue to work with the database now? (1 = yes, 0 = no) 1": **1**

Key database menu

Current key database is /u/casey/itso.kdb

- 1 - List/Manage keys and certificates
- 2 - List/Manage request keys
- 3 - Create new key pair and certificate request
- 4 - Receive a certificate issued for your request
- 5 - Create a self-signed certificate
- 6 - Store a CA certificate
- 7 - Show the default key
- 8 - Import keys
- 9 - Export keys
- 10 - List all trusted CAs

- 11 - Store encrypted database password
- 0 - Exit program

Enter option number (or press ENTER to return to the parent menu): **11**

The encrypted password has been stored in file /u/casey/itso.sth

Your request has completed successfully, exit gskkyman? (1 = yes, 0 = no) 0": **0**

After creating the key database, you need to store the encrypted database password, option 11. This creates a .sth stash file.

List all trusted CAs

Next, we need to determine if the Certificate Authority you plan to use is in the list of trusted CAs (Example 3-13). If you plan to use a self-signed certificate, refer to "Create a self-signed certificate" on page 117.

Example 3-13 List trusted Certificate Authorities this key database knows

Key database menu

Current key database is /u/casey/itso.kdb

- 1 - List/Manage keys and certificates
- 2 - List/Manage request keys
- 3 - Create new key pair and certificate request
- 4 - Receive a certificate issued for your request
- 5 - Create a self-signed certificate
- 6 - Store a CA certificate
- 7 - Show the default key
- 8 - Import keys
- 9 - Export keys
- 10 - List all trusted CAs
- 11 - Store encrypted database password
- 0 - Exit program

Enter option number (or press ENTER to return to the parent menu): **10**

Trust CA certificate list

Key database name is /u/casey/itso.kdb

Please choose one of the following keys to work with.

- 1 - Integrion Certification Authority Root
- 2 - IBM World Registry Certification Authority

- 3 - Thawte Personal Premium CA
- 4 - Thawte Personal Freemail CA
- 5 - Thawte Personal Basic CA
- 6 - Thawte Premium Server CA
- 7 - Thawte Server CA
- 8 - Verisign Test CA Root Certificate
- 9 - RSA Secure Server Certification Authority

Enter a key number or press ENTER for more labels: **<Enter>**
Trust CA certificate list

Key database name is /u/casey/itso.kdb

Please choose one of the following keys to work with.

- 10 - Verisign Class 1 Public Primary Certification Authority
- 11 - Verisign Class 2 Public Primary Certification Authority
- 12 - Verisign Class 3 Public Primary Certification Authority

Enter a key number or press ENTER to return to parent menu: **<Enter>**

Create key pair and certificate request

If you need to request a certificate to be signed by a well-known Certificate Authority or an unknown Certificate Authority, you need to create a key pair and certificate request (Example 3-14).

Example 3-14 Create new key pair and certificate request

Key database menu

Current key database is /u/casey/itso.kdb

- 1 - List/Manage keys and certificates
- 2 - List/Manage request keys
- 3 - Create new key pair and certificate request
- 4 - Receive a certificate issued for your request
- 5 - Create a self-signed certificate
- 6 - Store a CA certificate
- 7 - Show the default key
- 8 - Import keys
- 9 - Export keys
- 10 - List all trusted CAs
- 11 - Store encrypted database password

- 0 - Exit program

Enter option number (or press ENTER to return to the parent menu): **3**

Enter certificate request file name or press ENTER for "certreq.arm":

itsoreq.arm

Enter a label for this key.....> **ITSO Certificate**

Select desired key size from the following options (512):

1: 512

2: 1024

Enter the number corresponding to the key size you want: **2**

Enter certificate subject name fields in the following.

Common Name (required).....> **wtsc48oe.itso.ibm.com**

Organization (required).....> **IBM**

Organization Unit (optional).....> **ITSO**

City/Locality (optional).....> **RTP**

State/Province (optional).....> **NC**

Country Name (required 2 characters)..> **US**

Please wait while key pair is created...

Your request has completed successfully, exit gskkyman? (1 = yes, 0 = no) 0": **1**

The label you enter will be the label you see when you display the list of certificates.

The common name is the fully qualified host name of the TN3270 server. If you select server authentication on the Host On-Demand session properties, the common name must match the host name in the DNS server for the IP address of the TN3270 server. In our example, the host name of our TN3270 server is wtsc48oe.itso.ibm.com.

Using the file that was created (in our example, itsoreq.arm) you can then send the request to the Certificate Authority of your choice. For these examples, we set up an ITSO Certificate Authority server for signing certificates.

Store a CA certificate

After you receive the signed certificate from the CA, you need to add the unknown CA certificate to your list of Trusted CAs (Example 3-15). Enter **gskkyman** and open the key database you previously created. If you requested a certificate from a CA already in the trusted list, you can skip this step and go to "Receive a certificate issued for the request" on page 116.

Example 3-15 Store a CA certificate

Key database menu

Current key database is /u/casey/itso.kdb

- 1 - List/Manage keys and certificates
- 2 - List/Manage request keys

- 3 - Create new key pair and certificate request
- 4 - Receive a certificate issued for your request
- 5 - Create a self-signed certificate
- 6 - Store a CA certificate
- 7 - Show the default key
- 8 - Import keys
- 9 - Export keys
- 10 - List all trusted CAs
- 11 - Store encrypted database password

- 0 - Exit program

Enter option number (or press ENTER to return to the parent menu): **6**
Enter certificate file name or press ENTER for "cert.arm": **itsoca.cer**
Enter a label for this key.....> **ITS0 Certificate Authority**

Please wait while certificate is stored...

Your request has completed successfully, exit gskkyman? (1 = yes, 0 = no) 0": **0**

Key database menu

Current key database is /u/casey/itso.kdb

- 1 - List/Manage keys and certificates
- 2 - List/Manage request keys
- 3 - Create new key pair and certificate request
- 4 - Receive a certificate issued for your request
- 5 - Create a self-signed certificate
- 6 - Store a CA certificate
- 7 - Show the default key
- 8 - Import keys
- 9 - Export keys
- 10 - List all trusted CAs
- 11 - Store encrypted database password

- 0 - Exit program

Enter option number (or press ENTER to return to the parent menu): **10**

Trust CA certificate list

Key database name is /u/casey/itso.kdb

Please choose one of the following keys to work with.

- 1 - ITS0 Certificate Authority
- 2 - Integrion Certification Authority Root
- 3 - IBM World Registry Certification Authority

- 4 - Thawte Personal Premium CA
- 5 - Thawte Personal Freemail CA
- 6 - Thawte Personal Basic CA
- 7 - Thawte Premium Server CA
- 8 - Thawte Server CA
- 9 - Verisign Test CA Root Certificate

Enter a key number or press ENTER for more labels: 1

Key Menu

Currently selected key: ITSO Certificate Authority

Choose one of the following options to proceed.

- 1 - Show key information
- 2 - Set the selected key as default
- 3 - View certificate of the key
- 4 - Remove trust root status
- 5 - Copy the certificate of this key to a file
- 6 - Delete the key
- 7 - Export the key to another database
- 0 - Exit program

Enter option number (or press ENTER to return to the parent menu):1

Basic information of the currently selected key

Unique ID:	13
Label:	ITSO Certificate Authority
Chosen as default key:	false
Key size:	1024
Set as trusted:	true
Private key existence:	false
User defined field existence:	false

Certificate information for the selected key

Version:	3
Serial number:	0582cf5027da20a945f0dadf594849b1
Issuer name:	ITSORaleigh ITSO IBM Raleigh, NC US

```

Subject name:      mticknor@us.ibm.com
                   ITSORaleigh
                   ITSO
                   IBM
                   Raleigh, NC
                   US
                   mticknor@us.ibm.com
Effective date:    08/27/02
Expiration date:   08/27/04
Signature algorithm OID: sha1WithRSASignature
Issuer unique ID:  false
Subject unique ID:  false
Number of extensions: 5

```

Selecting to list all trusted CAs shows the new CA as trusted and then you can show the key information. You cannot make this certificate the default at this time. Once a private key from this CA has been added to the key database, then the certificate can be made the default.

Receive a certificate issued for the request

Once you receive the signed certificate from the CA, you can receive the certificate into the key database for your request (Example 3-16).

Example 3-16 Receive certificate after signed by CA

Key database menu

Current key database is /u/casey/itso.kdb

- 1 - List/Manage keys and certificates
- 2 - List/Manage request keys
- 3 - Create new key pair and certificate request
- 4 - Receive a certificate issued for your request
- 5 - Create a self-signed certificate
- 6 - Store a CA certificate
- 7 - Show the default key
- 8 - Import keys
- 9 - Export keys
- 10 - List all trusted CAs
- 11 - Store encrypted database password
- 0 - Exit program

Enter option number (or press ENTER to return to the parent menu): **4**

Enter certificate file name or press ENTER for "cert.arm": **itso.cer**

Do you want to set the key as the default in your key database? (1 = yes, 0 = no) 1": 1

Please wait while certificate is received.....

Your request has completed successfully, exit gskkyman? (1 = yes, 0 = no) 0":1

Create a self-signed certificate

Using gskkyman, you can create a self-signed certificate (Example 3-17). Once the certificate is created, make this certificate the default.

Example 3-17 Creating a self-signed certificate

Key database menu

Current key database is /u/casey/itso.kdb

- 1 - List/Manage keys and certificates
- 2 - List/Manage request keys
- 3 - Create new key pair and certificate request
- 4 - Receive a certificate issued for your request
- 5 - Create a self-signed certificate
- 6 - Store a CA certificate
- 7 - Show the default key
- 8 - Import keys
- 9 - Export keys
- 10 - List all trusted CAs
- 11 - Store encrypted database password

- 0 - Exit program

Enter option number (or press ENTER to return to the parent menu): 5

Enter version number of the certificate to be created (1, 2, or 3) 3": 3

Enter a label for this key.....> **ITSO self-signed cert**

Select desired key size from the following options (512):

- 1: 512
- 2: 1024

Enter the number corresponding to the key size you want: 2

Enter certificate subject name fields in the following.

Common Name (required).....> **wtsc48oe.itso.ibm.com**

Organization (required).....> **IBM**

Organization Unit (optional).....> **ITSO**

City/Locality (optional).....> **RTP**

State/Province (optional).....> **NC**

Country Name (required 2 characters)..> **US**

Enter number of valid days for the certificate 365": 365

Do you want to set the key as the default in your key database? (1 = yes, 0 = no) Y1": 1

Do you want to save the certificate to a file? (1 = yes, 0 = no) Y1": 1

Should the certificate binary data or Base64 encoded ASCII data be saved? (1 = ASCII, 2 = binary) 1": 2

Enter certificate file name or press ENTER for "cert.crt": **itsoself.crt**

Please wait while self-signed certificate is created...

Your request has completed successfully, exit gskkyman? (1 = yes, 0 = no) 0": 0

Key database menu

Current key database is /u/casey/itso.kdb

- 1 - List/Manage keys and certificates
- 2 - List/Manage request keys
- 3 - Create new key pair and certificate request
- 4 - Receive a certificate issued for your request
- 5 - Create a self-signed certificate
- 6 - Store a CA certificate
- 7 - Show the default key
- 8 - Import keys
- 9 - Export keys
- 10 - List all trusted CAs
- 11 - Store encrypted database password

- 0 - Exit program

Enter option number (or press ENTER to return to the parent menu): **10**

Trust CA certificate list

Key database name is /u/casey/itso.kdb

Please choose one of the following keys to work with.

- 1 - ITS0 self-signed cert
- 2 - ITS0 Site Certificate
- 3 - ITS0 Certificate Authority
- 4 - Integrion Certification Authority Root
- 5 - IBM World Registry Certification Authority
- 6 - Thawte Personal Premium CA
- 7 - Thawte Personal Freemail CA
- 8 - Thawte Personal Basic CA
- 9 - Thawte Premium Server CA

Enter a key number or press ENTER for more labels:

The label you enter will be the label you see when you display the list of certificates.

The common name is the fully qualified host name of the TN3270 server. If you select server authentication on the Host On-Demand session properties, the common name must match the host name in the DDNS server for the IP address of the TN3270 server.

Make certificates available to clients

The process of making the server's public certificate available varies with the type of client. The locally installed client obtains the server certificate from the same sources as the download and cached clients, the CustomizedCAs.class file or the Microsoft cryptographic database. However, on a locally installed client the CustomizedCAs.class file must reside on the client itself. There are two methods of updating this file on the client. We recommend the first method.

1. Have the administrator create the CustomizedCAs.class file for the download clients, then distribute it to every locally installed user.
2. Distribute the certificate to every locally installed user and have them run the Certificate Management Utility or the Certificate Wizard to create or update their local copy of the CustomizedCAs.class file. The procedure for doing this is the same as described in "Creating the CustomizedCAs.class file on the server" on page 119.

Downloaded and cached clients must be able to access the certificate from the Host On-Demand server. If the server is using a certificate from a well-known trusted CA, nothing more needs to be done because the certificate is already in the WellKnownTrustedCAs.class file in the "publish" directory. Therefore, it is accessible to the clients.

The Telnet server's certificate issued from an unknown CA, or a self-signed certificate can be made available to the client in one of two ways:

1. If the client is running on a Windows platform you add the certificate to the MSIE browser's keyring. This action is not automatic and must be performed by each user. Refer to 11.4.4, "Add MSIE browser's keyring" on page 448 for the procedures on how to do this.
2. Create a CustomizedCAs.class file, store it on the server, and it will be downloaded to the client.

Creating the CustomizedCAs.class file on the server

If the certificate is self-signed or from an unknown Certificate Authority, you should put it into the CustomizedCAs.class file in the publish directory using the Java keyring utility. The publish directory can be either the Host On-Demand server publish directory or a separate user directory. See Chapter 8, *IBM*

Websphere Host On-Demand Version 7.0 Planning, Installation, and Configuring Host On-Demand, SC31-6301-00. The utility can be issued one of two ways, either with the add option or connect option. We recommend the connect option because it issues a socket connection to the TCP/IP SSL port and verifies it is available and configured correctly.

Before you can issue the Java keyring utility, you need to have the TCP/IP TN3270 Telnet server configured for the SSL port you wish to connect. For TCP/IP profile definitions see “Configuring TCP/IP TN3270 server for SSL” on page 125. Run the Java keyring utility provided with Host On-Demand. This is a lengthy command and it is easy to make errors. The backslash is a continuation character; otherwise the command must be on one continuous line. The command for Java 1.3 is:

```
java -classpath ./usr/lpp/HOD/hostondemand/lib/sm.zip \
com.ibm.hodssligh.tools.keyrng CustomizedCAs connect ipaddr:port
```

where ipaddr is the address of your TN3270 Telnet server and port is the SSL port you wish to connect.

The command for Java 1.1.8 is:

```
java -classpath ./usr/lpp/HOD/hostondemand/lib/sm.zip:$CLASSPATH \
com.ibm.hodssligh.tools.keyrng CustomizedCAs \
connect ipaddr:port
```

Tip: You can create a shell script with the command. This enables you to easily reissue the command if needed. It also allows you to check the syntax of the command before running the script.

You will be prompted to enter the password for CustomizedCAs.class file. You must not give a password; just press Enter, or it will not work. The results of the command will look similar to the Java 1.3 example shown in Example 3-18.

Example 3-18 Java keyring utility output

```
CASEY @ SC48:/usr/lpp/HOD/hostondemand/HOD>javakeyrng
Password for CustomizedCAs.class:
Connecting to 9.12.6.126:6623
com.ibm.hodssligh.SSLException
    at com.ibm.hodssligh.SSLConnection.certificate(SSLConnection.java:979)
    at com.ibm.hodssligh.SSLClient.serverCertificate(SSLClient.java:272)
    at com.ibm.hodssligh.SSLClient.handshake(SSLClient.java:110)
    at
com.ibm.hodssligh.SSLConnection.handleData(SSLConnection.java(Compiled Code))
    at
com.ibm.hodssligh.SSLRecordLayer.receiveRecord(SSLRecordLayer.java:695)
    at com.ibm.hodssligh.SSLConnection.install(SSLConnection.java:212)
    at com.ibm.hodssligh.SSLClient.<init>(SSLClient.java:719)
```

```

        at com.ibm.hodssligh SSLSocket.install(SSLSocket.java:117)
        at com.ibm.hodssligh SSLSocket.<init>(SSLSocket.java:260)
        at com.ibm.hodssligh tools.keyrng.main(keyrng.java)
com.ibm.hodssligh.SSLException
time created=Wed Aug 29 16:52:27 EDT 2002
category=4 TRUSTPOLICY
error=1017 PEERCERTIFICATECHAINNOTTRUSTED
int1 =0
e=null

```

----- Server Certificate Chain -----

Site Certificate - Number 0

```

        Key : RSA/512 bits
        Subject: wtsc48oe.itso.ibm.com, Research Triangle Park, ITS0, IBM, US
        Issuer: ITS0Raleigh, Raleigh, ITS0, IBM, US
        Valid from: Mon Aug 27 10:53:17 EDT 2002
        Valid to: Tue Aug 27 11:03:17 EDT 2003
        Finger print: 2A:84:BA:46:C0:73:7C:4F:6D:98:AD:B1:44:72:BA:F8

```

Enter the number of the certificate to be added to CustomizedCAs.class (q to quit): 0
Adding the Site Certificate - 0 to CustomizedCAs.class
Done.

When executing in a Java 1.3 environment, we found that the flow of execution changes since Java 1.3 classifies certain exception conditions differently from Java 1.1.8. The result is that program flow under Java 1.3 follows a different exception handling path, a path not traversed under Java 1.1.8. In any case, the Java exception shown in Example 3-18 can be ignored. You can verify the certificate was added to the CustomizedCAs.class file with the following command:

```

java -classpath ./usr/lpp/HOD/hostondemand/lib/sm.zip \
com.ibm.hodssligh.tools.keyrng CustomizedCAs verify

```

The add option does not require the TN3270 server to be available, since no socket call is issued. You must specify the name of the certificate file as one of the input parameters. The command when using Java 1.3 is:

```

java -classpath ./usr/lpp/HOD/hostondemand/lib/sm.zip \
com.ibm.hodssligh.tools.keyrng CustomizedCAs \
add --certificatetype certificate.name

```

Where the certificate type is either `ca` if you are adding a CA root certificate or `site` if you are adding a site or self-signed certificate. The `certificate.name` is the fully qualified name of the actual certificate, for instance, `/u/casey/itso.cer`.

The command for Java 1.1.8 is:

```
java -classpath ./usr/lpp/HOD/hostondemand/lib/sm.zip:$CLASSPATH \  
com.ibm.hodsslght.tools.keyrng CustomizedCAs \  
add --certificatetype certificate.name
```

Server authentication

For basic SSL TN3270 server authentication connection, you need to configure a port capable of SSL. Refer to “Configuring TCP/IP TN3270 server for SSL” on page 125 for how to define the TELNETPARMS definitions.

With the instructions provided for creating a key database and requesting and receiving a certificate, you should be able to establish a TN3270 SSL connection for server authentication. In the session properties, select **Yes** for Enabled Security (SSL) and select **Yes** for Server Authentication (SSL) as shown in Figure 3-6.

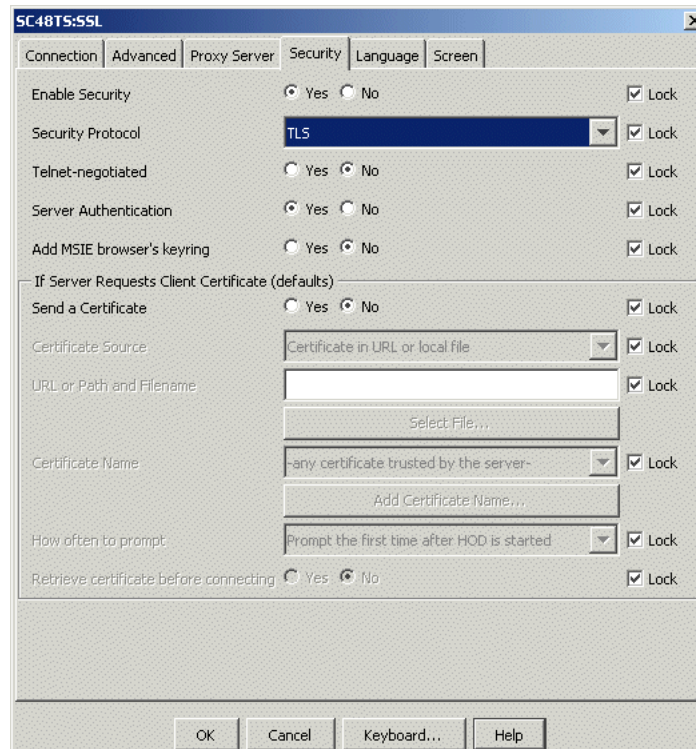


Figure 3-6 Session properties to enable SSL for server authentication

When you start the 3270 session the lock in the bottom right corner of the session window should be locked. If it is not, check the communication code on the bottom of the window. Click the up-arrow next to the message in the OIA to display the Status Bar History. Click ? for further details of the error.

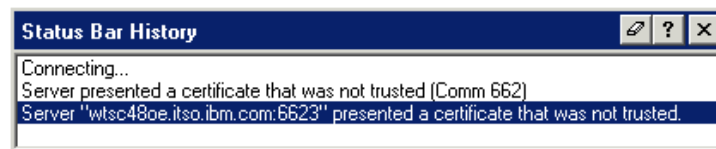
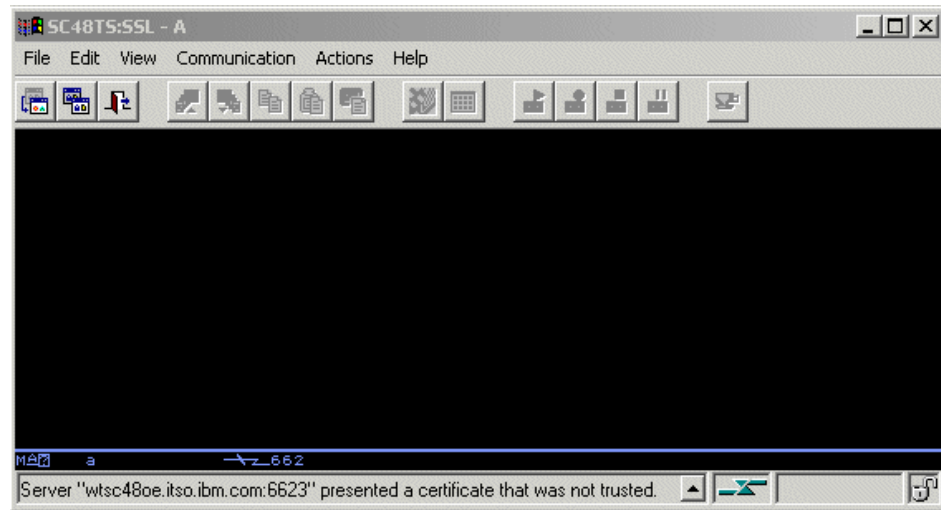


Figure 3-7 3270 session fails to connect, communications error

For this example, the CustomizedCAs.class file did not exist. It was created using the instructions in “Creating the CustomizedCAs.class file on the server” on page 119. In order to download the CustomizedCAs.class file, if using Internet Explorer, press and hold Ctrl, then click **Refresh** on the browser to reload the file; if using Netscape, click **Reload**.

Client authentication

In client authentication the Telnet server requests a certificate from the client to verify who it claims to be. To enable client authentication, server authentication must first be enabled. On the Host On-Demand session properties, several options are available to deploy client authentication. Refer to 11.3.3, “Client authentication” on page 443.

Restriction: Host On-Demand requires a Version 3 type certificate for client authentication. If you have created the certificate using GSKKYMANN on the z/OS you will need to convert the Version 1 PKCS12 created on the z/OS to a Version 3 PKCS12 file. The following lists the steps required to convert a PKCS12 Version 1 file to Version 3.

1. On the TN3270 server, use GSKKYMANN to create a self-signed certificate.
2. Create a PKCS12 format file (select 'Export keys', 'Export keys to a PKCS12 file' from the GSKKYMANN panels). This will create a file with the format filename.p12. If using CLIENTAUTH SAFCERT, use this file as the client certificate source if manually registering the client certificate to the SAF product.
3. FTP the PKCS12 file in binary mode to the client.
4. Use the Host On-Demand certificate management panels to import the PKCS12 file into the Host On-Demand key database.
5. Export the certificate using the Host On-Demand certificate management panels to create a new PKCS12 file. This is the file that Host On-Demand will use to retrieve the client certificate.

On the z/OS TN3270 server, the TCP/IP profile must be configured for client authentication. Refer to "Configuring TCP/IP TN3270 server for SSL" on page 125 for details on configuring the profile.

Configuring TCP/IP TN3270 server for SSL

To configure client authentication in IP services, the TCP/IP profile must be updated. Configure the TCP/IP profile data set, using the CLIENTAUTH statement in the TELNETPARMS block and choosing the security level you want. Client authentication is only supported by OS/390 V2R8 and higher. This is our TCP/IP profile Telnet statements:

Example 3-19 TCP/IP profile Telnet statements

```

; Basic TN3270 Telnet - non-SSL
TELNETPARMS
  PORT 623
ENDTELNETPARMS

; Basic SSL - provides Server Authentication
TELNETPARMS
  SECUREPORT 6623 KEYRING HFS /u/casey/itso.kdb
  CLIENTAUTH NONE
ENDTELNETPARMS

; Client Authentication, without RACF security
```

```

TELNETPARMS
  SECUREPORT 7723 KEYRING HFS /u/casey/itso.kdb
  CLIENTAUTH SSLCERT
ENDTELNETPARMS

; Client Authentication, with RACF security
TELNETPARMS
  SECUREPORT 8823 KEYRING HFS /u/casey/itso.kdb
  CLIENTAUTH SAFCERT
ENDTELNETPARMS

; Client Authentication, with RACF security and SERVAUTH class active
TELNETPARMS
  SECUREPORT 9923 KEYRING HFS /u/casey/itso.kdb
  CLIENTAUTH SAFCERT
ENDTELNETPARMS

BEGINVTAM
  PORT 623 6623 7723 8823 9923

DEFAULTLUS
  TCP48001..TCP48099
ENDDEFAULTLUS

DEFAULTAPPL SC48TS      ; TS0
  LINEMODEAPPL SC48TS
  ALLOWAPPL *

ENDVTAM

```

The Telnet server configuration can be updated dynamically by using the following command:

```
vary tcpip,,obeyfile,dataset.name
```

3.6.4 Certificate management using RACF

RACF can be used to create, register, store, and administer digital certificates and the private keys associated with the certificates. RACF can also be used to create and manage keyrings of stored digital certificates. In this section we describe how to manage your certificates using the RACF commands. Certificates are stored in the RACF database, while private keys may be stored in the ICSF Public Key Data Set (PKDS), encrypted under a 168-bit Triple-DES key.

Using RACF keyrings is the preferred method, because it provides better security for the certificates and their private keys. With RACF keyrings, stash files containing key database passwords are not used and access to keyrings and certificates is controlled by RACF.

RACF distinguishes three types of digital certificates:

1. Certificate Authority certificates: These certificates are associated with Certificate Authorities (CAs) and are used to verify signatures in other certificates.
2. Site certificates: These certificates are associated with servers or network entities in other locations than the local system.
3. User certificates: These certificates are associated with a RACF user ID and are used to authenticate a user's identity.

A user certificate or a certificate that has been connected to a keyring with USAGE(PERSONAL) is the only type of certificate whose private key can be used to create signatures. Therefore, all server certificates for local servers need to be user certificates or they need to be connected to an appropriate keyring with USAGE(PERSONAL).

The step-by-step example described in “Using a CA-signed certificate” on page 127 is generic in nature. It can be used to create a RACF keyring for the IBM HTTP Server for z/OS, the TN3270 server, or other servers that are SSL enabled.

For detailed information about the RACDCERT command refer to the *IBM SecureWay Security Server RACF Command Language Reference* manual for your operating system release.

Using a CA-signed certificate

This section presents the steps required to implement the SSL environment for the Host On-Demand Server. A similar procedure can be used for other SSL-enabled application servers. In this scenario, we use a server certificate signed by a public CA. The steps are:

- ▶ Generate a self-signed certificate
- ▶ Create a certificate request for the CA
- ▶ Store the returned certificate into a data set
- ▶ Store CA certificate for unknown Certificate Authority
- ▶ Replace the self-signed certificate
- ▶ Create a keyring for the server
- ▶ Connect the certificate to the keyring

- Connect the CA certificate to the keyring

Generate a self-signed certificate

We will use this self-signed certificate as a base for the certificate request we will be creating.

```
RACDCERT ID(STC) GENCERT
SUBJECTSDN(CN('wtsc48oe.itso.ibm.com')
O('IBM')
OU('ITSO')
L('RTP')
SP('NC')
C('US'))
WITHLABEL('HOD Server Certificate')
```

Make sure the common name (CN) is the same as the host or domain name of the server. The ID is the RACF defined ID associated with the Host On-Demand server started task.

Create a certificate request for the CA

The certificate request will be stored in an MVS data set with a name like 'CASEY.HODSRV.GENREQ'.

```
RACDCERT ID(STC) GENCERT
GENREQ(LABEL('HOD Server Certificate'))
DSN('CASEY.HODSRV.GENREQ')
```

This certificate request needs to be sent to the Certificate Authority. The format of the request is Base64-encoded text. The data set can be transmitted to a PC with FTP and pasted into the appropriate field in the certificate request. Alternatively, cutting and pasting between a host emulator window and the Web browser can be used.

Store the returned certificate into a data set

The CA usually returns the certificate using e-mail or similar means. The certificate is in Base64-encoded text format. Again, use the same technique as before to copy the certificate into a data set named, for instance, 'CASEY.HODSRV.CERT'.

Note: The data set organization must be variable blocked. If it is fixed blocked, you will receive error IRRD103I: An error was encountered processing the specified input data set. You may need to preallocate the data set as variable blocked prior to transferring the signed certificate.

Store CA certificate for unknown Certificate Authority

If your certificate is signed by an unknown Certificate Authority, you need to store the CA's certificate into the RACF database. We had created our own Certificate Authority to sign the certificates; therefore the CA certificate needed to be stored in the RACF database.

```
RACDCERT CERTAUTH ADD('CASEY.HODSRV.CACERT') TRUST
WITHLABEL('HOD Certificate Authority')
```

Replace the self-signed certificate

Replace the self-signed certificate with the certificate received from and signed by the CA.

```
RACDCERT ID(STC) ADD('CASEY.HODSRV.CERT') TRUST
WITHLABEL('HOD Server Certificate')
```

Create a keyring for the server

This keyring must not already exist for this user. Keyring names become names of RACF profiles in the DIGTRING class, and can contain only characters that are allowed in RACF profile names. Although asterisks are allowed in keyring names, a single asterisk is not allowed.

```
RACDCERT ADDRING(HODSERVER)
```

Connect the certificate to the keyring

Now we can create the connection between the digital certificate and the keyring with the RACDCERT CONNECT command and associate it with the Host On-Demand started task user ID.

```
RACDCERT CONNECT(ID(STC) LABEL('HOD Server Certificate') RING(HODSERVER)
DEFAULT USAGE(PERSONAL))
```

Connect the CA certificate to the keyring

If you had your certificate signed by an unknown Certificate Authority and had to store the CA certificate in the RACF data base, you need to connect the CA certificate to the keyring.

```
RACDCERT CONNECT(CERTAUTH LABEL('HOD Certificate Authority')
RING(HODSERVER) USAGE(CERTAUTH))
```

3.7 Express Logon Feature (ELF)

Express Logon Feature (ELF) was introduced in IBM Communications Server for OS/390 V2R10 IP Services. ELF allows a user on a workstation, with a TN3270 client and an X.509 certificate, to log on to an SNA application without entering a user ID or password.

Express Logon Feature allows users to:

- ▶ Reduce the time administrators spend maintaining user IDs and passwords.
- ▶ Reduce the number of user IDs and passwords that users must remember.
- ▶ Remove a potential security risk of users writing down user IDs and passwords, or sharing them with someone else.

For a complete discussion of Express Logon, refer to 11.8, “Express Logon Feature” on page 455.

Implementation of two-tier network design

The ELF two-tier design implementation is simpler than the three-tier design implementation, as you no longer need a DCAS for a middle-tier TN3270 server. Choose the three-tier design if you do not want to have a TN3270 server on z/OS or if you are going to use Host Publisher. Host Publisher acts as the client and the middle-tier server together.

Follow the steps below to implement ELF with the two-tier design, using Host On-Demand as the client and accessing TSO on z/OS:

1. Define an SSL session with client authentication level 2 (CLIENTAUTH SAFCERT in TCP/IP profile). The procedure to define an SSL session is in “Configuring TCP/IP TN3270 server for SSL” on page 125.
2. Define the EXPRESSLOGON parameter on TCP/IP profile. The ITSO profile statements used for ELF are shown in Example 3-20.

Example 3-20 Profile definition for ELF

```

TELNETPARMS
  SECUREPORT 23003
  KEYRING SAF tcpipa.tn3270.keyring
  CONNTYPE SECURE
  CLIENTAUTH SAFCERT
  EXPRESSLOGON
  DEBUG DETAIL
ENDTELNETPARMS

```

3. Define the PassTicket profile to RACF. For each application to which users are to gain access with a PassTicket, you must define a PTKTDATA class

profile. In our example, the application is TSO and the commands issued are those shown in Example 3-21.

Example 3-21 RACF definition for PassTicket

```

SETROPTS CLASSACT(PTKTDATA)
SETROPTS RACLIST(PTKTDATA)
RDEFINE PTKTDATA TSOSC64 SSIGNON(KEYMASKED(E6C9D30195D4C1E7)) UACC(NONE)
SETR RACLIST(PTKTDATA) REFRESH

```

The profile name (TSOSC64 in our case) must match the application ID configured on the Host On-Demand client window. For TSO, the rule to create a profile name is: TSO+smfid.

Define a key using KEYMASKED, even though the value is not significant. This is required.

For more details about PTKTDATA and rules of profile names, see the *IBM SecureWay Security Server RACF Security Administrator's Guide* manual for your operating system release.

4. Next, start TCP/IP and establish a session with port 23002. Now you are ready to create the ELF macro.

You can display the connection to check that ELF is being used as shown in Example 3-22 and Example 3-23.

Example 3-22 Connection display

```

D TCPIP,TCPIPA,T,CONN
EZZ6064I TELNET CONNECTION DISPLAY 664
      EN                      TSP
CONN  TY IPADDR..PORT      LUNAME  APPLID  PTR LOGMODE
-----
00000F76 4S 9.24.106.91..1347  TCP64002 SC64TS05  TAE D4C32XX3
-----
----- PORT:  23003    ACTIVE          PROF: CURR CONNS:      1
-----
3 OF 3 RECORDS DISPLAYED

```

Example 3-23 Connection display details

```

D TCPIP,TCPIPA,T,CONN,CO=F76
EZZ6065I TELNET CONNECTION DISPLAY 689
CONN: 00000F76          CLNTIP..PORT: 9.24.106.91..1347
LINKNAME: OSA22EOLINK   DESTIP..PORT: 9.12.6.60..23003
HOSTNAME: NO HOSTNAME
CONNECTED: 15:43:14 09/28/2001 STATUS: SESSION ACTIVE
PORT: 23003 ACTIVE SECURE      ACCESS: SECURE 4S  SAFCERT 1
PROTOCOL: TN3270E LOGMODE: D4C32XX3 DEVICETYPE: IBM-3278-3-E
OPTIONS: ETET--- 3270E FUNCTIONS: BSR----

```

```
NEWENV FUNCTIONS: E-
USERIDS
  RESTRICTAPPL: **N/A**  CLIENTAUTH: FASCINI  2
  EXPRESSLOGON: FASCINI  3
APPL: SC64TS05
LUNAME: TCP64002  TYPE: TERMINAL GENERIC
MAPS CONN IDENTIFIER  OBJECT  DEFAPPL  OPTIONS
LU MAPPINGS:
                                     >*DEFLUS* **N/A**  -----
DEFAULTAPPL:
  NL (NULL)  TSO  -----
USS TABLE: **N/A**
INT TABLE: **N/A**
PARMS:
PERS  FUNCT  DIA  SECURE  TIMERS  SMF  MAX  LINE
(LMTQ) (OATSSWH) (DRF) (SCKLECX) (IKPSTS) (ITIT) (RSQ) (BDCTT)
----  -----  ---  -----  -----  ----  ---  -----
----  --TS---  ---  -B--D--  ---STS  ----  RSQ  --C-- *DEFAULT
----  -----  ---  --S----  -----  ----  ---  ----- *TGLOBAL
----  -----  ---  --S----  -----  ----  ---  ----- *TPARMS
----  --TS---  DJ-  SSS-DFX  ---STS  ----  RSQ  --C-- TP-CURR
----  --TS---  DJ-  SSS-DFX  ---STS  ----  RSQ  --C-- FINAL
29 OF 29 RECORDS DISPLAYED
```

- 1 , 2 - CLIENTAUTH level 2 is being used for Express Logon.
- 3 - This is the RACF user ID associated to the client certificate.

Implementation of three-tier network design

The implementation of ELF with three-tier design is the same in both z/OS V1R2 and above and OS/390 V2R10. The following sections describe the implementation.

DCAS - DCAR connection

The Digital Certificate Access Server (DCAS) is a TCP/IP server that runs on OS/390 V2R10 and later. The middle-tier TN3270 servers connect to DCAS using Secure Socket Layers V3 (SSL). The purpose of DCAS is to receive an application ID and a digital certificate from a middle-tier TN3270 server, then ask RACF to return a valid user ID that has been associated with the certificate and to generate a PassTicket for the input user ID and application ID.

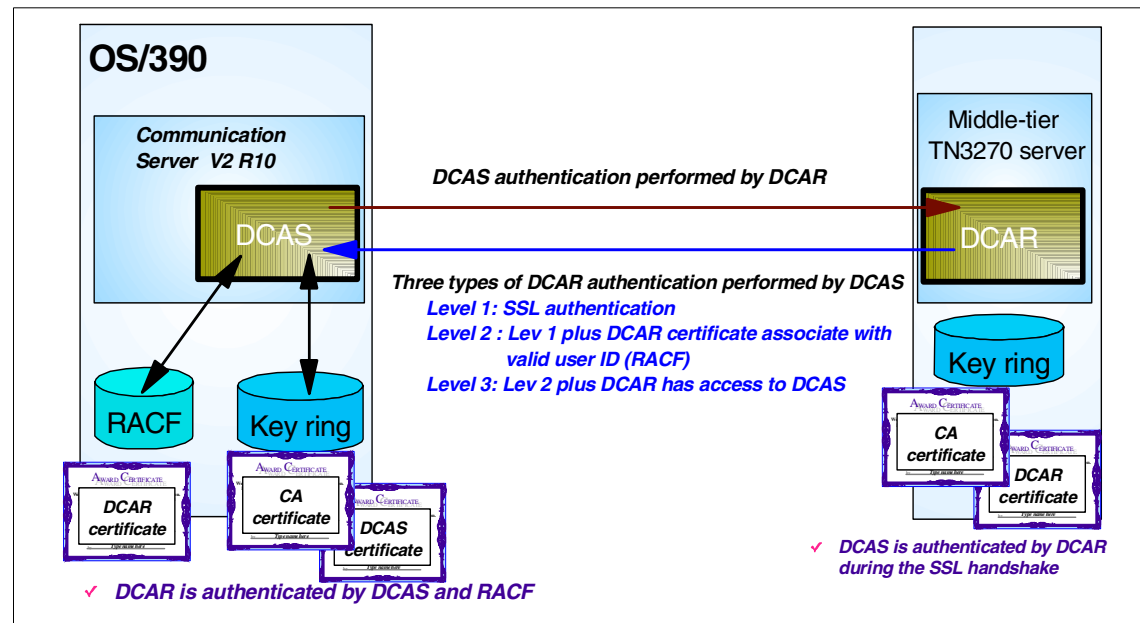


Figure 3-8 DCAS/DCAR

Authenticating the Digital Certificate Access Server

The DCAS authentication is always performed by the Digital Certificate Access Requestor (DCAR) during the SSL handshake. Authentication requires that the DCAS has a private key and an associated X.509 digital certificate defined in a keyring.

If you use a self-signed certificate, it has to be treated as a CA certificate by all TN3270 servers. Follow the steps below:

1. Export the DCAS self-signed certificate into a file in the DER binary format.
2. Send it to a TN3270 server, using FTP with the BINARY send option.
3. Store the certificate into a key database for the TN3270 server as a trusted Certificate Authority.

Authenticating the Digital Certificate Access Requestor

The DCAR is the client that interacts with the DCAS. Authenticating the DCAR involves additional levels of control in which the client must have a key database with a certificate. Depending on the control level, the certificate is authenticated by SSL and the DCAS using RACF services.

There are three levels of client authentication from which to choose:

- Level 1

With Level 1 authentication, the DCAS uses the client authentication provided by SSL at the time of the SSL handshake. The keyring used by the DCAS must contain the following certificates:

- The DCAS certificate
- The certificate of a CA that has signed the TN3270 server certificate. Or the TN3270 certificate itself, if a self-signed certificate is used for the TN3270 server.

To configure DCAS for this level of authentication, specify the CLIENTAUTH LOCAL1 keyword in the DCAS configuration file. Use the KEYRING or the SAFKEYRING keywords in the DCAS configuration file to specify the keyring used by the DCAS.

► Level 2

Level 2 includes Level 1 authentication plus additional verification that the DCAR certificate has been associated in RACF with a valid user ID. To configure DCAS for this level of authentication, specify the CLIENTAUTH LOCAL2 keyword in the DCAS configuration file. Use FTP (with the BINARY send option) to send the client's DER certificate to an MVS data set. Use the RACDCERT ADD command to add the certificate to RACF and associate it with a user ID, as shown in the following example:

```
RACDCERT ID(dcasid) ASID('DCAS.DCAR.CERT') TRUST
```

► Level 3

Level 3 includes level 2 authentication plus it verifies that the DCAR has access to the DCAS. The user ID derived from the certificate using the RACF checks from Level 2 is defined as having access to the SERVAUTH RACF class and the EZA.DCAS.cvtsysname resource in the SERVAUTH class. The following conditions apply:

- if the SERVAUTH class is not active or the EZA.DCAS.cvtsysname profile is not defined, or both, it is assumed this enhanced level is not requested.
- If the SERVAUTH class is active and the EZA.DCAS.cvtsysname profile is defined (but not for the user associated with the certificate) the requestor's connection is terminated.

Use the commands below to create the RACF profile and give the access permission to a user:

```
RDEFINE SERVAUTH EZA.DCAS.cvtsysname UACC(NONE)
PERMIT EZA.DCAS.cvtsysname CLASS(SERVAUTH) ACCESS (CONTROL) ID(dcasid)
SETR RACLIST(SERVAUTH) REFRESH
```

To configure DCAS for Level 3 authentication, follow these steps:

- Specify the CLIENTAUTH LOCAL2 keyword and value in the DCAS configuration file.

- Activate the SERVAUTH RACF class.
- Define a profile for the EZA.DCAS.cvtsysname resource and associate the profile with the user ID associated with the certificate.

Note: The ID associated with the certificate and the EZA.DCAS.cvtsysname can be any valid user ID.

DCAS customization

Follow the steps below to customize DCAS for ELF with three-tier design:

1. Define an SSL session between DCAS (z/OS) and DCAR (middle-tier Telnet Server) and between DCAR and TN3270 client (Host On-Demand in our case). The procedure to define an SSL session is in “Configuring TCP/IP TN3270 server for SSL” on page 125. Instead of using the TCP/IP name, use the DCAS name on RACF commands.

The user ID associated with the keyring and the DCAS server’s certificate has to be the user defined in the STARTED procedure of DCAS.

The DCAS certificate has to be imported into the key database used by the TN3270 server (middle-tier) and defined as trusted.

The Host On-Demand client certificate has to be defined in RACF.

2. Set up DCAS to use RACF services.

- Define started profile and OPERCMDS

```
ADDUSER DCAS DFLTGRP(OMVSGRP) OMVS(UID(0) HOME('/'))
```

```
RDEFINE STARTED DCAS.* STDATA(USER(DCAS))
SETROPTS RACLIST (STARTED) REFRESH
```

```
RDEFINE OPERCMDS(MVS.SERVGR.DCAS) UACC(NONE)
PERMIT MVS.SERVGR.DCAS CLASS(OPERCMDS) ACCESS(CONTROL) ID(DCAS)
SETROPTS RACLIST(OPERCMDS) REFRESH
```

- Permit the DCAS to use certificate services

```
SETROPTS CLASSACT(DIGTCERT DIGTRING)
RDEFINE FACILITY(IRR.DIGTCERT.LIST) UACC(NONE)
RDEFINE FACILITY(IRR.DIGTCERT.LISTRING) UACC(NONE)
PERMIT IRR.DIGTCERT.LIST CLASS(FACILITY) ID(DCAS) ACCESS(CONTROL)
PERMIT IRR.DIGTCERT.LISTRING CLASS(FACILITY) ID(DCAS) ACCESS(CONTROL)
SETROPTS RACLIST(DIGTRING DIGTCERT) REFRESH
```

- Define PassTicket data profile

```
SETROPTS CLASSACT(PKTCDATA)
SETROPTS RACLIST(PKTCDATA)
RDEFINE PKTCDATA TSORA03 SSIGNON(KEYMASKED(E6C9D30195D4C1E7))
UACC(NONE)
SETROPTS RACLIST(PKTCDATA) REFRESH
```

3. Define DCAS configuration file.

Some of the configuration parameters you can use in the DCAS configuration file are shown in Table 3-3.

Table 3-3 DCAS configuration parameters

Parameters	Description
IPADDR	Allows you to define the IP address to which the DCAS will bind.
PORT	Defines the port number on which DCAS will run.
KEYRING ¹	Defines the HFS key database file containing the certificate to be used during the SSL handshake.
STASHFILE	Specifies the password file to the associate key database file.
SAFKEYRING ¹	Defines the RACF-defined keyring containing the certificate to be used during the SSL handshake.
V3CIPHER	Specifies a subset of the supported SSL V3 cipher algorithms.

¹ - The keywords KEYRING and SAFKEYRING are mutually exclusive.

Here is a sample DCAS configuration file that was used in DCAS startup procedure in next step:

```
TCPIP TCPIPB
PORT 8990
CLIENTAUTH LOCAL2
SAFKEYRING r2617.mvs28b.dcas.keyring 1
# KEYRING /etc/dcas/dcas.kdb
# STASHFILE /etc/dcas/dcas.sth
```

¹ - The keyring name is case sensitive

4. Start DCAS

Following is a sample procedure for DCAS. It is also provided in hlq.SEZAINST(EZADCASP):

```
//DCAS PROC
//DCAS EXEC PGM=EZADCDMN,REGION=4M,TIME=NOLIMIT,
// PARM=('POSIX(ON) ALL31(ON)',
// 'ENVAR("_CEE_ENVFILE=DD:STDENV" ',
// '"DCAS_CONFIG_FILE=/etc/dcas.r2617.conf")/-d 3') 1
//CEEDUMP DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
```

```
//SYSERR DD SYSOUT=*
//SYSOUT DD SYSOUT=*
//STDENV DD DSN=TCPIP.TCPPARMS.R2617(DCASENV),DISP=SHR
```

1 - The DCAS configuration file is /etc/dcas.r2617.conf

By default DCAS writes the messages in the /tmp/dcas.log file. You can set the debug level in the -d start option.

When DCAS is started as an MVS started procedure, the following messages will show up in the MVS console:

```
S DCAS
$HASP100 DCAS ON STCINRDR
IEF695I START DCAS WITH JOBNAME DCAS IS ASSIGNED TO USER
TCPIP3 , GROUP OMVSGRP
$HASP373 DCAS STARTED
IEF403I DCAS - STARTED - TIME=20.36.55
EZZ8601I DCAS IS STARTING
EZZ8620I DCAS SECURITY SERVER SERVAUTH CLASS IS ACTIVE
EZZ8624I DCAS PROCESSING CONFIGURATION FILE /ETC/DCAS.R2617.CONF
EZZ8625I DCAS CONFIGURATION FILE PROCESSING IS COMPLETE
EZZ8618I DCAS LISTENING ON SECURE PORT 8990
```

5. Define a Host On-Demand TN3270 session and create a macro for Express Logon. Refer to 11.8.2, “Record the macro” on page 457 for procedures for Host On-Demand or “Recording the ELF macro” on page 894 for Personal Communications Version 5.6 clients. The client setup is identical for two-tier and three-tier designs.

For more information about the ELF implementation with OS/390 V2R10 using the three-tier network design, refer to the following documents:

- ▶ *IBM Communications Server IP Configuration Guide* for your operating system release
- ▶ *IBM Communications Server for OS/390 TCP/IP 2000 Update Technical Presentation Guide*, SG24-6162

3.8 LDAP directory server

The OS/390 Lightweight Directory Access Protocol (LDAP) server is part of the SecureWay Security Server for OS/390 or z/OS. This server may be the LDAP Server for one or more Host On-Demand systems on any platform. The OS/390 LDAP server is configured in one of three modes: RDBM, SDBM or TDBM. Host On-Demand only works when the LDAP server is configured with a DB2 back-end database, RDBM or TDBM. TDBM implementation, introduced with

OS/390 Release 10, is the recommended implementation because it has an improved schema that includes the HOD required schema, and it uses a DB2 database that was designed for performance. For detailed information about the OS/390 LDAP Server, refer to the following:

- ▶ *OS/390 SecureWay Security Server LDAP Client Application Development Guide and Reference*, SC24-5878
- ▶ *IBM SecureWay Security Server LDAP Server Administration and Usage Guide* for your operating system release.

If you have an existing OS/390 Security Server, verify the following:

- ▶ The server is configured as described in the appropriate *SecureWay Security Server LDAP Server Administration and Usage Guide*.
- ▶ A suffix has been added and associated with an object class.

The remainder of this section focuses on the configuration of the OS/390 LDAP directory server. For information about using the LDAP server, refer to Chapter 8, “LDAP directory server” on page 381.

3.8.1 Schema installation

In order to use Host On-Demand with your existing LDAP directory you will need to use the IBM schema shipped with the LDAP server. The process for setting up this schema is different for RDBM and TDBM.

Schema set up using RDBM backend

To add the IBM schema to the existing LDAP directory:

- a. Edit the LDAP directory configuration file, `slapd.conf`, and modify the include statements as follows:

Original	IBM schema
<code>slapd.at.system</code>	<code>schema.system.at</code>
<code>slapd.cb.at.conf</code>	<code>schema.IBM.at</code>
<code>slapd.at.conf</code>	<code>schema.user.at</code>
<code>slapd.oc.system</code>	<code>schema.system.oc</code>
<code>slapd.cb.oc.conf</code>	<code>schema.IBM.oc</code>
<code>slapd.oc.conf</code>	<code>schema.user.oc</code>

- b. Restart the LDAP directory server.

See the program directory for further information regarding DB2 tables.

Schema set up using TDBM backend

To add the IBM schema to the existing LDAP directory:

- a. Use the **ldapmodify** command to add the schema.user.ldif that is shipped with the LDAP server.
- b. Use the **ldapmodify** command to add the schema.IBM.ldif that is shipped with the LDAP server.

3.8.2 Directory Tree

To configure an existing LDAP directory for Host On-Demand, familiarize yourself with the LDAP directory. Decide how Host On-Demand will fit into your network and organizational structure, and then design the LDAP directory information tree. For example:

- ▶ To build a directory information tree for an entire organization, use the organization object class for the suffix:

```
dn: o=MyOrganization
objectclass: organization
o: MyOrganization
```
- ▶ To build a directory information tree for one division of an organization, use the organizationalUnit object class for the suffix:

```
dn: ou=MyDivision, o=MyOrganization
objectclass: organizationalUnit
ou: MyDivision
```

The directory information tree should be defined in an LDAP Data Interchange Format (LDIF) file. Examples of the directory information tree can be found in /usr/lpp/ldap/examples/sample_server/sample.ldif. We created a file called its0.ldif with the following:

```
dn: ou=ITS0, o=IBM
objectclass: organizationalUnit
ou: ITS0
```

To add to the directory information tree in the LDAP directory, use the **ldapadd** command. Details on using this command can be found in the *IBM SecureWay Security Server LDAP Client Application Development Guide and Reference*, SC24-5878.

3.8.3 Performance considerations

When using RDBM the following configuration changes are offered as possible performance enhancements that can be added to slapd.conf:

```
index pr nc palPtr eq
index dc eq
index o eq
index name eq
```

```
index objectClass eq
index u d eq
```

Sizelimit is a parameter in the slapd.conf. This is the number of entries the LDAP directory server will return on a search request; change the sizelimit to 5000 (this applies when using either RDBM or TDBM).

3.9 Native Authentication

The Native Authentication code runs as a separate executable module called HODRAPD, which is invoked using the hodrapd.sh shell script. The HODRAPD module is installed during SMP/E CALLLIBS processing and it is automatically link-edited during the JCLIN CALLLIBS processing in the APPLY process.

When the Native Platform Authentication Service is started from UNIX System Services, the HODRAPD module is executed from SYS1.LINKLIB or your alternate LINKLIB data set. If you choose to move the HODRAPD module to an alternate LINKLIB data set, that data set must be accessed by the system LNKLIST or LPALIB.

In order to use the Native Authentication service, Host On-Demand must enable an LDAP directory for the storage of preferences.

3.9.1 Installation of Native Authentication service

During installation of Host On-Demand V7, the hod70mvs.sh shell script not only untars the Host On-Demand V7product, it also creates the necessary link so that when the user starts Native Authentication with hodrapd.sh shell script, the HODRAPD load module is executed. If the link to HODRAPD gets unlinked, the statements below can be used to restore the link.

Example 3-24 Restore link for HODRAPD

```
export HOD_DIR=/usr/lpp/HOD
touch $HOD_DIR/hostondemand/private/HODRAPD
ln -s $HOD_DIR/hostondemand/private/HODRAPD (continued on next line)
$HOD_DIR/hostondemand/private/hodrapd
chmod 744 $HOD_DIR/hostondemand/private/HODRAPD
chmod +t $HOD_DIR/hostondemand/private/HODRAPD
```

The Native Authentication code logs its messages to the syslog, which may need to be configured to log the desired level of messages. The HODRAPD module writes its messages to the user.* entry in the syslog.conf file.

3.9.2 Starting Native Authentication service

To start the Native Authentication code, run the hodrapd.sh shell script (located in the /usr/lpp/HOD directory). The shell script may need to be edited if you installed Host On-Demand in a directory path other than /usr/lpp/HOD.

The shell script also has options that can be set. Options such as logging, time-out values, and maximum number of requests the server will allow can be specified when you start the service. You must keep the -x option, but can append any of the following options. Edit the line where the HODRAPD module is called and append the following options if desired:

- l** Enable logging (for example -xl)
- t** Set socket timeout value, in seconds, default is 20 (for example -xt100)
- c** Set the max number of requests the server will allow (for example -xc100)

The shell script must be started by a user with root authority.

HODRAPD can be started from the OMVS shell or as a started task. To start HODRAPD from the OMVS shell go your Host On-Demand install directory and run the shell script. For example:

```
cd /usr/lpp/HOD
hodrapd.sh
```

The following is a sample procedure we used to start HODRAPD:

Example 3-25 Sample HODRAPD started procedure

```
//HODRAPD PROC
//* HOST ON DEMAND VERSION 7
//HODSRVG EXEC PGM=BPXBATCH,REGION=OM,TIME=NOLIMIT,
//      PARM='sh /usr/lpp/HOD/hodrapd.sh'
//SYSPRINT DD SYSOUT=A
//SYSERR DD SYSOUT=A
//SYSOUT DD SYSOUT=A
//STDENV DD DSN=TCPIPOE.SC48.TCPPARMS(HODENV),DISP=SHR
//SYSIN DD DUMMY
//STDOUT DD PATH='/tmp/HODRAPD.stdout',
//      PATHOPTS=(OWRONLY,OCREAT,OTRUNC),
//      PATHMODE=SIRWXU
//STDERR DD PATH='/tmp/HODRAPD.stderr',
//      PATHOPTS=(OWRONLY,OCREAT,OTRUNC),
//      PATHMODE=SIRWXU
```

Because our system had more than one TCP/IP stack, we had to make sure the HODRAPD task established affinity to the correct stack. We set the following environment variable in the data set member pointed to by DD name STDENV:

```
_BPXK_SETIBMOPT_TRANSPORT=TCPIPOE
```

When the Native Authentication service is completely started, a message is displayed on the z/OS console unless you have logging enabled. Then it is displayed in the syslog. The message reads:

```
PAS0001 Starting IBM Platform Authentication Service.
```

You can check the status of the Native Authentication service by issuing the **netstat conn** command from TSO or **netstat -a** from USS (UNIX System Services). The command will display the status of the service. The Native Authentication service uses the well-known port 2569. The **netstat** command will display the following status based on the user ID for the Native Authentication service:

MVS TCP/IP	onetstat CS V2R10	TCPIP Name: TCPIPOE	11:03:39
User Id	Conn Local Socket	Foreign Socket	State
-----	-----	-----	-----
HODRAPD2	00005CCE 0.0.0.0..2569	0.0.0.0..0	Listen

If the port is not in listening status, verify the permissions of HODRAPD in the private directory. The file must have the sticky bit turned on as shown below:

```
-rwxr--r-T 1 AAAAAA SYS1 0 Dec 8 2000 HODRAPD
```

If the sticky bit (denoted by the T) is not set, use the **chmod** commands shown in Example 3-24 on page 140 to set the bit.

When the Native Authentication service is started, two UNIX System Services processes are started as shown below:

Example 3-26 HODRAPD process IDs

UID	PID	PPID	C	STIME	TTY	TIME	CMD
AAAAAAA	50332132	33554917	-	10:34:41	?	0:00	hodrapd -xl
AAAAAAA	33554917		1	10:34:39	?	0:00	hodrapd -xl

The hodrapd.sh shell script appends to your LIBPATH and NSLPATH. You may want to append these statements to your /etc/profile in USS. For example:

```
export NLSPATH=$NLSPATH:/usr/lpp/HOD/hostondemand/lib/messages/%N
export LIBPATH=$LIBPATH:/usr/lpp/HOD/hostondemand/lib/
```

3.9.3 Testing Native Authentication service

In order to verify that the Native Authentication service is working correctly perform the following steps:

1. Log on to the Host On-Demand administrator and insure that the LDAP directory is enabled as shown in Figure 3-9.

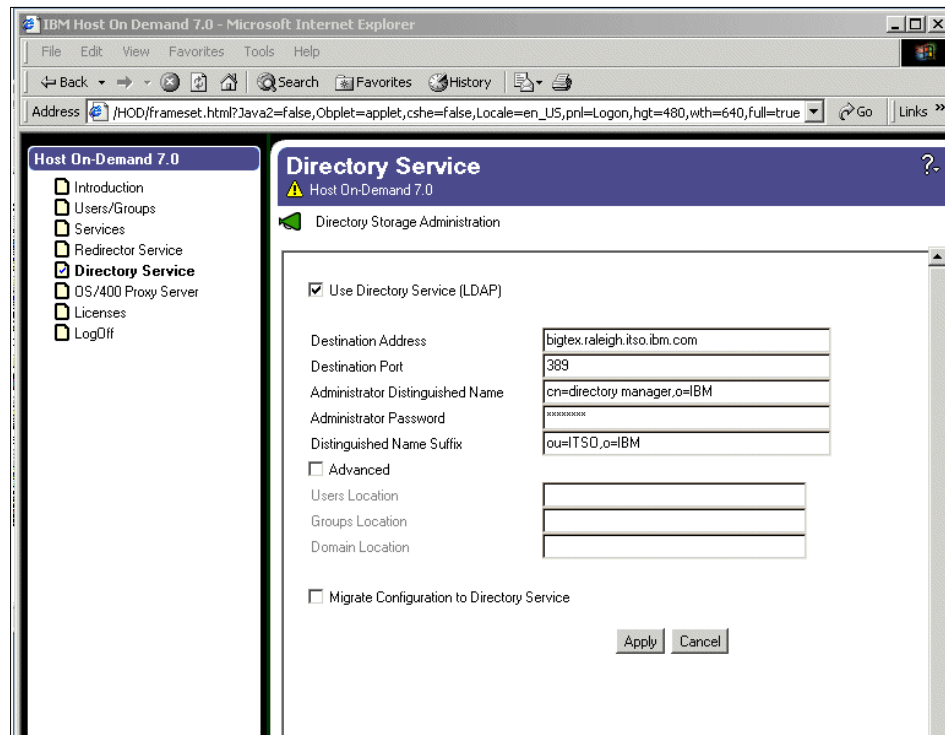


Figure 3-9 Using Directory Service

2. Verify that Native Authentication is enabled for at least one user by selecting **Use Native Authentication** and providing a native user ID as shown below:
3. Click **OK**.
4. Log off the Host On-Demand administrator.
5. Download a Host On-Demand client, and log onto the ID using the password of the Native user ID that you specified. Using the example above, the Host On-Demand user ID is CASEYTEST and the password is the RACF password of native ID CASEY.

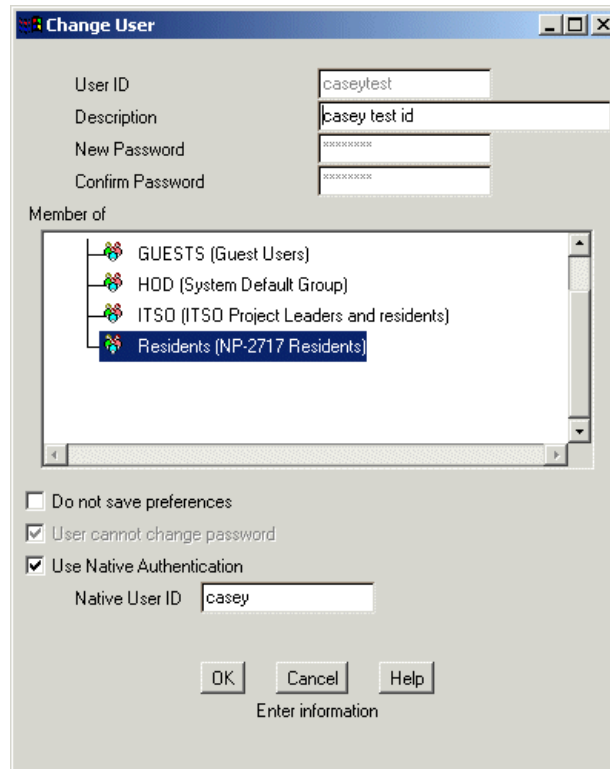


Figure 3-10 Enabling Native Authentication

Problem determination

If you receive an invalid logon (Figure 3-11) and you know without a doubt that the user ID and password is correct, verify you have defined localhost in the /etc/hosts file. You must be able to resolve localhost. If necessary, add an entry for localhost as follows in the /etc/hosts file:

```
127.0.0.1    localhost
```

If you make a modification to the /etc/hosts file, you must recycle the Host On-Demand server. The HODRAPD task does not need to be recycled.



Figure 3-11 Native Authentication logon failure

Stopping the Native Authentication service

When the service is started, two UNIX System Services processes are started as described in 3.9.2, “Starting Native Authentication service” on page 141. The service can be stopped in one of two ways. Both ways require that you determine the PID of the first process that is started. From the OMVS shell, issue the **ps -ef** command or from the MVS console issue **d omvs,a=a11** and find the two processes. Using Example 3-26 on page 142, the HODRAPD service can be stopped with either of the following commands

- ▶ From OMVS shell:
`kill -9 33554917`
- ▶ From the MVS console:
`f bpxoinit,term=33554917`

4



iSeries tips

In this chapter we cover some of the top OS/400-related tips and techniques from the Host Access Call Center Team, including:

- ▶ The iSeries as a Host On-Demand server
- ▶ Performance tips
- ▶ Other iSeries tips

4.1 Upgrade JVM level to 1.3

Some modest performance gains can be obtained by using the most advanced Java release level. However, some applications are not compatible with this level yet. OS/400 can support multiple JVM levels.

1. As new JVM levels are announced, they will become available via PTFs. Refer to <http://www-912.ibm.com>. You may also want to obtain the latest Java group PTF and cumulative service.

Table 4-1 Current OS/400 PTFs

OS/400 level	Java Group PTF	JVM 1.3 PTF
V4R4	SF99067	SF63322
V4R5	SF99068	SF63319
V5R1	SF99069	*
V5R2	SF99169	*
* = included with base CD set for OS/400		

2. Install the JVM. Even though the CD is distributed as a PTF, the installation instructions will instruct you to use the RSTLICPGM command.
3. Apply the Cumulative Service CDs via option 8 on the GO PTF menu.
4. Apply the Java Group PTF CD via option 8 on the GO PTF menu.
5. Adjust Host On-Demand to use JVM 1.3 the next time it is started:
 - CFGHODSVM
 - Page down
 - Add the property (java.version 1.3) to the Java options as shown in Figure 4-1 on page 149.
 - Press Enter.
6. Restart the Host On-Demand Service Manager:
 - ENDHODSVM
 - STRHODSVM
7. Validate that JVM 1.3 is being used:
 - WRKJOB QHODSVM
 - Choose the active job
 - Option 4 to view printouts for the job
 - Option 5 to view the printout

- The message “Finding native method library: QJAVA QJVI013” indicates that the 1.3 JVM is being used.

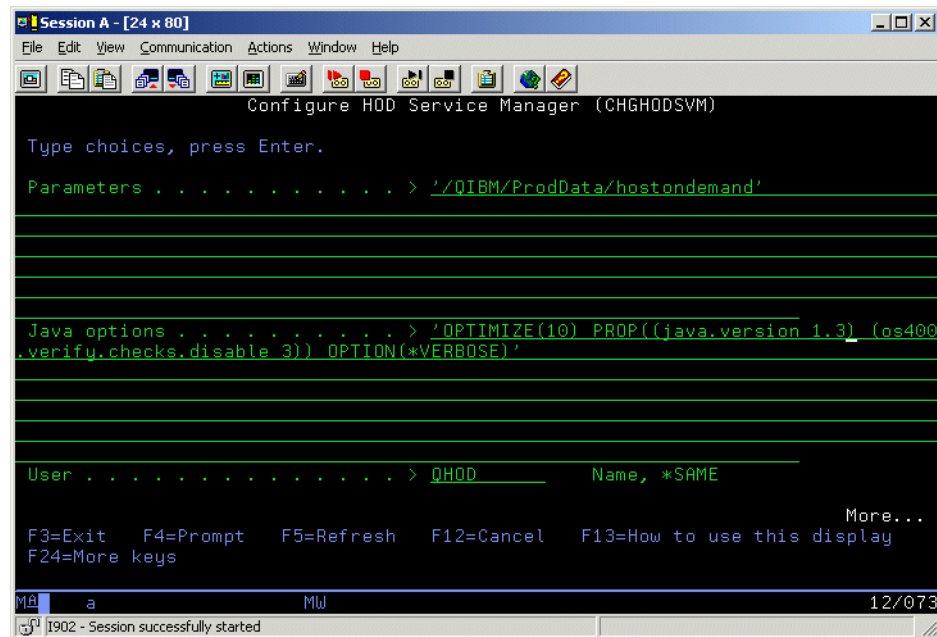


Figure 4-1 Switch JVM to 1.3

4.2 Using IBM HTTP Server (Powered by Apache)

The trend for OS/400 is to switch from the 5769-DG1 HTTP server to the Apache server. Customers may have switched their default Web instances and would like to handle Host On-Demand with Apache.

The following describes how to add the “hod” directive to an existing Apache Web instance.

For the latest information on the HTTP server (powered by Apache), refer to <http://www.ibm.com/servers/eserver/series/software/http/services/apache.htm>

1. From a browser, start the main Web page for your iSeries:
 http://<system.name>:2001/HTTPAdmin (if you are using V5R1 or V5R2)
 http://<system.name>:2002/HTTPAdmin (if you are using V4R5)
2. Click **Manage HTTP Servers** under General Server Administration. See Figure 4-2.

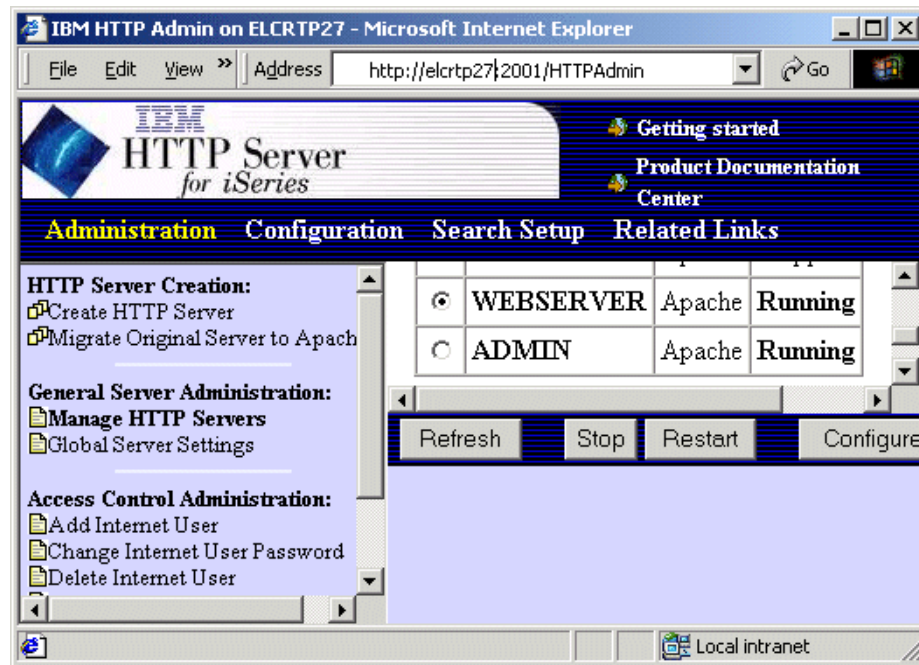


Figure 4-2 Starting the iSeries Apache Configuration tool

3. Select the instance that you want to update. In this case, we are updating the WEBSERVER Apache instance (root directory = /www/webserver).
4. The configuration window is shown for the WEBSERVER instance. Click the **Aliases and Redirection** link.

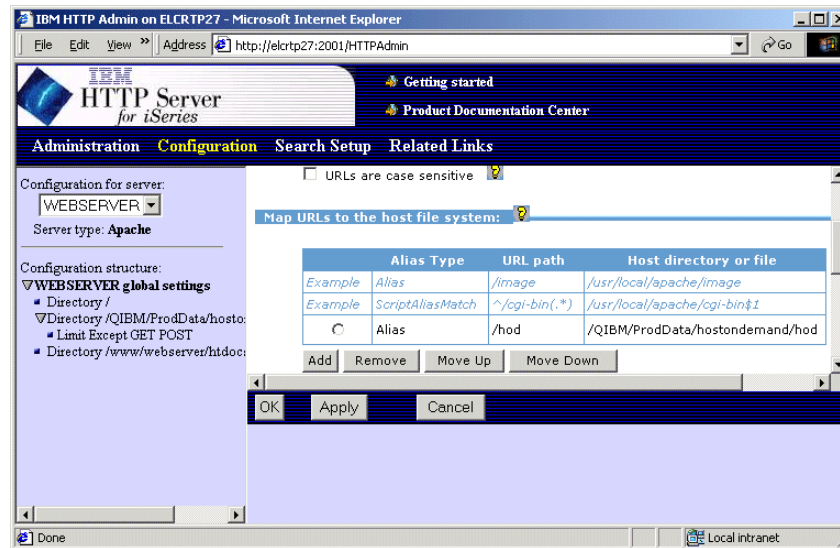


Figure 4-3 Adding the /hod alias

5. Click **Add** in the Map URLs to the host file system section. Then select **Alias**. Set the URL path to /hod and the Host directory to /QIBM/ProdData/hostondemand/hod. Finally, click **OK**.
6. Click **Edit Configuration File** (located in the bottom right). Add the following text lines to the configuration just before the <Directory /> line.

```
<Directory /QIBM/ProdData/hostondemand/hod>
  <LimitExcept GET POST>
    order deny,allow
    deny from none
  </LimitExcept>
  AllowOverride None
  UseCanonicalName Off
  HostNameLookups off
  Options +FollowSymLinks
</Directory>
```

7. Click **OK**. You will return back to the instance configuration window.
8. Click **Restart**. In a few moments, you should be able to bring up the http://as400/hod/hodmain.html window.
9. An interesting option that you may want to consider is to require the user to sign on to the HTTP server. This can be accomplished by replacing the information entered in step 6 above with the following:

```
<Directory /QIBM/ProdData/hostondemand/hod>
```

```
PasswdFile %%SYSTEM%%
AuthType Basic
UserID %%CLIENT%%
AuthName usr
require valid-user
<LimitExcept GET POST>
Order allow,deny
Allow from none
</LimitExcept>
AllowOverride None
UseCanonicalName Off
HostNameLookups off
Options +FollowSymLinks
</Directory>
```

4.3 Using Lotus Domino HTTP Server

Some customers have a Domino HTTP server as their default server or they may want users to authenticate to their Domino HTTP server before any Web pages are served.

The following instructions were tested on Lotus Notes for iSeries Release 5.04. Refer to the *Getting Started with Lotus Notes for iSeries 5.07* manual, which will guide you in configuring your server so that a basic Web page can be served.

This example assumes that you have previously installed the Domino Administrator tool on a PC. On the PC, perform the following commands:

1. Click **Start -> Programs -> Lotus Applications -> Lotus Domino Administrator**.
2. Click **File -> Tools -> User id** (choose the user ID for the Domino administrator).
3. Click **File -> Open server -> <specify server name> -> OK**.
4. A window similar to Figure 4-4 will be shown.

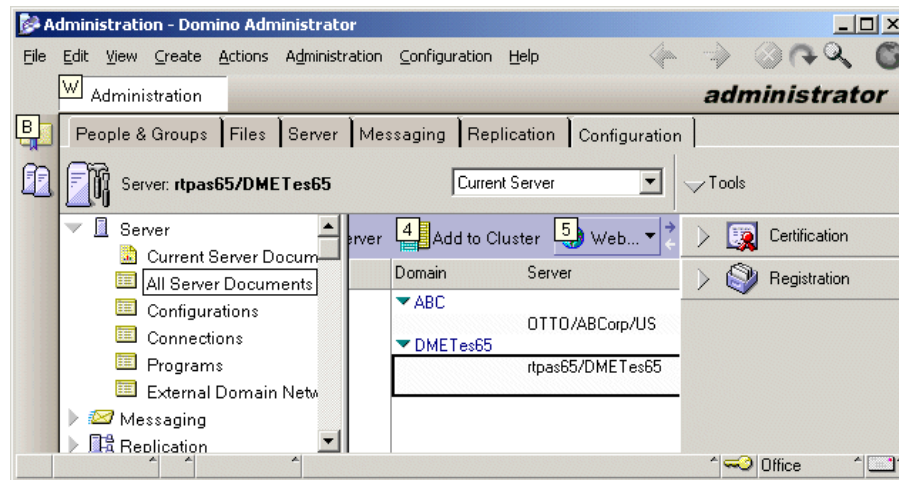


Figure 4-4 Starting the Domino Administration tool

5. Click the **Configuration** tab.
6. Open the server documents.
7. Click **All Server Documents**.
8. Find the server you wish to update. In the example below, we chose the DMETes65 system.
9. Click the **Web...** action on the menu bar (its icon is a globe symbol). Note that you may have to use the (->) left arrow on the action bar to view the icon if your screen is limited in size.
10. Click **URL mapping**
11. Leave the settings for the Basics and Site Information fields blank (so that the information applies to all Virtual Servers). Click the **Mapping** tab. A window similar to Figure 4-5 will be displayed.

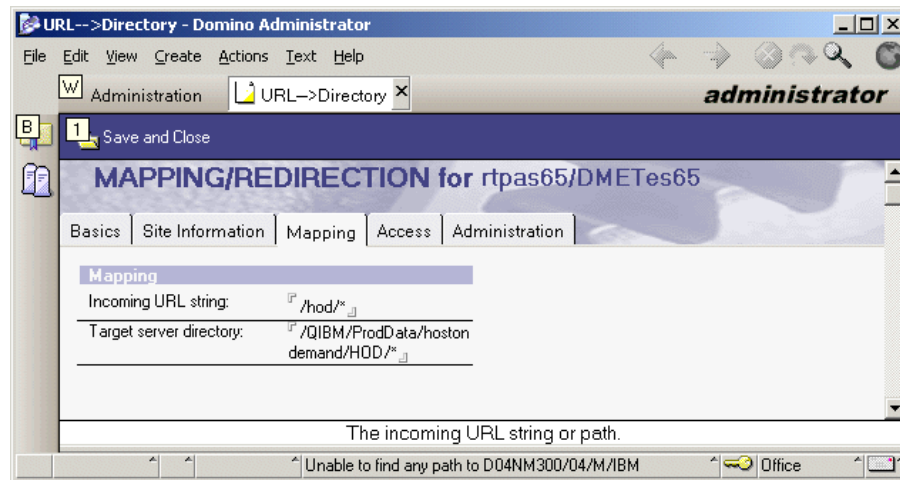


Figure 4-5 Specify the map to the Host On-Demand directory

12. Type /hod/* for the Incoming URL string (see Figure 4-5).
13. Type /QIBM/ProdData/hostondemand/hod/* for the Target server directory field.
14. Click **Save** and **Close**.

4.3.1 Restarting the Domino HTTP Server

The Web server must be restarted before the new directive becomes effective. The following procedure will cause a quick restart.

1. Click the **Server** tab.
2. Locate the HTTP Web Server task as shown in Figure 4-6.

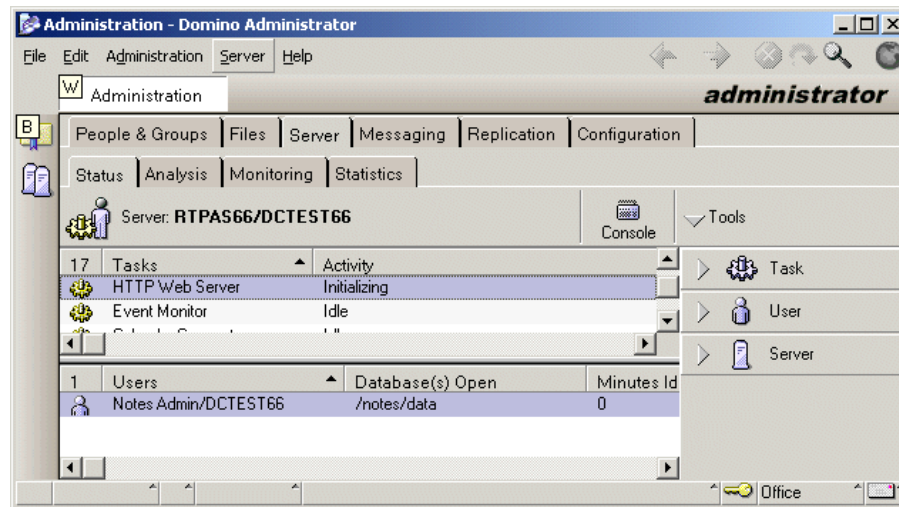


Figure 4-6 Restarting the Domino HTTP server

3. Click the **Server** menu option on the top bar.
4. Click **Task -> Tell -> Restart HTTP server -> Clear cache**.

4.3.2 Using the Domino HTTP Server and Host On-Demand

The typical Host On-Demand Web pages should serve in a manner similar to the DG1 product. For example, <http://rtpas65/hod/hodmain.html>.

4.4 Using the Configuration Servlet

An installation script has been provided to install the Host On-Demand Configuration Servlet. Additional information on the Host On-Demand Configuration Servlet can be found in Chapter 9, "Configuration Servlet" on page 397.

Installation notes:

- ▶ WebSphere V4.0 or V3.5 must be installed and the subsystem must be active. Advanced, Standard or Advanced Edition Single Server versions are supported.
- ▶ WebSphere security must be temporarily disabled.
- ▶ To install:
 - a. Run **qsh**

- b. Enter `cd /qibm/proddata/hostondemand/lib/samples/HodServlet`
 - c. Run `CfgHodServlet-OS400.sh`
 - **Note:** You may optionally specify a WebSphere instance using the `-instance xxx` parameter. If the parameter is not specified, the instance is assumed to be default.
 - The installation script may run for a minute or two. The installation program is complete when a dollar sign (\$) is shown.
 - d. Exit from qsh by pressing F3.
 - e. Run `EDTF '/QIBM/ProdData/hostondemand/hod/config.properties'`
 - f. Add the following line:


```
ConfigServerURL=/HODServlet/HODServlet/hod
```
 - g. Press Enter, then press F3 to update the config.properties file.
 - h. Start the Websphere Application Server instance . Refer to the Websphere Getting Started Manual for additional details.
 - i. Restart the Host On-Demand service manager.
 - ENDHODSVM
 - STRHODSVM
- being careful that you enter /HOD in uppercase.
- ▶ Attempt to use the Web page. `http://my400/HOD/hodmain.html`.
Be very careful that you enter /HOD in uppercase
 - ▶ For faster execution, consider the method discussed in 4.7.2, "Compile Host On-Demand for faster execution" on page 159.

4.5 Screen Customizer and 5250 subfiles

By default, Screen Customizer will use the "GUI version" for OS/400 commands, if they are available. To have Screen Customizer display the screen in the same manner as Host On-Demand, use the following procedure:

1. Enter `WRKLNK '/QIBM/ProdData/hostondemand/HOD/hod*.html'`
 - Use option 2 to edit each entry.
 - Add the following lines to each Host On-Demand Web page:
 - `<PARAM NAME=DisableSubfiles VALUE=True>`
 - `<PARAM NAME=UseHostColors VALUE=True>`
2. If you have custom-designed Web pages, use the Deployment Wizard to add the parameters.

- Refer to 4.8.4, “Mapping a network drive to the iSeries” on page 162 for information on how to map a network drive to the iSeries.
- Open an existing customized screen.
- On the Additional Options window, click **Advanced Options**. Then click the **Additional Parameters** tab.
- Type DisableSubfiles in the Name field. Type True in the Value field. Click **Set**.
- Type UseHostColors in the Name field. Type True in the Value field. Click **Set**.

This adjustment is only applicable to Screen Customizer-enabled sessions and should not distort any custom-designed sub files.

4.6 Add Printer Definition Table entry

A printer definition table allows a custom printer to be created. To create a new printer definition table, perform the following:

1. Map a network drive to the iSeries. See 4.8.4, “Mapping a network drive to the iSeries” on page 162.

```
net use z: \\my400.ibm.com\hodpdt /user:bob
```

2. Use a text editor to create a definition file. Type the following after clicking **Start -> Run** on your PC:

```
notepad z:\newprt.pdf
```

3. Use a text editor to modify the script. Type the following after clicking **Start -> Run** on your PC:

```
notepad z:\pdtcompilerapplication-OS400
```

- Locate the word NONGUI_COMMAND.
- Add a new line (as follows):

```
NONGUI_COMMAND='newprt.pdf "my description" '
```

4. Compile the newprt.pdt file. Type the following OS/400 commands:

- qsh
- cd /qibm/proddata/hostondemand/hod/samples
- cd pdtcompilercommandfiles
- PdtCompilerApplication-OS400

Important: PdtCompilerApplication-OS400 is case sensitive. Enter it as shown.

For additional information, refer to Chapter 19, “Host printing” on page 661 for more information on 5250 Host Print, and to the online *Host Printing Reference* document.

4.7 Performance tips

By following the suggestions in this section you should be able to improve the overall performance of your iSeries Host On-Demand system.

4.7.1 Web page caching

We found that when using the original iSeries Web server, 5769-DG1 and 5722-DG1 HTTP server, Host On-Demand can utilize the HTTP server “local caching” feature (57% performance improvement in the hits/sec/CPW). A read from main memory is much faster than accessing the object from disk. However, if memory is being required by a system process, the objects will be paged out, which negates the performance gains. Refer to *AS/400 HTTP Server Performance and Capacity Planning Redbook*, SG24-5645, for additional details.

To enable Web caching for “original” iSeries Web instances:

1. WRKHTTPCFG
2. Add the following directives:
 - CacheLocalMaxBytes 100 M
 - LiveLocalCache On
 - CacheLocalFile /QIBM/ProdData/hostondemand/hod

To enable web caching for iSeries web instances “powered by Apache”:

1. Start the Administration web page. See Section 4.2, “Using IBM HTTP Server (Powered by Apache)” on page 149 for details.
2. Click on the **Global Settings** settings link in the left menu panel. See Section Figure 4-2, “Starting the iSeries Apache Configuration tool” on page 150.
3. Click on the **Performance** link near the bottom of the menu.
4. Under the “Files to cache when server is started” section, click the **Add** button.

5. Type `/QIBM/ProdData/hostondemand/HOD/*`, then select Copy into memory.
6. Since the information in the HOD directory is stable, the normal setting for “Dynamically cache files based on file usage” is off.
7. “Update cache when files are modified” is normally set to on.
8. Click **OK**, then click the **Restart** button for the web instance.

4.7.2 Compile Host On-Demand for faster execution

The largest performance gain we noticed was by installing the JVM 1.3. See 4.1, “Upgrade JVM level to 1.3” on page 148. Starting with OS/400 V4R5 and higher, Java will automatically perform Just In-Time compilation.

To create a more efficient environment, OS/400’s JVM compiles Java classes into native code as they are loaded. However, since the compilation process can be lengthy, depending on the number of classes and the size of the ZIP and JAR files, this may not be desirable because it will take a long time to start a session or load a function.

To avoid delay and provide good performance, you should compile the class files, ZIP files and JAR files immediately after installation. This also allows better optimization between classes within packages.

The files to compile are:

- ▶ `sm.zip`
- ▶ `ods.jar` (see note below)
- ▶ `jndi.jar`
- ▶ `ibmjndi.jar`
- ▶ `jsdk.jar`
- ▶ `cfgsrvlt.jar` (only if you are planning to use WebSphere Configuration Servlet)

All of the files reside in `/QIBM/ProdData/hostondemand/lib`.

To compile the files, run a command like the following for each file except `ods.jar`:

```
CRTJVAPGM CLSF('QIBM/ProdData/hostondemand/lib/sm.zip') OPTIMIZE(30)
```

Note that the `ods.jar` files requires an additional option. In prompt mode for the command `CRTJVAPGM`, press F10 for additional parameters and, in the field labeled Licensed Internal Code options, replace `*optimize` with `errorreporting=2`. This option is only available on V4R3 and later. The syntax for the command line option is:

```
CRTJVAPGM CLSF('/QIBM/ProdData/hostondemand/lib/ods.jar') OPTIMIZE(30)
LICOPT('errorreporting=2')
```

Optimization can take a long time and use a lot of processor capacity. It depends on many conditions, including the power of the iSeries and what else it is doing at the time. It is best done when the machine is not busy with other tasks.

4.8 iSeries as a target host

The following tips are for use when the iSeries is the target host system.

4.8.1 5250 Workstation ID

Starting with Host On-Demand V 5.04, Host On-Demand supports some special values for workstation ID (device name). This allows Host On-Demand 5250 display and printer sessions to generate a non-arbitrary device name for a session without requiring per-session customization or a user exit.

Table 4-2 Special values for 5250 workstation ID

Character	Function	Example string	Example devices
*	Short Session ID	A123*	A123A, A123B
%	Session type: S=display P=printer	%DEV	SDEV
=	Collision avoidance. If device is in use, generate.	%DEV=	If SDEV1 is in use, then try SDEV2, ... until success
&COMPN	Computer name. Obtained from TCP settings.	&COMPN%= (MYPC=computer S=display A=short ID)	MYPCSA
&USERN	User name. (Windows clients only)	&USERN%= (BOB=user, S=display, A=short ID)	BOBSA

Character	Function	Example string	Example devices
+	Trim the excess from the right side.	+&COMPN (computer= CLIENTACCESS	

If the resulting device name exceeds 10 characters, the excess will be trimmed from the left side. This produces fewer duplicate device names for “left to right” languages, such as English. Excess characters can alternatively be trimmed from the right side by prefixing the CN keyword with a plus sign character (for example, +&COMPN).

Restrictions:

1. A numeric character in the first position of a DEVNAME is invalid, and may be converted by OS/400 to the "#" (pound or hash) character.
2. Only supported on Win32 platforms.

4.8.2 5250 Telnet dropout

If you are using a firewall, make sure that the firewall inactivity time-out value for Telnet connections is at least as long as the session keep alive timeout (TIMMRKTIM0) parameter on the CHGTELNA OS/400 command.

4.8.3 Tip for 5250 printing

The first time an output queue is used, a CPA3394 (“Load form type ‘*STD’ in device xxx”) message is directed to the message queue for the OUTQ. The message must be answered before the printouts begin to print.

Caution: If you use the autoreply command below, the feature is automatically set for all printers for the iSeries. When the autoreply is activated, the printer will not prompt the printer operator for form changes.

To automatically have the system answer the message, use the following command:

```
ADDRPYLE SEQNBR(9999) MSGID(CPA3394) RPY(G)
```

4.8.4 Mapping a network drive to the iSeries

The iSeries can participate in a Windows Network Neighborhood. It may be helpful to create the following shares:

Table 4-3 Typical shares for Host On-Demand and Screen Customizer

Share	Target directory	Used for
hodpubl	/QIBM/ProdData/hostondemand/hod	Publish custom Web pages using the Deployment Wizard.
hodpdt	/QIBM/ProdData/hostondemand/hod/samples /PdtCompilerCommandFiles	Publish printer definition tables.
scpubl	/QIBM/ProdData/hostondemand/hod/custom	Publish Screen Customizer maps.

Tip: When you attempt to map a network drive to the iSeries, the Windows user ID and password must match your iSeries user ID and password. If your workstation operating system is Windows NT or Windows 2000, you may click **Connect using a different user name** on the Map Network Drive.

For additional information, refer to:

<http://www.ibm.com/servers/eserver/iseries/netserver>

4.8.5 Additional iSeries-related Web pages

Table 4-4 lists some useful iSeries Web pages.

Table 4-4 Additional iSeries related Web pages

Title	Web page
Common SSL problems. Also has a table describing the telnet-ssl return codes	http://publib.boulder.ibm.com/iseries/v5r1/ic2924/tstudio/tech_ref/tcp/telntssl/Index.htm
Telnet exits-filter Telnet service by IP address	http://publib.boulder.ibm.com/iseries/v5r2/ic2924/info/rzaiw/rzaiwmantelsrvr.htm
National Language exit - ADDNLS tool	http://publib.boulder.ibm.com/iseries/v5r1/ic2924/tstudio/tech_ref/tcp/telex/telexdwn.htm
iSeries Performance Estimator Tool	http://publib.boulder.ibm.com/pubs/html/iseries/online/chgfrm.htm



Clients

Host On-Demand provides a variety of types of clients: emulator clients, FTP clients, CICS Gateway clients, and database clients. Most of these are available as either a cached client or a download client.

This chapter introduces the various Host On-Demand clients and describes how you can use and customize them.

This chapter also discusses support for clients running on Java 2 enabled browsers.

5.1 Host On-Demand default clients

The following figure shows the default clients supplied with Host On-Demand.

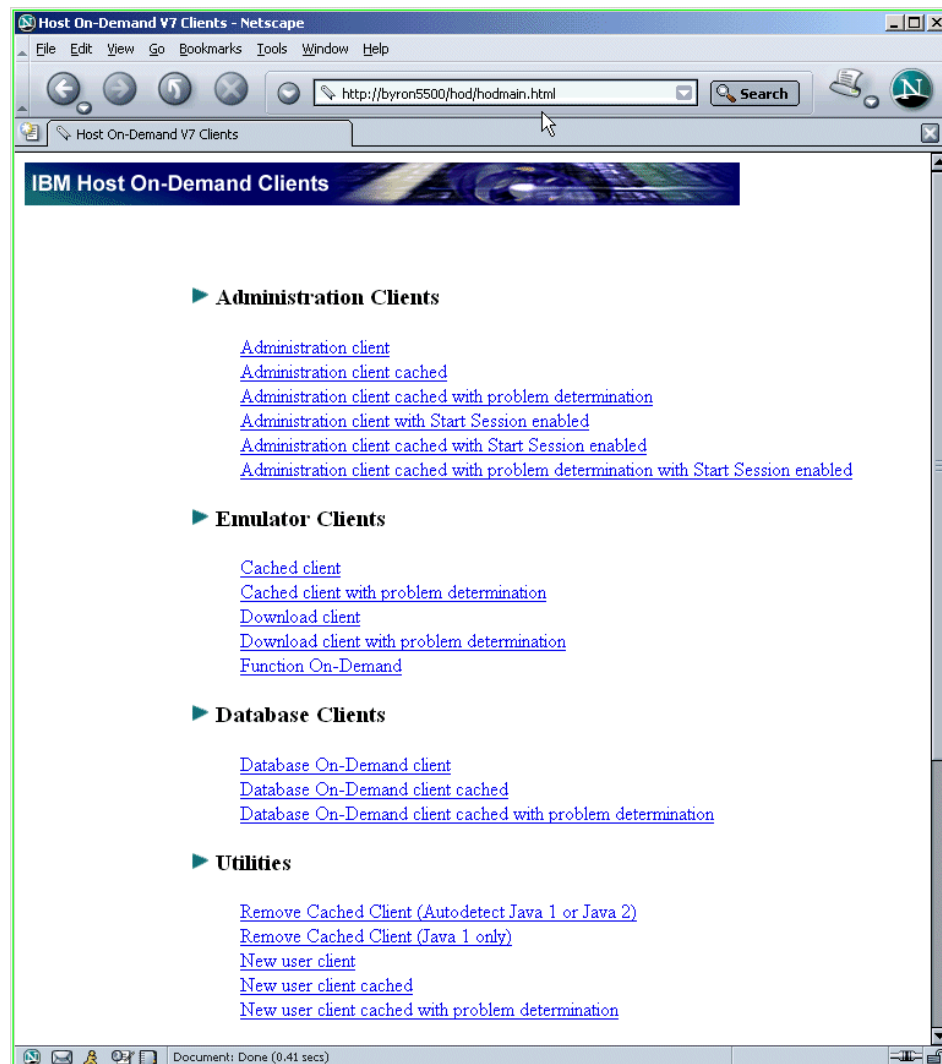


Figure 5-1 /hod/hodmain.html

The following table lists the Host On-Demand default clients. These clients use the Configuration server-based model. The Deployment Wizard may be used to create HTML files with custom versions of the emulator clients. See Chapter 14, “Deployment Wizard” on page 529 for details.

Table 5-1 Host On-Demand clients

Package	Client	HTML File
Administration clients	Administration client download	HODAdmin.html
	Administration client cached	HODAdminCached.html
	Administration client cached with problem determination	HODAdminCachedDebug.html
	Administration client with Start Session enabled	HODAdminFull.html
	Administration client cached with Start Session enabled	HODAdminCachedFull.html
	Administration client cached with problem determination with Start Session enabled	HODAdminCachedDebugFull.html
Emulator Clients	Cached client	HODCached.html
	Cached client with problem determination	HODCachedDebug.html
	Download client	HOD.html
	Download client with problem determination	HODDebug.html
	Download client with Screen Customizes/LE Interface	HODCustom.html
	Function On-Demand client	HODThin.html
Database Clients	Database On-Demand client	HODDatabase.html
	Database On-Demand client cached	HODDatabaseCached.html
	Database On-Demand client cached with problem determination	HODDatabaseCachedDebug.html
Utilities	Remove cached client (Autodetect Java 1 or Java 2)	HODRemove.html
	Remove cached client (Java 1 only)	HODRemove.html
	New user client	NewUser.html
	New user client cached	NewUserCached.html
	New user client cached with problem determination	NewUserCachedDebug.html

5.1.1 Administration clients

The administration client (HODAdmin.html) starts the Administration window, where you can:

- ▶ Manage users, groups, and sessions
- ▶ Configure, manage and trace the Redirector service
- ▶ Configure Database On-Demand
- ▶ Enable security
- ▶ View trace and message logs
- ▶ Disable functions to end users

Refer to Chapter 7, “Administration” on page 273 for complete details on the functions and operations of the administration client. You must use one of the following clients to do administration. The Deployment Wizard does not have the capability to create customized administration pages.

Administration client cached

This client starts the administration client in a cached environment. Load this client if you want to use the administration client in a cached environment without problem determination. The advantage of the administration client cached is that it can be cached along with other cached clients in the browser. In releases prior to Version 5, the cached client had to be removed before the administration client could be loaded.

If you want to bookmark the Administrator client cached, you must manually create the bookmark. It must point to HODAdminCached.html, so that Host On-Demand can compare the cached version to the server version. This allows Host On-Demand to recognize and notify you that a newer version of the administration client cached is available at the server.

Administration client cached with problem determination

This client also starts the administration client in a cached environment. Load this client if you need to use the administration client in a cached environment with problem determination (session logging and tracing).

Administration client with Start Session Enabled

Loads the download version of the full Administration client. The full administration client gives the administrator the additional ability of starting sessions to configure runtime properties. However, the download size of the full administration client is larger than the download size of administration client.

Administration client cached with Start Session enabled

Loads the cached version of the full Administration client. Like the cached version of the regular Administration client, this client can be cached along with the cached client in the browser.

Administration client cached with problem determination with Start Sesssion enabled

Loads the cached version of the full Administration client with problem determination (session logging and tracing) enabled.

5.1.2 Download clients

A download client is one where the code is downloaded from the server on every invocation of the client. The advantage of the download client is that the browser does not need to be stopped and then restarted.

This client can be used when:

- ▶ You do not want to take up disk space on client machines by installing the cached client.
- ▶ The download time is not an issue.

Important: Running a download client with a cached client loaded in the browser will result in inaccurate and unpredictable results. You must first remove a cached client from the browser before using a download client. See 5.4.2, “Remove cached client” on page 175 for more information.

For information on the Java 2 download client, see “The Java 2 download client” on page 220.

5.1.3 Cached clients

A Host On-Demand cached client has all the functionality of the download client. It is cached on your local disk the first time you download it. The next time you start the emulator session, only a small applet downloads from the server, reducing the time needed to start the session. When using a Java 1 or Java 2 enabled browser, the applet that is downloaded checks to see if the software on the server is more recent than the software that has been cached, and if so, the cached software is updated.

For more information about the Java 2 cached client see “The Java 2 cached client” on page 210.

Beginning with Host On-Demand Version 5, all clients consist of a collection of smaller JAR/CAB files, called components, to allow for the administrator to create smaller clients, and to provide the ability to update individual components rather than the entire client. This has been called componentization and is more fully documented in 5.2, “Componentization” on page 172.

Since the clients are broken into components, only the specific component will be updated, and then only after that component has been referenced. Under most circumstances, you can continue to use the current level of the cached client to connect to a host while the newer components are downloading. See 5.3, “Smart caching” on page 174 for further information.

The cached client is persistent across operating system restarts and browser reloads. If you want to remove it, you must load HODRemove.html. See 5.4.2, “Remove cached client” on page 175

Restricted users on Windows XP and Windows 2000

Restricted users of Windows 2000 and Windows XP can now install and use the Java 1 or Java 2 Host On-Demand cached client. A separate version of the cached client will be installed for each restricted user.

Previously only users with Administrator or Power User authority could install and use the cached client on Windows 2000 and Windows XP.

Sharing the cached client on the Windows platform

Sharing the Host On-Demand Java 2 cached client

Users cannot share the Host On-Demand Java 2 cached client.

Sharing the Host On-Demand Java 1 cached client

Multiple users on Windows XP, Windows 2000, Windows ME, Windows NT, or Windows 95/98 can share a single Java 1 cached client installation.

For users to share a single cached client installation, the system administrator must take the following steps:

1. Use the Additional Parameters tab of the Advanced options panel of the Deployment Wizard to add the ShareCachedClient parameter to the cached client HTML file.

This step is required if some of the machines on which the cached client installation will be shared are running Windows 2000 or Windows XP.

This step is not required if all the machines on which the cached client installation will be shared are running Windows NT, Windows Me, or Windows 95/98.

2. Modify each client machine to include the IBMHOD directory:

- a. On client machines running Windows 2000, Window XP, or Windows NT, have a user with Administrator or Power User status take the following steps.
 - i. Create a directory named IBMHOD under the system “all users” directory. For example, if Windows 2000 is installed on the C: drive, the directory path would be:
`c:\Documents and Settings\All Users\IBMHOD`
 - ii. Change the security settings for the IBMHOD directory so that restricted users have read, write, and modify access.

If this step is omitted then a restricted user attempting to install or use a shared cached client will not be able to do so. Instead, an error message will be displayed indicating that there may be a problem with the file system.
 - iii. If the IBMHOD directory already exists, run `HODRemove.html` to remove the previous version of the cached client.

Note: a user with Administrator or Power User status can create the IBMHOD directory automatically merely by installing the cached client. However, the user with Administrator or Power User status must still change the security settings of the IBMHOD directory as described above.
- b. On client machines running Windows ME or Windows 95/98, have any user create the following path:
`c:\Documents and Settings\All Users\IBMHOD`

After the above setup, any user can:

- ▶ Install the shared cached client
- ▶ Use the shared cached client
- ▶ Upgrade the shared cached client

After the shared cached client is installed, any user attempting to use the shared cached client for the first time will be prompted to restart the browser.

Java 1 cached client support across the Internet

If you deploy the Java 1 cached client to the Internet, consider that your users might use Host On-Demand with other business partners running Host On-Demand servers at different service levels. This could be a problem if your user needs different functions when accessing servers at different service levels. Components of different service levels are not supported within a single cached client, and there can be only one cached client on a machine.

This section discusses the problems that might occur.

Also, if you want your users to be able to attach across the internet to servers running different versions of Host On-Demand and you want your users to be able to run the cached client with these servers, then you must install Host On-Demand version 5.0.4 or higher on each server.

Note: These problems do not occur with the Java 2 cached client. See “Increased flexibility with Java 2 cached clients” on page 216.

The remainder of this subsection is applicable only to the Java 1 cached client, not to the Java 2 cached client.

Installed Java 1 cached client version is later than HOD server

If the software on the server is an earlier version than the cached software, the cached client applet checks the version levels of the components and prevents caching of any new components. To cache new components, remove the more recent version of the cached client and then install the earlier version of the cached client. To avoid this problem, select all the functions the user needs (across all sites the user accesses) in the preload list when you create the HTML page using the Deployment Wizard.

Installed Java 1 cached client version is earlier than HOD server

When a client points to a server running a later version of Host On-Demand, and the upgrade test passes, all cached components are automatically upgraded (not only the components defined in the HTML page's preload list). Because all cached and new components are upgraded simultaneously, the upgrade might generate additional Web server load. After the upgrade, the client can point back to the server running the earlier version of Host On-Demand, and the later version of the cached client will function correctly.

Workarounds with Java 1 cached client

To prevent these complications with Java 1 cached clients, you can do some or all of the following:

- ▶ Select all the functions a user needs (across all sites the user accesses) in a preload list when you create an HTML page using the Deployment Wizard.
- ▶ Use the disable function of the Deployment Wizard to disable all functions not in the preload list and the functions that are not needed for your users.
- ▶ Create separate HTML pages for different user groups.
- ▶ Give your HTML pages a name that identifies your company.

If you are using locally stored preferences, the custom HTML pages you create *must* have names unique to your company, because the HTML file names are used to differentiate between the locally stored preferences of different sites. Using generic names could cause preference conflicts for your users.

- ▶ Always install Screen Customizer to prevent users who are accessing your server from losing Screen Customizer functions when accessing other sites.

If you use the cached client on the Internet, you must install Screen Customizer on your server. If a full-function version of Screen Customizer is cached and Screen Customizer is not installed on the server, the cached client applet issues an error message and prevents the upgrade.

If you have problems managing a cached client deployment across the Internet, see the Host On-Demand support Web site for more information:

<http://www.ibm.com/software/webservers/hostondemand>

5.1.4 Emulator clients

The emulator clients provide emulation support for:

- ▶ 3270 displays
- ▶ 3270 printers
- ▶ 5250 displays
- ▶ 5250 printers
- ▶ VT displays
- ▶ FTP clients

These are the most used clients. The emulator clients listed in Table 5-1 on page 165 provide support for all of these terminal emulators. The Deployment Wizard, (see Chapter 14, “Deployment Wizard” on page 529) provides you with the capability of custom creating an emulator client that supports one or more of these device types.

5.1.5 Problem determination clients

Problem determination clients have special functions that allow them to trace sessions and log information for problem determination purposes. The default Host On-Demand problem determination clients have the character string Debug appended to the name of the file, for example H0DDDebug.html. The Deployment Wizard can also create debug clients; however, the name of the file will not indicate that debugging components have been included.

5.1.6 Function On-Demand client

The Function On-Demand (HODThin.html) client is much smaller than the other clients. On the initial download only the basic functions are downloaded, thus greatly reducing the startup time. Other functions are downloaded only when they are required. Some functions may be required immediately (such as the 3270 emulator), while other functions (file transfer, for example) might never be invoked or might not be needed for a long time.

The Function On-Demand client can be configured with the traditional “green screen” interface, or it can be configured with the Screen Customizer interface.

You can also custom build your own function on-demand client using the Deployment Wizard, specifying what functions are enabled and what functions are to be downloaded initially. We recommended that you use the Deployment Wizard to create a customized HTML file instead of the Function On-Demand client to better suit your requirements.

The default function on-demand client, HODThin.html, is a download client; however, with the Deployment Wizard you can create a cached function on-demand client that initially loads the functions you wish. Refer to Chapter 14, “Deployment Wizard” on page 529 for details.

Note: The Function On-Demand client is not available with Java 2-enabled Web browsers. See 5.14, “The Java 2 download client” on page 220.

5.2 Componentization

The Host On-Demand cached client has been identified as the overwhelmingly preferred Host On-Demand client. In Host On-Demand Version 4, the cached client needed to include all the class files that could possibly be used by all four emulator types and all functional components, such as macro recording and playback, ColorRemap, and all possible code pages. This produced a very large archive file of class files, many of which would probably never be used. Downloading these files, although done only when the files changed, created a response time problem for users on slower speed lines as well as a network utilization problem.

The cleanest way to resolve these concerns was to break each function into its own archive file, and then a smaller client could be built that contained only the functions required by the user. This technique is referred to as componentization.

With the introduction of componentization, Host On-Demand was able to implement smart caching. See 5.3, “Smart caching” on page 174.

Table 5-2 provides a breakdown of JAR /CAB files that are sent to the workstation for a cached client installation. In this table, the base install is represented by the first five JAR files (CAB files if using Internet Explorer). These files are common across all client emulators and represent the minimum cache install available.

The lower portion of Table 5-2 includes files that are required for specific emulation requirements such as 3270 Display. By selecting a specific column, such as 3270 Display, a list of required files may be obtained. The administrator can calculate the approximate size of the cached client and estimate installation time for a new installation or code update.

Table 5-2 Required class files by client

Class File Names	3270 Display	5250 Display	3287 Printer	5250 Printer	VT100 / 220 Display	CICS Gateway
ha_en.jar	X	X	X	X	X	X
habasen.jar	X	X	X	X	X	X
hacp.jar	X	X	X	X	X	X
hodbasen.jar	X	X	X	X	X	X
hoding.jar						
ha3270n.jar	X		X			
hafntap.jar	X	X				
hafntib.jar	X	X				
ha5250n.jar		X				
haprintn.jar			X	X	X	
havtn.jar					X	
hacicsn.jar						X
ha3270pn.jar			X			
ha5250n.jar				X		
ha5250pn.jar				X		
For further information, please refer to the IBM Host Access Toolkit documentation						

5.3 Smart caching

Smart caching is the ability to cache and upgrade individual components of the client, even components that were not included in the initial loading of the client. A side benefit is the ability to create a smaller client footprint for network distribution that includes only basic functionality for downloading to the workstation, and incrementally adding only those functions that the user actually uses rather than all the functions that the user may use. In Table 5-2 on page 173, you can see the basic components that are required to have a functional Host On-Demand client. The remainder of the associated JAR/CAB files are downloaded and maintained in permanent cache only when the user requests a function requiring that function, for example file transfer.

This configuration or packaging of required components is accomplished with the use of the Deployment Wizard. Refer to Chapter 14, “Deployment Wizard” on page 529 for details. Several independent configurations can be created to satisfy specific client requirements within a large diverse environment. For example, several organizations require keyboard remapping and file transfer capabilities. Instead of shipping the necessary JAR file to everyone, a specific HTML page is created for their unique requirements reducing the installation time and network contention. Other advantages include controlling specific configuration of the client’s capabilities. This may be necessary if the workstation is a shared device or security may be an issue.

5.4 Utility clients

There are three valuable utility clients:

- ▶ New user client

This client allows non-administrative users to create other user accounts.

- ▶ Remove cached client (autodetect Java 1 or Java 2)

This client is used to remove a Java 1 or Java 2 cached client from a browser.

- ▶ Remove cached client (Java 1 only)

This client is used to remove a Java 1 cached client from a browser.

5.4.1 New user client

If the administrator has checked the **Allow users to create accounts** option in the Users/Groups view of the Host On-Demand administration applet, users will be allowed to load a special client, `NewUser.html`, that allows them to create accounts for themselves or other users. The purpose of this client facility is to remove some of the load from the administrator and delegate the responsibility for creating users to department managers, site managers, or other designated people.

The default HTML file will insert users into the default Host On-Demand group, HOD, when an account is created. This file can be used as a template to create customized applets that will allow a user to be inserted into specific groups or combinations.

To add users to a group other than HOD, you must modify the following HTML parameter:

```
<PARAM NAME="Groups" VALUE="HOD">
```

You may replace HOD with any previously defined group. If you are using the Host On-Demand default data store, you may specify more than one group separated by commas. For example by specifying the following parameter:

```
<PARAM NAME="Groups" VALUE="ProjectLeader, HOD">
```

Users will be added to both the ProjectLeader and HOD groups.

This utility is available in three forms:

- ▶ Download client - `NewUser.html`
- ▶ Cached client - `NewUserCached.html`
- ▶ Cached problem determination - `NewUserCachedDebug.html`

5.4.2 Remove cached client

`HODRemove.html` is the remove cached client utility. In Host On-Demand Version 4 this utility was used to remove the cached from the Netscape browser after Host On-Demand stopped using Netscape's smartupdate function to manage the persistently cached applications. Users of Internet Explorer continued to use the facilities of Internet Explorer to remove the cached client. With the introduction of Host On-Demand Version 5 and componentization (see 5.2, "Componentization" on page 172), it became necessary for Host On-Demand to assume management of the cached clients directly. This required all browsers to use the `HODRemove.html` tool to clear the cache.

In Host On-Demand 7.0, the remove cached client function was divided into two types, both of which are handled by HODRemove.html.

1. Remove Cached Client (Autodetect Java 1 or Java 2)

This selection causes HODRemove.html to detect the browser type, either Java 1 or Java 2 enabled, and to remove the corresponding type of cached client data.

If the browser type is Java 1, then HODRemove.html removes Java 1 cached client data, if any is present.

In contrast, if the browser type is Java 2 enabled, then HODRemove.html removes Java 2 cached client data, if any is present.

2. Remove Cached Client (Java 1 only)

This selection causes HODRemove.html to remove Java 1 cached client data, if any is present. (However, this selection does not work if the web browser is Netscape 6.x).

For more information see “Removing the cached client” on page 217.

5.5 CICS Gateway client

The CICS Gateway client is a special 3270 emulator client. It connects only to a distributed CICS Gateway, thus forcing a three-tiered environment. The CICS Gateway client acts as a 3270 Telnet server, communicating with the client via TN3270 protocol and with CICS via SNA protocol.

There are several limitations to the CICS Gateway client that force most users to select the standard 3270 emulator client:

- ▶ SSL sessions are not supported
- ▶ You can't sign on to CICS
- ▶ The CECI transaction is always started
- ▶ You are limited by the EPI subset (no BMS PAGING/ACCUM, no RETURN IMMEDIATE)
- ▶ No ATI support (STARTed transactions)
- ▶ The CICS Gateway client must connect via a distributed CICS Gateway, thus forcing a three-tiered environment and arbitrarily increasing path lengths

Because of the above limitations, we recommend that under normal circumstances you use the standard TN3270 emulator client.

5.6 Database On-Demand

Refer to Chapter 6, “Database On-Demand” on page 247 for details on the Database On-Demand client.

5.7 The emulator session window

An emulator session window consists of a title bar, a menu bar, a toolbar (with or without explanatory text), a presentation space, an operator information area (OIA), a keypad, and a status bar at the bottom of the window, as illustrated in Figure 5-2.

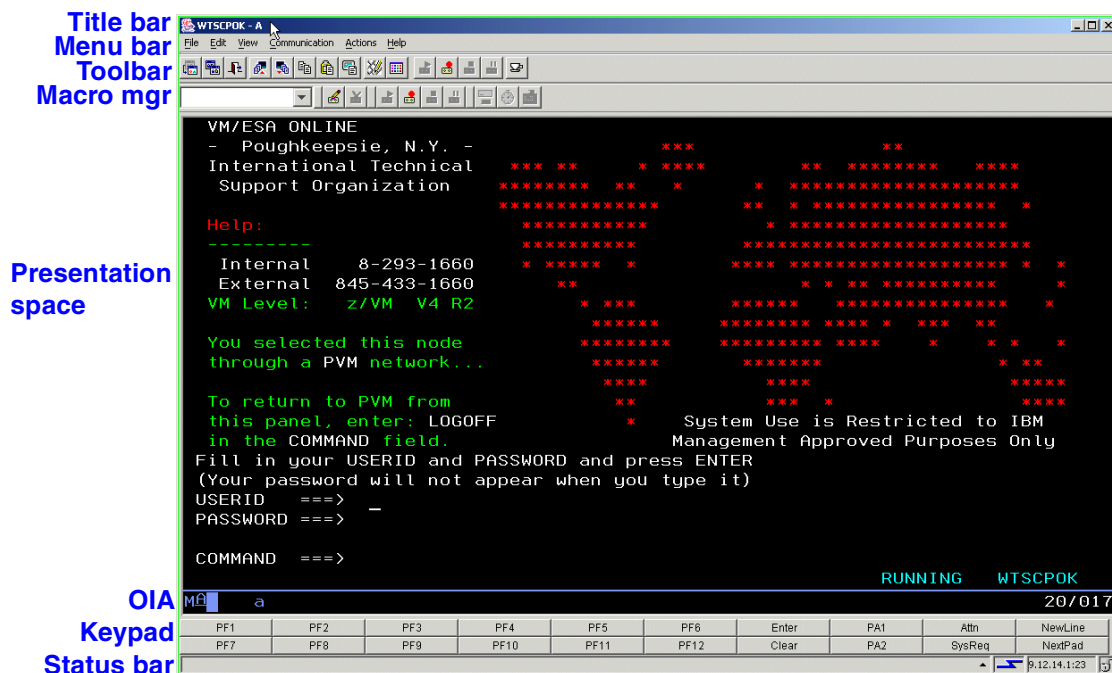


Figure 5-2 Host On-Demand session

In this section we will describe the OIA, and then discuss how the user or administrator tailor an emulator session in the following ways:

- ▶ Customizing the toolbar
- ▶ Color remapping
- ▶ Keyboard remapping

5.7.1 Operator information area

The operator information area (OIA) is located across the bottom portion of the session window. This is where communication information is displayed. Almost every position in the area is used at some time, though many may be blank at any one time. The indicators are all explained in the Host On-Demand online help, but some deserve special mention:

- ▶ The first three columns show the type of connection and the condition of the host application. As you log on and move through an application, the indicator in column 3 changes.
- ▶ If the session is using SSL security and is connected in encrypted mode, column 4 has a + sign.
- ▶ Column 7 shows the session ID or short name, from a to z.
- ▶ Starting in column 19, you will sometimes see a Communications or Program Check message, which includes a number. Such messages do not necessarily indicate a problem but merely the status of the connection. For example, you may see COMM 657, followed by COMM 655 as the handshaking progresses and session connects; the length of time during which the messages appear depends on the performance of the various links and devices in the path.
- ▶ Column 75 through 80 indicate the position of the cursor by row and column. The position does not vary according to the screen size.

Table 5-3

Column	Status Character	Description
1	M	Indicates a connection has been established to a Telnet server.
2	A	The protocol in use is TCP/IP.
3	* or p ?	<ul style="list-style-type: none"> - The session has established an LU-LU connection with an application program. - A SSCP-LU connection has been established but the connection has not been established to the application. -The session bind has not been established or is not connected.
4	+	When the session data is encrypted, the character "a" will change to "a+".
7	a-z	Indicates which host session you are using.

Column	Status Character	Description
9-17	X[] X SYSTEM X <-o->	- (3270 session only) System response time. It is made up of network hops and system response time until the keyboard is unlocked for additional input. - Application or transaction is in response mode. Keyboard is locked until process is complete. -Indicates that an attempt was made to insert a character into a protected field. Press reset and move to an edit or update field.
19-26		
75-80		Cursor position

Color remapping of the OIA cannot be modified by using the mouse pointer and selecting an area. It requires the modification to be done by utilizing the Advanced color remapping window (see 5.7.3, “Color remapping” on page 183).

5.7.2 Customizing the toolbar

Toolbar customization, new in Host On-Demand Version 6.0, allows Administrators and users to add, edit and remove buttons on the session toolbar. These buttons can be configured to launch an applet, run an application, go to a URL, run a macro or perform a menu function. The toolbar settings are saved for future sessions.

Administrators can deploy their own customized toolbars either by customizing a session during the Deployment Wizard stage or at a later stage by importing a customized session for a group or user in the administration utilities. By pre-customizing the toolbar, this feature can be disabled for the user so they can not add, modify, remove or reset the toolbar.

By default the toolbar appears with the following buttons; for clarity, the toolbar text has been enabled (see Figure 5-3).

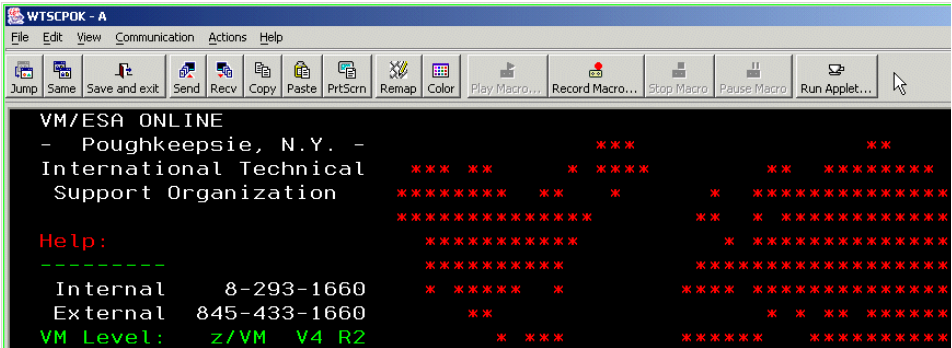


Figure 5-3 Default 3270 toolbar with toolbar text enabled

Add button

By right-clicking in the toolbar area and selecting **Add Button...** you will be presented with the window shown in Figure 5-4.

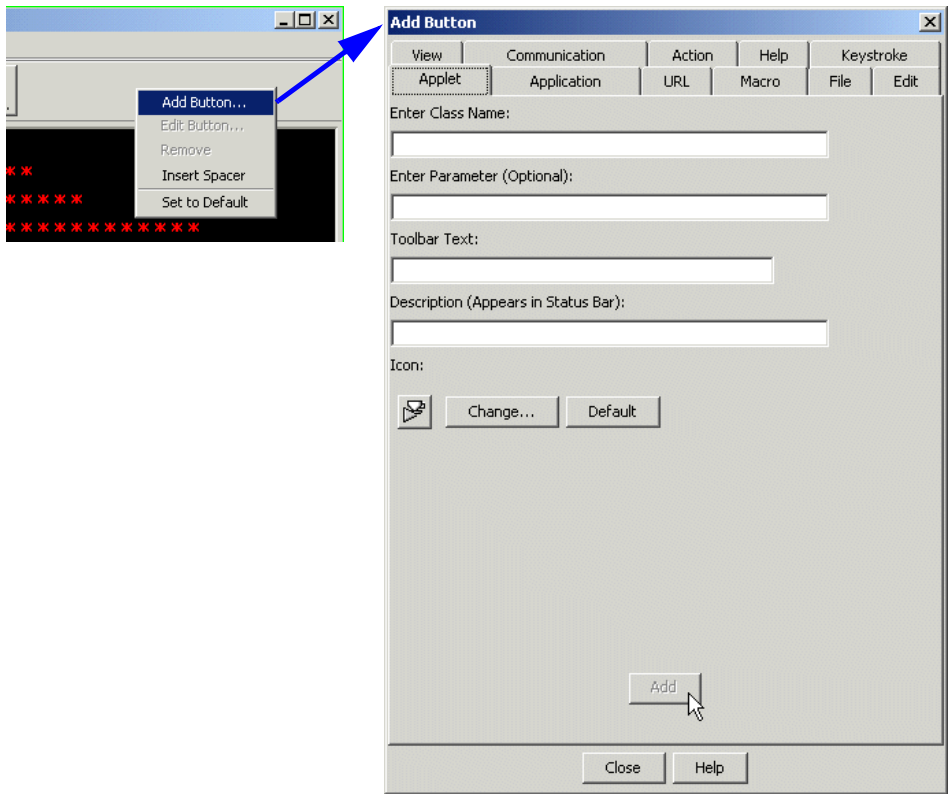


Figure 5-4 Add Button window for 3270 session

Each button type has a unique attribute, for example the Class name for the applet, along with the two common fields, Toolbar Text and the Description. The Description will appear in the status bar at the bottom of the client window when the mouse passes over the button.

If a user has a disabled function, for example playing a macro, they will not see that tab on the Add Button window.

The tabs are arranged in the following order:

► Applet

This allows you to specify an applet you want to launch. Type the class name with or without an extension. The applet must implement the `ECLAppletInterface` in order to run.

► Application

Specify the full application path, along with any parameters in accordance with the platform syntax. A Browse feature is provided as an aid in locating the application.

► URL

Enter the URL that will be opened in a browser window.

► Macro

Choose from one of the prerecorded macros. The Toolbar Text and Description fields will default to the macro name. These default values can be edited if desired.

The File, Edit, View, Communication, Action and Help tabs are associated with the drop-down menu functions.

The new button will appear to the left of where you right-clicked on the toolbar. You can also add a button by selecting from the drop-down menu **Edit -> Preferences -> Toolbar -> Add Button....** When selecting from the drop-down menu, the new button will be added to the end of the toolbar.

Modifications to the existing toolbar buttons and layout can be achieved by right-clicking either the button to be modified, or elsewhere in the toolbar to insert spacers.

Edit button

By right-clicking the desired button, and choosing **Edit Button....**, an Edit window is displayed, allowing you to modify the button attributes (Figure 5-5).

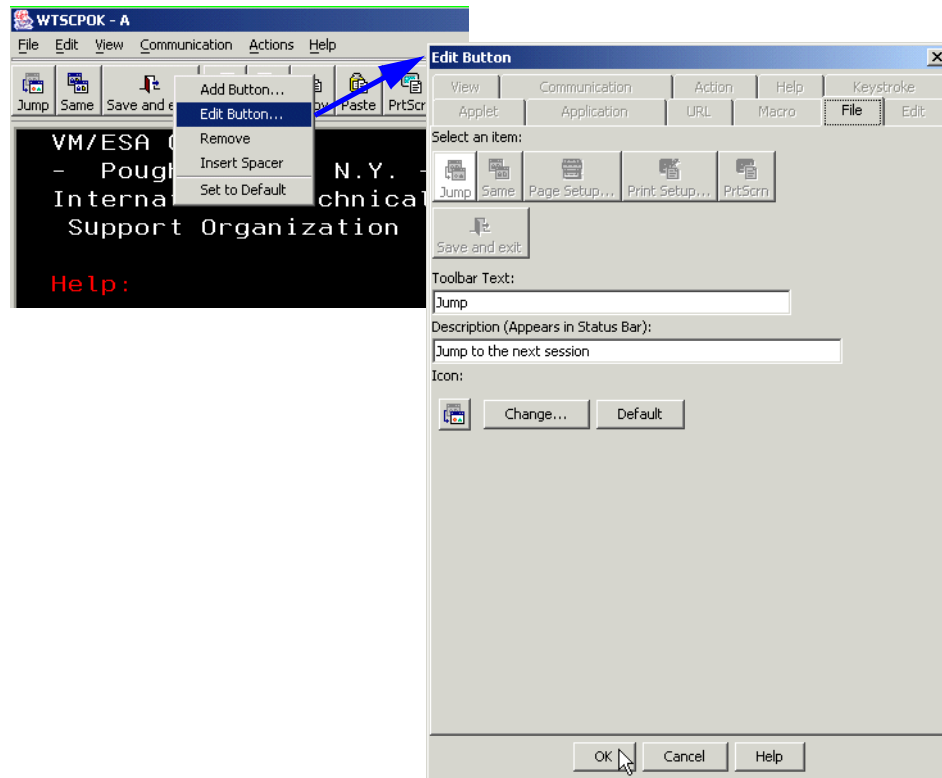


Figure 5-5 Modifying existing toolbar buttons

Remove button

Removing a toolbar button is done by right-clicking the button, and selecting **Remove**.

Insert Spacer

Toolbar buttons can be grouped together or separated by the use of toolbar spacers. By right-clicking the toolbar and selecting **Insert Spacer**, a spacer is inserted to the left of where you clicked, with the outer buttons being shifted to the right.

Set to Default

By selecting **Set to Default**, the toolbar customizations will be removed, and the toolbar will be reset back to the default settings. If a user resets a session that contains customizations defined by the administrator for a group, as well as personal customizations, both levels of customizations will be removed for the period of that login. The administrator-defined customization will return on the next login; however, the personal customization will have been permanently removed.

Customizing the toolbar for spawned clients will not be saved, for example making a file transfer in the VT session brings up the FTP client. While it is possible to modify the toolbar for the spawned FTP client, the next invocation of this FTP client will show the default toolbar.

Note: The Macro Manager toolbar item can *not* be customized.

5.7.3 Color remapping

Each host screen is made up of fields with attributes and elements. Elements are simply a way to group fields that share the same attributes. When you remap a color, all the fields that share those same attributes throughout your host applications will also remap to the new color. If you are not familiar with field elements and attributes, you may be surprised to see that other fields throughout your host applications will be remapped to the same color. In addition you may find other fields that were the original color will not be changed. These fields do not contain the same attributes so they are different elements.

There are two ways to access the color mapping windows for Host On-Demand:

- ▶ From the drop-down menu, selecting **Edit -> Preferences -> Color...**
- ▶ Clicking the **Setup display color** button on the toolbar

The first window displayed is the basic color window (see Figure 5-6 on page 184). To change a screen element, you must first click it in the session window. The sample text will then adopt the attributes of this element. If the element has foreground or background colors, they can be modified by clicking the desired color in the palette. Foreground or background colors may also be specified using RGB values by clicking either the **Foreground color** or **Background color** buttons adjacent to the color palettes. In order to modify the OIA, you must use the Advanced window as described below.

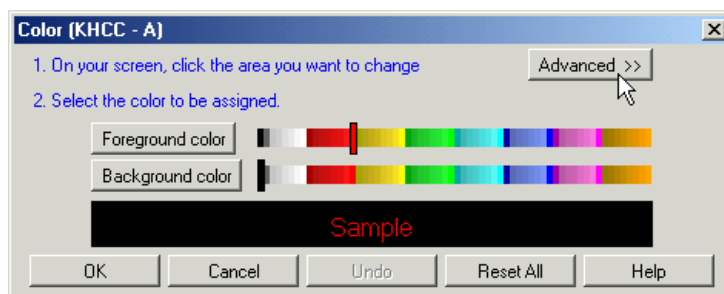


Figure 5-6 Basic color mapping window

The Advanced color mapping window (see Figure 5-7), which is toggled by clicking **Advanced**, allows modification to the base attributes, extended background as well as the operator information area (OIA). Each session type, 3270, 5250 and VT, has its own unique elements that may be modified. These elements and attributes are listed within the Host On-Demand online help.

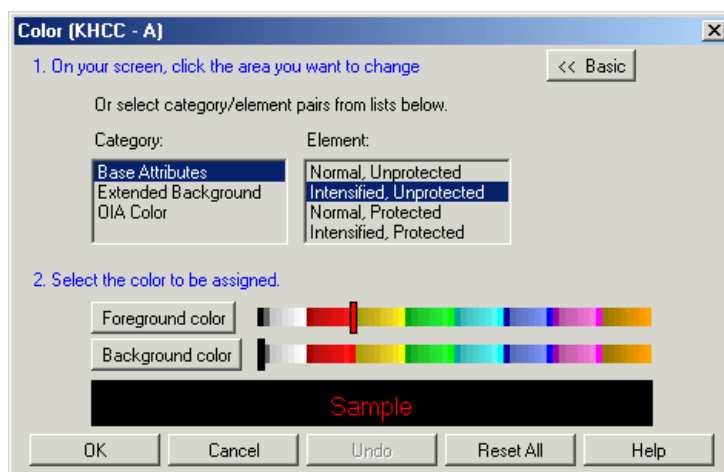


Figure 5-7 Advanced color mapping window

The Host On-Demand online documentation provides further information and procedures for modifying the host session colors.

5.7.4 Keyboard remapping

Most common host system functions are mapped to a key, but some are not. You may want to change the function of a key, map an undefined function or create a new function that currently isn't mapped. The keyboard remapping function provides the ability to display keyboard assignments on a per-key basis. The basic procedures are covered the Host On-Demand online documentation; however, it may prove helpful to discuss assigning keys to custom functions.

Assigning keys to custom functions

If you want to assign a key or key combination to a custom function that is not listed under any categories in the Keyboard window, you must first define the functions by adding them via HTML parameters. For pages generated via the Deployment Wizard, the custom functions must be added using the Advanced Options window as shown in Figure 5-8.

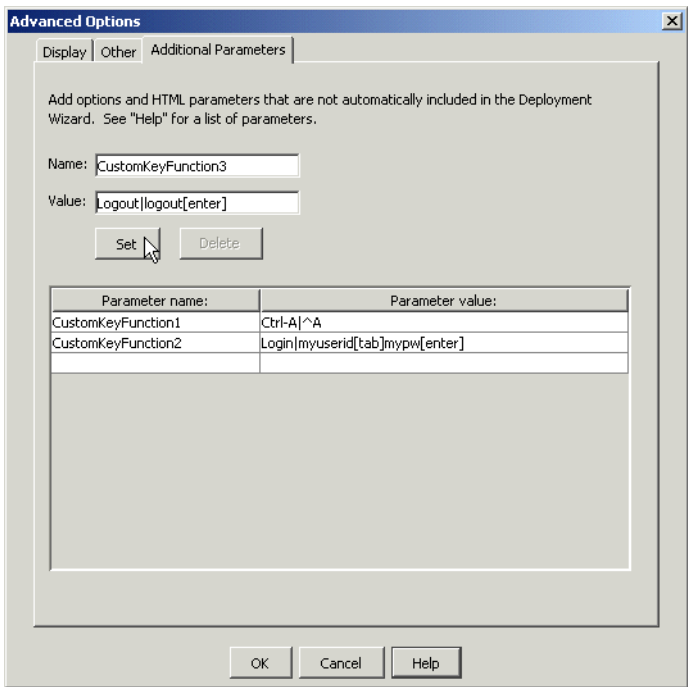


Figure 5-8 Creating custom functions in the Deployment Wizard

In the example, adding a “Logoff” function requires the following:

Parameter name This must be CustomKeyFunctionX where X is the next ordinal number, in this example 3.

Parameter value This is the combination of *Custom Function identified I function data*, for example *Logoff|logoff[enter]*. Executing this function would be the equivalent of typing logoff followed by pressing the Enter key.

For HTML pages not generated via the Deployment Wizard, you add the parameter to the applet tag as highlighted in Example 5-1.

Example 5-1 Adding custom key functions to the applet tag

```
<applet archive=CachedAppletSupporter.jar mayscript name="CachedAppletLoader"
code="com.ibm.eNetwork.HOD.cached.appletloader.CachedAppletLoader" width="584"
height="450">
<param name=Cabinets      value=CachedAppletSupporter.cab>
<param name=BookmarkPage value=AutoHODCached.html>
<param name=CachedClient value=true>
<!-- put Host On-Demand applet parameters here -->
<param name=CustomKeyFunction1 value=Logoff|logoff[enter]>

<p>If you are reading this message, your client platform is not capable of
running
IBM Host On-Demand. To run IBM Host On-Demand, you must have a Java-enabled
web
browser such as Netscape Navigator or Microsoft Internet Explorer.
</applet>
```

Note: Further information on coding the parameter value is included in the Host On-Demand online help.

After completing the session information in the Deployment Wizard, or having modified the HTML, and refreshing the session in the browser, you will now see Custom Functions listed in the Category list box shown in Figure 5-9.

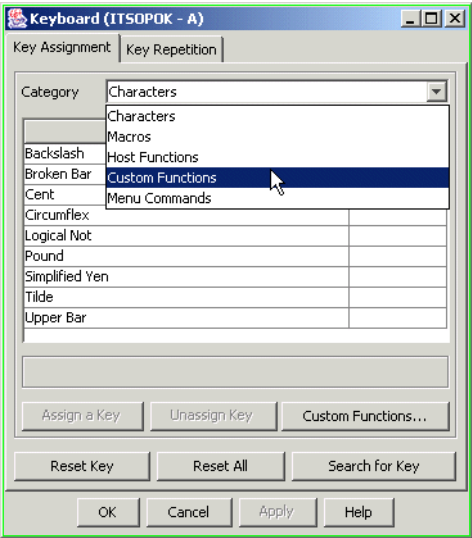


Figure 5-9 Custom Functions has been added to the category drop-down menu

Keys can now be assigned to the custom functions in the usual way, as shown in Figure 5-10.

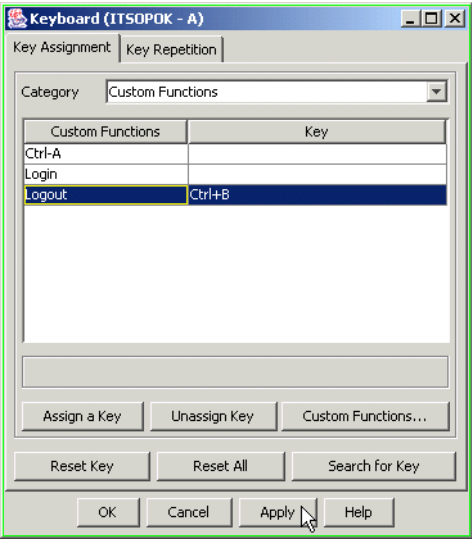


Figure 5-10 Mapping keyboard shortcuts to custom functions.

After this mapping has been saved, in this example, pressing Ctrl+B is the same as typing logoff followed by pressing the Enter key.

5.8 Improvements to Java 2 support

5.8.1 Terms defined

Please note the terms below and their meanings in this book.

Table 5-4 Terms defined

Term:	Meaning in this book:
Java 1	Refers to a Java 1.1.x JVM.
Java 2	Refers to a Java 1.3.x or Java 1.4.x JVM.
Java 1 class file	A class file produced by a Java 1 compiler.
Java 2 class file	A class file produced by a Java 2 compiler.
Java 1 browser	Netscape 4.x, or Internet Explorer with only its built-in Java 1 JVM
Java 2 enabled browser	Netscape 6.x or Internet Explorer with the Java 2 plug-in installed.
Java 1 cached client, Java 1 download client	The version of Host On-Demand compiled with a Java 1 compiler, and intended to be run primarily on a Java 1 browser.
Java 2 cached client, Java 2 download client	The version of Host On-Demand compiled with a Java 2 compiler, and intended to be run on a Java 2 enabled browser.

5.8.2 Java 2 support before Host On-Demand 7.0

Host On-Demand 6.0 enabled a download client or a cached client to run Netscape 6.x with the Java 2 plug-in. Version 6.0.3 allowed clients to run on Internet Explorer with the Java 2 plug-in.

However, the Host On-Demand Version 6.0 client code that ran on these Java 2 enabled browsers was exclusively Java 1 code, compiled with Java 1 compilers. There was as yet no Java 2 version of Host On-Demand.

In addition, some cached client features were not supported completely on Java 2 browsers:

- ▶ Delayed upgrades were not supported Java 2 cached clients.
- ▶ A Java 2 cached client configured with a preload list could not later download an additional module.

5.8.3 Java 2 support with Host On-Demand 7.0

In version 7.0, Host On-Demand greatly expanded its Java 2 support and added new features that take advantage of Java 2 capabilities.

Java 2 support now includes:

- ▶ Complete Java 1 and Java 2 versions of the Host On-Demand download client and cached client.

In the Java 2 versions, all the graphical user interface components are Swing (Java 2) components: panels, frames, menus, text fields, buttons, and so on.

- ▶ Swing-enabled bean components in the toolkit.

For example, the Terminal bean is now a JPanel that can be integrated into a Swing application.

- ▶ The ability to specify in the Deployment Wizard whether an HTML file can be run by Java 1 browsers, by Java 2 enabled browsers, or by either type of browser.

This setting is called the client Java type. For more information about the client Java type see “Client Java type: Java 1, Java 2, or Auto Detect” on page 199 and “Effect of client Java type at startup” on page 202.

At run time, the HTML file detects the type of browser and whether the Java 2 plug-in is available, looks at the client Java type, and determines whether to run the Java 1 version of Host On-Demand, run the Java 2 version, or take some other action.

- ▶ Support for Java 2 plug-ins from IBM, Sun, and Hewlett-Packard.
- ▶ Inclusion of an install image for the IBM Java 2 plug-in version 1.3.1 runtime for Win32.

Win32 Host On-Demand clients can download this install image from the Host On-Demand server, no matter what platform the server is running on.

- ▶ Java 2 detection built into the default HTML files, such as HOD_en.html, HODCached_en.html, and so on.
- ▶ The Deployment Wizard re-implemented to run as a Java 2 applet as well as an application, with usability and other improvements.
- ▶ Fixes for two limitations of Java 2 cached clients present in Host On-Demand 6.0 (see the previous section):
 - Delayed upgrades are now supported with Java 2 cached clients.
 - Subsequent downloads are now supported on a Java 2 cached client configured with a preload list.
- ▶ Increased flexibility with Java 2 cached clients.

Users who are running the cached client can switch among several different servers running different versions of Host On-Demand without having to remove and re-install the cached client.

5.8.4 Features that take advantage of Java 2

The following features of Host On-Demand, which were introduced in Version 7.0, are available to a Host On-Demand client only if:

- ▶ The client is running a Java-2-enabled browser, and
- ▶ The client Java type of the HTML file has been set to Java 2 or Auto Detect.

The features are:

- ▶ Visual settings and key remapping features that make computers more accessible to persons with physical disabilities. These changes affect the runtime and the Deployment Wizard.
- ▶ Auto IME (Input Method Editor) and on-the-spot conversion for DBCS languages.

Auto IME is the ability to switch between editing modes when moving from a DBCS field to a non-DBCS field or vice versa. On-the-spot conversion is the ability to select from among several closely related DBCS characters using a popup that appears in a location that is contiguous to the editing area.

- ▶ Screen print improvements: page header and footer and the ability to set margins, orientation, paper size, and paper source.
- ▶ The ability to start a session minimized -- useful for starting associated printer sessions.

For additional information on these features see the Host On-Demand online documentation.

5.8.5 Look and feel with Java 2 version of Host On-Demand

This section describes some of visible differences between the Java 1 version and the Java 2 version of the Host On-Demand runtime.

If you are running an HTML file that was created or edited with the Host On-Demand 7.0 Deployment Wizard, and the HTML file's client Java type is Java 2, and you are not seeing the differences listed below, then try clearing the browser's cache and restarting the browser.

- ▶ The window containing the session icons on the Host On-Demand desktop has a tab labeled Host On-Demand Client. The tab is on the left side of the upper edge of the window. See the figure below.

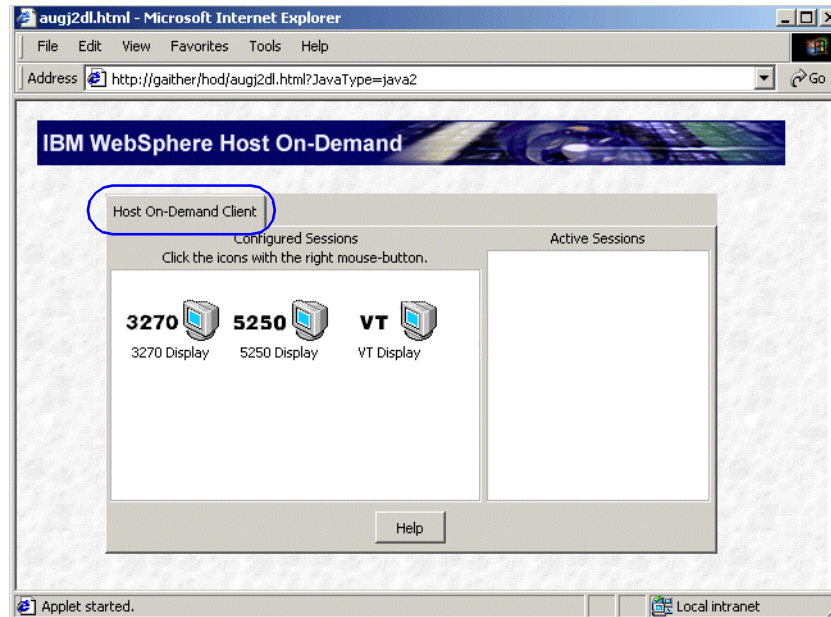


Figure 5-11 Java 2 window containing session icons has tab

- After a session is started, clicking File on the session panel's menu bar pops up a submenu which includes the option Print Screen Setup. The figure below shows a session panel with the Print Screen Setup menu option selected.

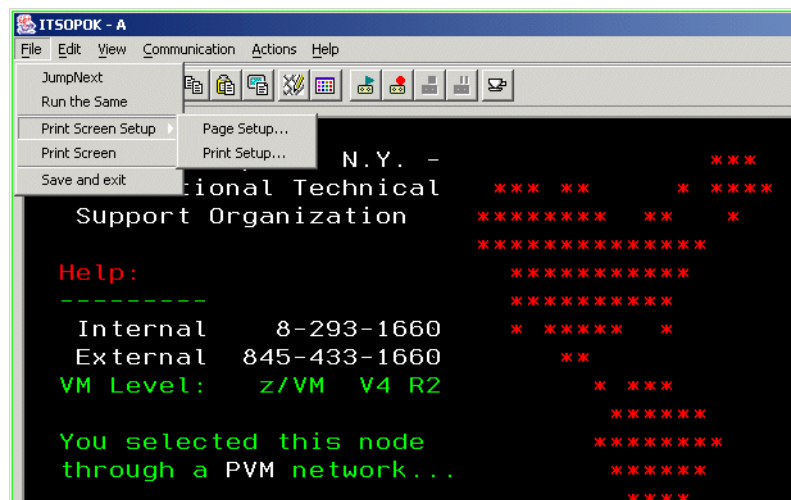


Figure 5-12 Java 2 session window with Print Screen Setup option

- After a session is started, moving the mouse pointer over a graphics image inside the session panel causes a small text popup to appear. This is an illustration of the capability of Java 2 to provide assistive information for computer users with physical handicaps. The figure below shows one of these popups with the text "Set up display colors".

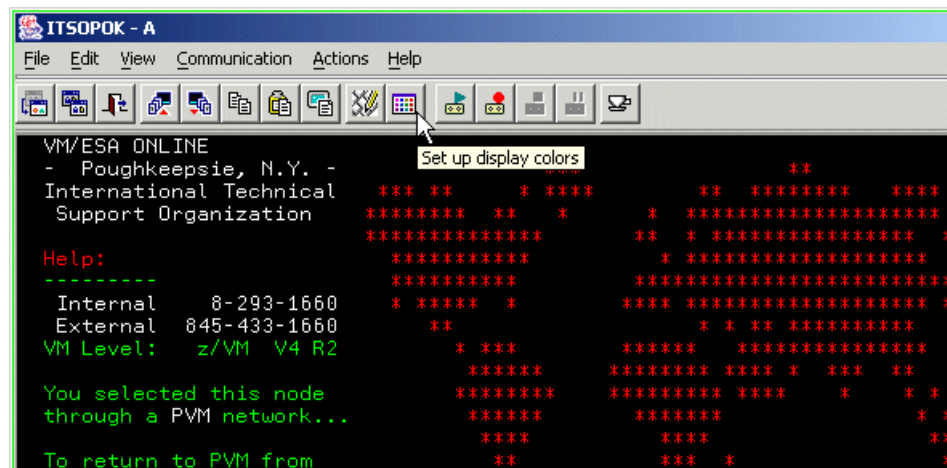


Figure 5-13 Java 2 session window with text popup over graphic image

- The Host On-Demand version information on the session panel's About box includes the word's Java 2. The figure below shows the About box. Note the words Java 2 at the bottom of the box as part of the version information.



Figure 5-14 About box for Java 2 version of Host On-Demand

- The Java 2 Plug-in's Java console includes the same version information as the About box, including the words Java 2. See Figure 5-30 on page 229.
- Different Java security message box.

The Java 2 security message box is different from the Java 1 security message box. Also, for Java 2 just one message box is popped up and is for all the privileges requested. In contrast, for Java 1 a separate message box is popped up at the time each type of privilege is requested. The figure below show the Java 2 security message box for the IBM Java 2 Plug-in 1.3.1 for Win32.

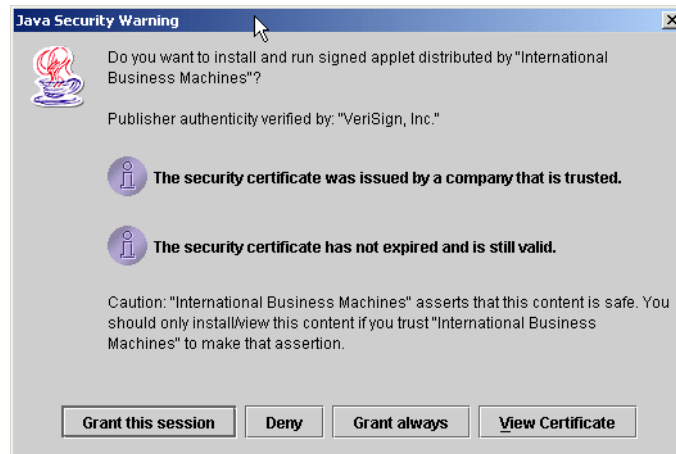


Figure 5-15 Java 2 Plug-in security warning

5.9 Java 2 practical issues

This section addresses practical issues involved in using Java 2 version of Host On-Demand.

The information in this section is relatively brief and at a high level. For more specific, detailed information:

- ▶ On client Java types (Java 1, Java 2, Auto Detect), see 5.10, “Client Java type: Java 1, Java 2, or Auto Detect” on page 199 and 5.11, “Effect of client Java type at startup” on page 202.
- ▶ On the Java 2 download client, see 5.14, “The Java 2 download client” on page 220.
- ▶ On the Java 2 cached client, see 5.13, “The Java 2 cached client” on page 210.
- ▶ On removing the Java 1 and Java 2 cached clients, see 5.13.7, “Removing the cached client” on page 217.
- ▶ On browsers, see 5.15, “Web browsers: Java 1 and Java 2 enabled” on page 221.

- ▶ On the Java 2 plug-in, see 5.16, “The Java 2 plug-in” on page 227.
- ▶ On the sticky cache, see 5.17.1, “More information on the Java 2 sticky cache” on page 236.

5.9.1 Advantages of switching clients to Java 2 enabled browsers

The advantages of having clients switch to Java 2 enabled browsers are:

- ▶ Your clients will be able to use the new features of Host On-Demand that are available when the client is running the Java 2 version of Host On-Demand. See “Features that take advantage of Java 2” on page 190.
- ▶ You will have support for the JVMs.

IBM, Sun, and Hewlett-Packard currently provide support (bug fixes) for their Java 2 JVMs.

In contrast, IBM and Sun no longer fix bugs in their Java 1 JVMs. However, Microsoft is continuing for now to support the Java 1 JVM that is included as a part of Internet Browser.

- ▶ You are better positioned for the future.

It is likely, although not a foregone conclusion, that in future releases of Host On-Demand IBM may eventually be obliged to implement some new features only in the Java 2 version of Host On-Demand, because of the extra cost of implementing new features in both the Java 2 and the Java 1 version.

5.9.2 Limitations and workarounds

This section describes limitations and workarounds in Java 2 support by Host On-Demand.

Cached client

The following limitation exist in Host On-Demand cached client support:

- ▶ Java 2 cached clients cannot be upgraded in the background. There is no workaround at present.

Download client

The following limitations exist in Host On-Demand download client support. For the reasons behind these limitations see “Reasons for three limitations on the Java 2 download client” on page 239.

1. When a Java 2 download client is configured with a preload list, no additional components can be downloaded later.

The workaround is to configure the client so that every component that may be needed is included in the preload list.

Note: components in the preload list are downloaded as JAR files. Because JAR files contain compressed data, components that are specified in the preload list are downloaded more quickly than loose class files would be downloaded.

2. The Function On-Demand client (HODThin.html) will not run correctly on Java 2-enabled Web browsers.

The workaround is to create a custom HTML file using the Deployment Wizard that includes all the components that the client will need.

3. The default download clients (such as HOD_en.html, HODCached_en.html, and so on) do not include some components. For more information see the online documentation.

The workaround is to create a custom HTML file using the Deployment Wizard that includes all the components that the client will need.

Miscellaneous

For additional limitations and workarounds, especially those having to do with particular versions of the Java 2 plug-in used with certain national languages, see the Host On-Demand online documentation and the readme file.

5.9.3 Effects on system resources

Running the Java 2 version of Host On-Demand affects the client system in the following ways:

- ▶ A slightly longer time is required before the Host On-Demand desktop appears.

The delay is due to the facts that:

- If the client Java type is Java 2 or Auto Detect, then the HTML file has to detect the browser type and determine whether the Java 2 plug-in is present.
- The Java 2 plug-in has to be loaded.

- ▶ Additional disk space is required if the Java 2 cached client attaches to multiple servers.

The reason is that the Java 2 cached client components are installed in the Java 2 plug-in's sticky cache, which stores a separate set of components for each server visited.

For example, if a user visits two servers running the same level of Host On-Demand, and runs an HTML file that is the same on both servers, then two sets of Host On-Demand components will be stored in the Java 2 sticky cache. Each set will be associated with one of the servers and will be re-used if the user visits the server again.

5.9.4 Must I migrate my existing HOD 6.0 Deployment Wizard files?

Migration in this section refers to using the Host On-Demand 7.0 Deployment Wizard to regenerate HTML files that were created using the Host On-Demand 6.0 Deployment Wizard.

- ▶ You do not need to migrate HTML files that are used only by clients running Java 1 browsers.

Specifically we are referring to the following situation:

- Your clients have been running Java 1 browsers and will continue to do so.
- You created HTML files using Host On-Demand 6.0 Deployment Wizard.
- Your clients have been using these files to connect with a Host On-Demand 6.0 server.
- Your clients now are going to use these files to connect with a Host On-Demand 7.0 server.

In this situation, you do not need to migrate your files using the Host On-Demand 7.0 Deployment Wizard.

- ▶ IBM recommends that you migrate HTML files that are used by clients running Java 2 enabled browsers.

Specifically we are referring to the following situation:

- Your clients have been running Java 2 enabled browsers and will continue to do so.
- You created HTML files using Host On-Demand 6.0 Deployment Wizard.
- Your clients have been using these files to connect with a Host On-Demand 6.0 server.
- Your clients now are going to use these files to connect with a Host On-Demand 7.0 server.

In this situation, you actually do not need to migrate your files using Host On-Demand 7.0 Deployment Wizard. Your files will continue to function as they did when your clients attached to a Host On-Demand 6.0 server.

Unfortunately, this level of functionality means that the Java 2 enabled browsers will download Host On-Demand 7.0 Java 1 code modules, not Java 2 code modules. Your clients will not be able to use any of the Java 2 functionality in Host On-Demand 7.0.

If you want your clients to use any Java 2 functionality in Host On-Demand 7.0, then you must migrate the HTML files that you created with Host On-Demand 6.0.

For these reasons, IBM recommends in this situation that you migrate your HTML files.

5.9.5 What if I want to continue running Java 1 browsers only?

If your clients are running Java 1 browsers only, and you want to continue that practice for the time being, then you have only a few tasks, or none, to perform.

- ▶ Your clients should already be running Netscape 4.x or Internet Explorer without the Java 2 plug-in.
- ▶ You do not have to migrate your existing HTML files that you created with the Deployment Wizard from Host On-Demand 6.0. These files will continue to run with Host On-Demand 7.0. See “Must I migrate my existing HOD 6.0 Deployment Wizard files?” on page 196.
- ▶ If you do want or need to migrate your existing HTML files, use the Deployment Wizard from Host On-Demand 7.0. On the Additional Functions page, set the Client Java Type field to Java 1.
- ▶ If your clients use one of the default HTML files, such as HOD_en.html or HODCached_en.html, you can improve start up time by making the following change. Edit the HTML files with a text editor as follows:

-- Find the JavaScript line

```
var hod_JavaType = 'detect' ;
```

-- Change it to

```
var hod_JavaType = 'java1' ;
```

This change will cause the HTML file to immediately launch the Host On-Demand applet on the browser's Java 1 JVM rather than try to detect whether the browser is Java 1 or Java 2 enabled.

- ▶ On new workstations:
 - Verify that a Java 1 browser is installed.
 - Verify that the Java 1 browser has access to a Java 1JVM. See 5.15, “Web browsers: Java 1 and Java 2 enabled” on page 221.

5.9.6 What if I am already running Java 2 enabled browsers?

If some or all of your clients are running Java 2 enabled browsers, then you may have a few tasks to perform.

- ▶ Your clients should already be running a Java 2 capable browser with a Java 2 plug-in.
- ▶ If necessary, for client machines running Internet Explorer, use the Java 2 Plug-in Control Panel to verify that the default JVM for Internet Explorer is NOT set to the Java 2 plug-in. For more information see “Default JVM for Internet Explorer must be MS Java 1 JVM” on page 224.
- ▶ Although your existing Java 2 HTML files would run on Host On-Demand 7.0, they would continue to download Java 1 modules just as they did in Host On-Demand 6.0. Therefore you should migrate your existing HTML files. See 5.9.4, “Must I migrate my existing HOD 6.0 Deployment Wizard files?” on page 196.
- ▶ If you want to or need to migrate your existing HTML files, use the Host On-Demand 7.0 Deployment Wizard. On the Additional Options page:
 - Set the Client Java Type field to Java 2 if ALL your clients are using Java 2 enabled browsers, or
 - Set the Client Java Type field to Auto Detect if some of your clients are using Java 2 enabled browsers and others are using Java 1 browsers, or if you are not sure.
- ▶ Users should remove any previous Java 1 cached client or Java 2 cached client from their workstations.
- ▶ On new workstations:
 - Verify that a Java 2 capable browser is installed.
 - Verify that a Java 2 plug-in is installed.
 - For Internet Explorer, use the Java 2 Plug-in Control panel to verify that Internet Explorer's default JVM is NOT set to the Java 2 plug-in. For more information see “Default JVM for Internet Explorer must be MS Java 1 JVM” on page 224.

5.9.7 What if I want to migrate my users to Java 2 enabled browsers?

If all your clients are running Java 1 browsers, and you want to change some or all of your clients to Java 2 enabled browsers, you have a few tasks to perform.

- ▶ You probably want your clients that are changing to Java 2 enabled browsers to download the Java 2 version of Host On-Demand, rather than to download Java 1 modules as in Host On-Demand 6.0. Therefore, you should migrate your existing HTML files. See “Must I migrate my existing HOD 6.0 Deployment Wizard files?” on page 196.
- ▶ If you want to or need to migrate your existing HTML files, use the Deployment Wizard from Host On-Demand 7.0. On the Additional Options page:

- Set the Client Java Type field to Java 2 if ALL your clients are using Java 2 enabled browsers, or
- Set the Client Java Type field to Auto Detect if some of your clients are using Java 2 enabled browsers and others are using Java 1 browsers, or if you are not sure.
- ▶ Users should remove any previous Java 1 cached client or Java 2 cached client from their workstation.
- ▶ On all workstations that will use Java 2:
 - Verify that a Java 2 capable browser is installed.
 - Verify that a Java 2 plug-in is installed.
 - If necessary, for client machines running Internet Explorer, use the Java 2 Plug-in Control Panel to verify that Internet Explorer's default JVM is NOT set to the Java 2 plug-in. For more information see “Default JVM for Internet Explorer must be MS Java 1 JVM” on page 224.

5.10 Client Java type: Java 1, Java 2, or Auto Detect

Host On-Demand 7.0 added to the Deployment Wizard the concept of client Java type. The possible settings for client Java type are: Java 1, Java 2, and Auto Detect.

5.10.1 Overview

The Client Java Type field appears on the Additional Options page of the Deployment Wizard. This page is shown in the figure below. The client Java type is set to Java 1.

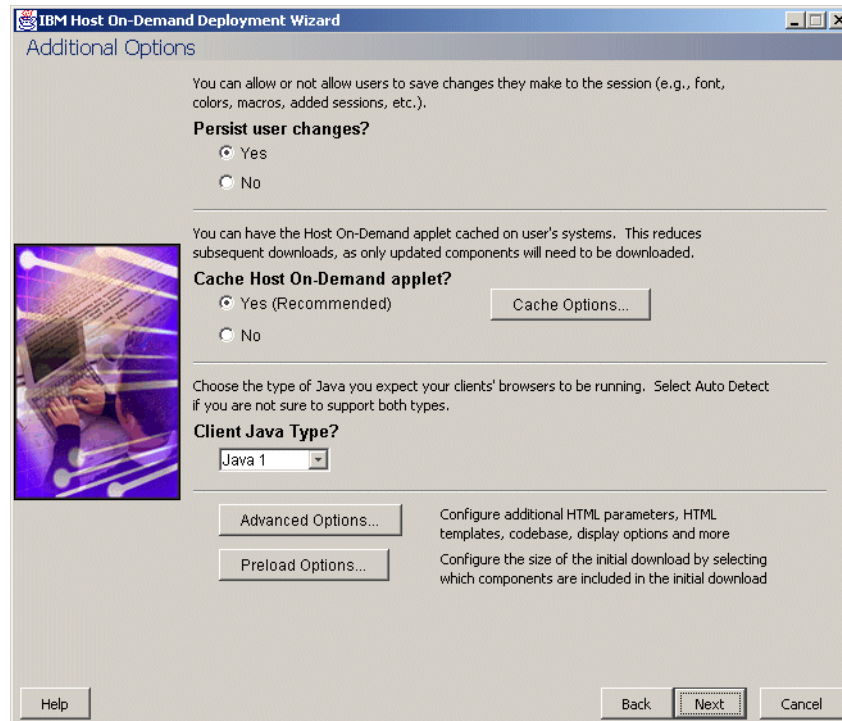


Figure 5-16 Deployment Wizard page showing Client Java Type field

The following sections discuss the client Java types. For the specific effect that each of these settings has at startup see 5.11, “Effect of client Java type at startup” on page 202.

5.10.2 Java 1

Choose this setting if you know that all your Host On-Demand clients are running Java 1 browsers.

Choosing this setting instead of Auto Detect will save your clients the small but, for slower systems, appreciable time that would be required for the HTML file to detect whether the client browser type is Java 1 or Java 2 enabled.

In one unusual situation you must choose the Java 1 setting rather than Auto Detect. This situation, which is also described in the online documentation, is:

- ▶ Your deployment uses the Configuration server-based model.
- ▶ Your server has been running Host On-Demand 6.0.x or earlier.
- ▶ You are now installing Host On-Demand 7.0 on the server.

- You want to use cached client controls so that not all Java 1 clients are upgraded on their first try (deferred upgrades).

In this situation, you must specify Java 1 rather than Auto Detect. If you specify Auto Detect, your clients will not be able to run Host On-Demand properly. Alternatively, you can change your HTML file settings not to use deferred upgrades.

After all your clients have upgraded to Host On-Demand 7.0, you are no longer exposed to this problem. At this point you can change the client Java type of the HTML file from Java 1 to Auto Detect if you wish.

5.10.3 Java 2

Choose this setting if you know that all your Host On-Demand clients are running Java 2 enabled browsers.

Choosing this setting instead of Auto Detect will save your clients the small but, for slower systems, appreciable time that would be required for the HTML file to detect whether the client browser type is Java 1 or Java 2 enabled.

Choosing this setting will also require your users, or in some scenarios merely remind them, to use a Java 2 enabled browser in order to take advantage of the capabilities of the Java 2 version of Host On-Demand.

Note: Occasionally a problem occurs that causes the HTML file to fail to detect the Java 2 plug-in when it is in fact present. As a result the HTML file behaves at startup as if the Java 2 plug-in were not installed. The workaround is to retry the operation. The detection usually succeeds on the second try.

5.10.4 Auto detect

Choose this setting if you know that some of your Host On-Demand clients are using Java 1 browsers while other clients are using Java 2 enabled browsers, or if you are not sure.

Choosing this setting instead of Java 1 or Java 2 will allow both types of clients to use Host On-Demand, but will also impose a small but, for slower systems, appreciable delay while the HTML file detects whether the client browser type is Java 1 or Java 2 enabled.

For an unusual situation in which you must use Java 1 instead of Auto Detect see section 5.10.2, “Java 1” on page 200.

5.11 Effect of client Java type at startup

The tables in this section show what occurs at startup for each client Java type. The results depend not only on the client Java type but also on the browser type and whether the client is a download client or a cached client.

These tables assume that Internet Explorer's default JVM is set to the Microsoft Java 1 JVM. For more information on this topic see "Default JVM for Internet Explorer must be MS Java 1 JVM" on page 224.

5.11.1 Messages

The figure below shows the message displayed when:

- ▶ The client Java type is Java 1 or Java 2.
- ▶ The browser is Netscape 6.x without a Java 2 plug-in.
- ▶ The client is a download client or a cached client.



Click here to get the plugin

Figure 5-17 Netscape 6.x message prompting user to download a Java 2 plugin

The next figure shows the messages displayed when:

- ▶ The client Java type is Java 2.
- ▶ The browser is Netscape 4.x or Internet Explorer without the Java 2 plugin.
- ▶ The client is a download client or a cached client.
- ▶ The platform is Win32.

This figure shows the message for the Win32 platform:

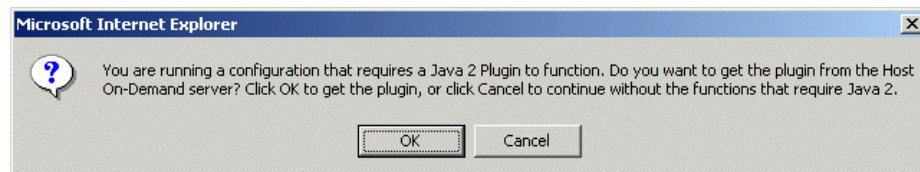


Figure 5-18 You are running a configuration that requires a Java 2 Plug-in to function

A similar message is shown in the same situation for the non-Win32 platform, but the user is told to contact the system administrator in order to get the plug-in.

Example 5-2 Non-Win32 platform message

You are running a configuration that requires a Java 2 Plugin to function. Please contact your administrator to obtain the necessary Java 2 Support. Host On-Demand will continue without the functions that require Java 2.

The figure below shows the message displayed when:

- ▶ The client Java type is Java 1.
- ▶ The browser is Netscape 6.x with a Java 2 plug-in.
- ▶ The client is a cached client.

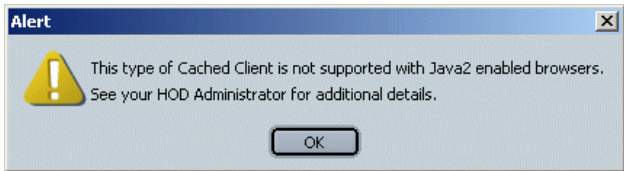


Figure 5-19 This type of cached client is not supported with Java 2 enabled browsers

5.11.2 Startup behavior for Java 1 download client

The table below shows the startup behavior for the Java 1 download client based on the client Java type and the browser type.

Table 5-5 Startup behavior for Java 1 download client

Client Java type in Deployment Wizard:	Browser type:	Result:
Java 1 or Auto Detect	Netscape 4.x	HTML file runs Java 1 download client
Java 1 or Auto Detect	Internet Explorer without Java 2 plug-in	HTML file runs Java 1 download client
Java 1	Internet Explorer with Java 2 plug-in	HTML file runs Java 1 download client
Java 1 or Auto Detect	Netscape 6.x without Java 2 plug-in	Netscape 6.x displays warning message in Figure 5-17 on page 202. User options: - Quit; or - Click to download Sun Java 2 plug-in

Client Java type in Deployment Wizard:	Browser type:	Result:
Java 1	Netscape 6.x with Java 2 plug-in	HTML file runs Java 1 download client

5.11.3 Startup behavior for Java 2 download client

The table below shows the startup behavior for the Java 2 download client based on the client Java type and the browser type.

Table 5-6 Startup behavior for Java 2 download client

Client Java type in Deployment Wizard:	Browser type:	Result:
Java 2	Netscape 4.x	HTML file displays warning message, see Figure 5-18 on page 202. User options if Win32 platform: - Cancel; or - Run Java 1 download client; or - Download IBM Java 2 plug-in for Win32 User options if non-Win32 platform - Cancel; or - Run Java 1 download client
Java 2	Internet Explorer without Java 2 plug-in	HTML file displays warning message, see Figure 5-18 on page 202. User options if Win32 platform: - Cancel; or - Run Java 1 download client; or - Download IBM Java 2 plug-in for Win32 User options if non-Win32 platform - Cancel; or - Run Java 1 download client
Java 2 or Auto Detect	Internet Explorer with Java 2 plug-in	HTML file runs Java 2 download client
Java 2 or Auto Detect	Netscape 6.x without Java 2 plug-in	Netscape 6.x displays warning message, see Figure 5-17 on page 202. User options: - Quit; or - Download Sun Java 2 plug-in

Client Java type in Deployment Wizard:	Browser type:	Result:
Java 2 or Auto Detect	Netscape 6.x with Java 2 plug-in	HTML file runs Java 2download client

5.11.4 Startup behavior for Java 1 cached client

The table below shows the startup behavior for the Java 1 cached client based on the client Java type and the browser type.

Table 5-7 Startup behavior for Java 1 cached client

Client Java type in Deployment Wizard:	Browser type:	Result:
Java 1 or Auto Detect	Netscape 4.x	HTML file installs Java 1 cached client. User restarts browser. HTML file launches Java 1 cached client.
Java 1 or Auto Detect	Internet Explorer without Java 2 plug-in	HTML file installs Java 1 cached client. User restarts browser. HTML file launches Java 1 cached client.
Java 1	Internet Explorer with Java 2 plug-in	HTML file installs Java 1 cached client. User restarts browser. HTML file launches Java 1 cached client.
Java 1 or Auto Detect	Netscape 6.x without Java 2 plug-in	Netscape 6.x displays warning message, see Figure 5-17 on page 202. User options: - Quit; or - Click to download Sun Java 2 plug-in
Java 1	Netscape 6.x with Java 2 plug-in	HTML file: - Displays error message, see Figure 5-19 on page 203. - Refuses to install cached client. User option: - Quit, then see system administrator.

5.11.5 Startup behavior for Java 2 cached client

The table below shows the startup behavior for the Java 2 cached client based on the client Java type and the browser type.

Table 5-8 Startup behavior for Java 2 cached client

Client Java type in Deployment Wizard:	Browser type:	Result:
Java 2	Netscape 4.x	1) HTML file displays warning message, see Figure 5-18 on page 202. User options if Win32 platform: - Cancel; or - Choose to install Java 1 cached client; or - Download IBM Java 2 plug-in for Win32 User options if non-Win32 platform - Cancel; or - Choose to install Java 1 cached client 2) Assume: user chooses to install Java 1 cached client. 3) HTML file installs and launches Java 1 cached client.
Java 2	Internet Explorer without Java 2 plug-in	1) HTML file displays warning message, see Figure 5-18 on page 202. User options if Win32 platform: - Cancel; or - Choose to install Java 1 cached client; or - Download IBM Java 2 plug-in for Win32 User options if non-Win32 platform - Cancel; or - Choose to install Java 1 cached client 2) Assume: user chooses to install Java 1 cached client. 3) HTML file installs and launches Java 1 cached client.
Java 2 or Auto Detect	Internet Explorer with Java 2 plug-in	HTML file installs and launches Java 2 cached client.
Java 2 or Auto Detect	Netscape 6.x without Java 2 plug-in	Netscape 6.x displays warning message, see Figure 5-17 on page 202. User options: - Quit; or - Click to download Sun Java 2 plug-in
Java 2 or Auto Detect	Netscape 6.x with Java 2 plug-in	HTML file installs and launches Java 2 cached client.

5.12 Download client and cached client implementation

This section describes:

- ▶ The applets that are launched for the download client and the cached client.
- ▶ How the Host On-Demand components are stored for the download client and the cached client.

For more details see 5.17.4, “More information on launching the Host On-Demand applets” on page 241.

5.12.1 HostOnDemand applet and CachedAppletSupport applet

The HostOnDemand applet

The HostOnDemand applet is the applet that is launched for a download client by the HTML file. The actual class file for the Java 1 version of Host On-Demand is HostOnDemand.class, and the actual class file for the Java 2 version of Host On-Demand is a separate module that is also named HostOnDemand.class. When the HostOnDemand applet is launched the applet puts up the appropriate version (Java 1 or Java 2) of the Host On-Demand desktop and manages the desktop and sessions.

When the download client is running, if the JVM is Java 1, and there is a preload list, and an additional component is needed, the browser downloads from the server the loose class files that make up the component. In contrast, if the JVM is Java 2, no additional components can be downloaded. See 5.14, “The Java 2 download client” on page 220.

The CachedAppletSupport applet

The CachedAppletSupport applet is the applet that is launched for a cached client by the HTML file. The actual class file for the Java 1 version of Host On-Demand is CachedAppletSupportApplet.class, and the actual class file for the Java 2 version of Host On-Demand is CachedAppletLoader.class.

When the CachedAppletSupport applet is launched, it checks whether the cached client components have been installed. If not, then CachedAppletSupport applet installs the cached client components. Then it either tells the user to restart the browser in order to start the cached client (if this is the Java 1 CachedAppletSupport applet) or else it immediately starts the cached client (if this is the Java 2 CachedAppletSupport applet). To start the cached client, the CachedAppletSupport applet launches the HostOnDemand applet, the same applet that is used for the download client.

When the cached client is running, whether the JVM is Java 1 or Java 2, if there is a preload list, and an additional component is needed, the CachedAppletSupport applet arranges for the component to be downloaded from the server. Then the user must restart the browser.

5.12.2 How Host On-Demand component modules are stored

A component is a functional unit, such as 3270 Display Sessions or 3270 Printer Sessions. A component is made up of one or possibly more than one downloadable module.

A downloadable module may be:

- ▶ A signed archive file, such as a JAR file or a CAB file, that contains a collection of related individual Java class files; or
- ▶ An individual Java class file. The class file may be a Java 1 class file (that is, created by a Java 1 compiler) or a Java 2 class file (that is, created by a Java 2 compiler).

The following table summarizes how the modules that make up Host On-Demand components are downloaded and stored for the Java 1 and Java 2 download clients and cached clients.

Table 5-9 How components are downloaded and stored

JVM:	Download client:	Cached client:
Java 1	Components in the preload list are downloaded as JAR or CAB files containing Java 1 class files.	Components in the preload list are downloaded as JAR or CAB files containing Java 1 class files..
	Components not in the preload list are downloaded as Java 1 class files.	Components not in the preload list are likewise downloaded as JAR or CAB files containing Java 1 class files
	Downloaded JAR or CAB files are not unpacked.	For Internet Explorer, Host On-Demand unpacks the class files from each CAB file. For Netscape 4.x, the class files are not unpacked from the JAR files.
	The modules reside in the browser cache.	The modules reside in a user directory.
	Each component, whether in the preload list or downloaded as needed, is downloaded anew each time the download client is run.	Each component, whether in the preload list or downloaded as needed, is downloaded once.

JVM:	Download client:	Cached client:
Java 2	-- Components in the preload list are downloaded as JAR files containing Java 2 class files.	-- Components in the preload list are downloaded as JAR files containing Java 2 class files.
	-- Components not in the preload list cannot be downloaded later.	-- Components not in the preload list are likewise downloaded as JAR files containing Java 2 class files.
	-- The class files are not unpacked from the JAR files.	-- The class files are not unpacked from the JAR files.
	-- Components reside in the Java 2 plug-in's temporary cache (not in the sticky cache) .	-- Components reside in the Java 2 plug-in's sticky cache.
	-- Each component is downloaded anew each time the download client is run.	-- Each component, whether in the preload list or downloaded as needed, is downloaded once.

Download client

For the Java 1 download client, components in the preload list are downloaded as JAR (for Netscape) or CAB (for Internet Explorer) files containing Java 1 class files. The class files remain in the JAR or CAB files. In contrast, components not in the preload list are downloaded as individual Java 1 class files. All the downloaded modules, including JAR or CAB files and loose class files, reside in the browser cache. An example of a JAR and a CAB file are habasen.jar and habasen.cab.

For the Java 2 download client, components in the preload list are downloaded as JAR files containing Java 2 class files. The class files are not unpacked but rather remain in the JAR files. Components not in the preload list cannot be downloaded later. The JAR files reside in the Java 2 plug-in's temporary cache (not the sticky cache). These Java 2 JAR files have names that are similar to the names of the Java 1 JAR files but that are distinguished by a '2' appended to the file name. An example is habasen2.jar.

For both Java 1 and Java 2 enabled browsers, each component, whether in the preload list or downloaded as needed, is downloaded anew each time the download client is run.

When the Java 1 or Java 2 download client is running, and the JVM needs to find a Host On-Demand class file, the JVM first looks among the loose class files if there are any. If the class file is not found, the JVM then looks among the Host On-Demand JAR files until it finds the correct one and then finds the class file within the JAR file.

The Java 1 and Java 2 download clients do not interfere with each other because the Java 1 downloaded modules and the Java 2 downloaded modules have different names and are stored in different places.

Cached client

The Java 1 and Java 2 cached clients use the same modules as their download client counterparts, but store the modules differently.

For the Java 1 cached client, components in the preload list are downloaded as JAR or CAB files containing Java 1 class files. Components not in the preload list are likewise downloaded as JAR or CAB files containing Java 1 class files. For Internet Explorer, Host On-Demand unpacks the class files from each CAB file. For Netscape 4.x, the class files are not unpacked from the JAR files. The files, whether loose class files or JAR files, reside in a user directory. But the user directory is different for Internet Explorer than for Netscape 4.x.

For the Java 2 cached client, components in the preload list are downloaded as JAR files containing Java 2 class files. Components not in the preload list are likewise downloaded as JAR files containing Java 2 class files. The class files are not unpacked but remain in the JAR files. The JAR files reside in the Java 2 plug-in's sticky cache.

For both Java 1 and Java 2 enabled browsers, each component, whether in the preload list or downloaded as needed, is downloaded once.

The Java 1 and Java 2 cached clients do not interfere with each other because the Java 1 downloaded modules and the Java 2 downloaded modules have different names and are stored in different places.

For more information on how the modules are stored for the Java 1 and Java 2 cached clients see 5.13.5, "Handling cached client components for Java 1 and Java 2" on page 214.

5.13 The Java 2 cached client

When the Java 2 cached client is started the first time it displays a message similar to the one shown in the figure below. The text is the same whether the cached client is for Java 1 or Java 2, but the download sizes are different for Java 1 and Java 2. The figure below shows the message displayed by Internet Explorer for the Java 2 cached client:

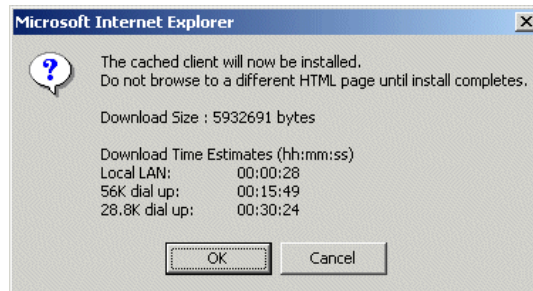


Figure 5-20 Install cached client

However, when the user presses OK and the installation begins, the Java 2 cached client does NOT display the progress indicator frame displayed by the Java 1 cached client. Instead, the words "Loading Java Applet ..." appear in the middle of the Host On-Demand desktop. Also, some type of indicator may appear as each component is downloaded. For example, the following indicator is displayed when a component is downloaded by Internet Explorer running with the IBM Java 2 plug-in:

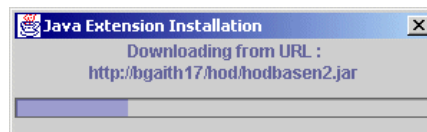


Figure 5-21 Indicator displayed by Internet Explorer running with Java 2 plug-in

When the installation is complete the Host On-Demand applet is launched immediately. The user does not have to restart the browser.

5.13.1 Java 2 cache options

Like the Java 1 cached client, the Java 2 cached client supports the following cache options:

- ▶ Control of user upgrades.
- ▶ Debug cached client installation process.

Unlike the Java 1 cached client, the Java 2 cached client does not support the following cache option, even though it is selectable in the Deployment Wizard:

- ▶ Upgrade in the background.

5.13.2 Downloading a Java 2 component not on the preload list

If the user attempts to use a Java 2 component not on the preload list, a message is displayed. For example, the figure below shows the message displayed when the Run Applet component is not on the preload list and the user starts a session and clicks File, Run Applet. This is the message displayed with Java 2, but the same message would be displayed with Java 1.

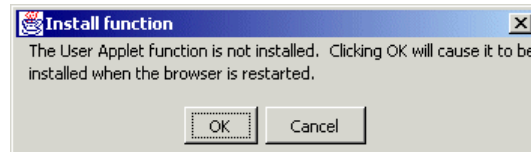


Figure 5-22 Message displayed for component that is not on the preload list

When the user clicks OK, Host On-Demand modifies the caching list so that the component will be downloaded when the browser is restarted. The user must restart the browser.

The following table compares this operation on the Java 1 version of Host On-Demand and on the Java 2 version:

Table 5-10 Downloading a component not on the preload list.

Item:	Java 1 cached client:	Java 2 cached client:
OK/Cancel message is displayed:	Yes	Yes
Update method:	Component is downloaded immediately.	Caching list is updated. Component is downloaded when browser is restarted.
User must restart browser:	Yes	Yes

5.13.3 Java 2 cached client does not interfere with download client

The Java 2 cached client does not interfere with the Java 2 download client. That is, the user can run the Java 2 download client without first removing the Java 2 cached client.

The reason is that the Java 2 download client's components are stored in the Java 2 plug-in's temporary cache, while the Java 2 cached client's components are stored in the Java 2 plug-in's sticky cache. For more information about how component modules are stored see 5.12.2, "How Host On-Demand component modules are stored" on page 208.

5.13.4 Java 2 cached client upgrades

When the Java 2 cached client detects a newer version of Host On-Demand on the server, a message such as the one in the figure below is displayed. This is the message displayed when the cached client is running on Internet Explorer:

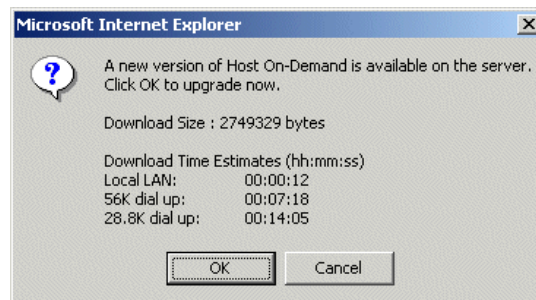


Figure 5-23 Upgrade message

The upgrade takes place in the foreground. When the upgrade is complete, the Host On-Demand desktop is displayed. The browser does not have to be restarted.

Upgrading in the background not supported.

The Java 2 cached client does not support upgrading in the background, even though the option is selectable in the Deployment Wizard.

Avoiding an extra download going from Java 1 to Java 2

If you are migrating users from Java 1 browsers to Java 2 browsers, and you are also upgrading the level of Host On-Demand on the server from Host On-Demand 6.x to Host On-Demand 7.0, then you can prevent your users from having to download their cached clients twice by migrating users to Java 2 browsers before they connect to the Host On-Demand 7.0 server.

Compare the following two procedures.

1. This procedure would cause users to have to download a new cached client twice:
 - The user is running a Java 1 browser.

- The user already has a Java 1 cached client installed from Host On-Demand 6.0.
 - The user connects to a server HODSRV1 running Host On-Demand 7.0. At this point, the Host On-Demand 7.0 Java 1 cached client has to be downloaded.
 - Now the system administrator installs a Java 2 enabled browser and a Java 2 plug-in on the user's machine.
 - The user again connects to HODSRV1. At this point the Host On-Demand 7.0 Java 2 cached client has to be downloaded. This is the second download.
2. In contrast, the following procedure would cause users to have to download a new cached client only once:
- The user is running a Java 1 browser.
 - The user already has a Java 1 cached client installed from Host On-Demand 6.0.
 - Now the system administrator installs a Java 2 enabled browser on the user's machine.
 - The user connects to a server HODSRV1 running a newer version of Host On-Demand. At this point the Host On-Demand 7.0 Java 2 cached client has to be downloaded. This is the first and only download.

5.13.5 Handling cached client components for Java 1 and Java 2

The table below summarizes how the Host On-Demand components are handled by the Java 1 and Java 2 cached clients.

Table 5-11 How components are handled by the Java 1 and Java 2 cached clients

Item:	Java 1 cached client:	Java 2 cached client:
Modules are downloaded as:	JAR or CAB files containing class files compiled by the Java 1 compiler.	JAR files containing class files compiled by the Java 2 compiler.
Where the Java class files are kept:	Unpacked as loose class files in a browser-specific user work area (Internet Explorer) or packed in JAR files in a browser-specific user work area (Netscape 4.x).	Packed in JAR files in the Java 2 sticky cache.
Cache management file name:	HOD_CCR.ccr	server.dirname.HOD_CCR2.ccr

Item:	Java 1 cached client:	Java 2 cached client:
Are components associated with a server?	No	Yes

Java 1 cached client

For the Java 1 cached client each component is downloaded as one or more signed archives (a JAR file for Netscape, a CAB file for Internet Explorer).

For Internet Explorer, the class files are unpacked from the CAB files and stored in a user work area under a subdirectory called HODCC. On Windows 2000 the HODCC subdirectory is located in a path such as the following:

```
c:\Documents and Settings\JASmith\HODCC
```

where JASmith is the user name. The individual class files are placed in appropriate subdirectories under the HODCC subdirectory depending on the complete name of the Java package to which the class files belong.

For Netscape 4.x, the class files are not unpacked but remain in the JAR files. The JAR files are stored under a directory in a user work area. On Windows 2000 the directory is located in a path such as the following

```
c:\Program Files\Netscape\Users\JASmith\cache
```

where JASmith is the user name.

The component name, file version, and other information about each downloaded JAR or CAB file are stored in a file named HOD_CCR.ccr which resides in a subdirectory close to the data. On Windows 2000 the HOD_CCR.ccr file for Internet Explorer is located in a path such as the following:

```
c:\Documents and Settings\JASmith\HOD_CCR.ccr
```

where JASmith is the user name. On Windows 2000 the HOD_CCR.ccr file for Netscape 4.x is located in a path such as the following:

```
c:\Program Files\Netscape\Users\HOD_CCR.ccr
```

Java 2 cached client

For the Java 2 cached client each component is downloaded as one or more Java 2 JAR files (whether the browser is Netscape 6.x or Internet Explorer with the Java 2 plug-in). The JAR file or files is placed in the Java 2 plug-in's sticky cache. A separate copy of each JAR file is maintained in the sticky cache for each Host On-Demand server that the user visits. For more information on the

sticky cache see “More information on the Java 2 sticky cache” on page 236. For both Netscape 6.x and Internet Explorer with the Java 2 plug-in, the JAR file is not unpacked into loose class files, and therefore a HODCC directory is not created.

The module name, file version number, and other information about each downloaded JAR file are stored in a file named `server.dirname.HOD_CCR2.ccr`, where `server` is the Host On-Demand server's name and `dirname` is the name of the server's public directory. For example, the file might be named `HODSRV1.hod.HOD_CCR2.ccr`, where `HODSRV1` is the server's TCP/IP hostname and `hod` is the server's public directory. This file is located in the same directory as the Java 1 `HOD_CCR.ccr` file is located for the Java 1 cached client for Internet Explorer. On Windows 2000 the `HOD_CCR2.ccr` file is located on a path such as:

```
c:\Documents and Settings\JASmith\HODSRV1.hod.HOD_CCR2.ccr
```

where `JASmith` is the user name. The same path is used for both Netscape 6.x and Internet Explorer with the Java 2 plug-in.

For Java 2 there will be a separate `server.dirname.HOD_CCR2.ccr` file for each Host On-Demand server visited by the user, just as there are separately downloaded JAR files in the sticky cache for each Host On-Demand server visited.

5.13.6 Increased flexibility with Java 2 cached clients

Because the Java 2 sticky cache stores separate JAR files and separate `HOD_CCR2.ccr` files for each Host On-Demand server visited, the Java 2 cached client is not exposed to the same problems that the Java 1 cached client is exposed to when the user visits several Host On-Demand servers running different service levels of Host On-Demand. See “Java 1 cached client support across the Internet” on page 169.

Users who are running the Java 2 cached client can switch among several different servers running different service levels of Host On-Demand without having to remove and re-install the cached client.

Also, the user can switch back and forth between download and cached clients without having to remove the cached client code.

5.13.7 Removing the cached client

Why does the Java 1 cached client need to be removed?

Situations requiring removal

There are at least two situations in which Java 1 cached client would need to be removed:

1. If the client is running a Java 1 browser, and the cached client is installed, then the download client cannot be run until the cached client is removed.

For the reason for this limitation see “Reason for restriction on Java 1 download client” on page 239.

2. Certain upgrade scenarios require the cached client to be removed.

For one such scenario see “Scenario requiring Java 1 cached client to be removed” on page 238.

Situations not requiring removal

The Java 1 cached client does not need to be removed in order for the Java 2 cached client to be installed. As described in 5.13.5, “Handling cached client components for Java 1 and Java 2” on page 214, the components and caching information for the Java 1 cached client are maintained in different files than the components and caching information for the Java 2 cached client.

However, if a user migrates to a Java 2 enabled browser and does not plan to use the Java 1 cached client data again, then the user might want to remove the Java 1 cached client in order to free up a few megabytes of disk space.

Why does the Java 2 cached client need to be removed?

As mentioned earlier, the Java 2 cached client does not need to be removed in order for the Java 2 download client to be run. See section 5.13.3, “Java 2 cached client does not interfere with download client” on page 212.

Nor does the Java 2 cached client need to be removed in order for an earlier version of the cached client modules to be downloaded from a different server. The reason that this problem does not exist for the Java 2 cached client is that the Java 2 plug-in downloads a separate set of components for each server visited. See section 5.13.5, “Handling cached client components for Java 1 and Java 2” on page 214.

The main scenario in which the Java 2 cached client needs to be removed is when the Java 2 plug-in's sticky cache becomes full.

Another scenario in which the the user would want to remove the cached client is for disk cleanup.

Which removal option?

The HTML file H0DMain.html displays two choices for removing the cached client (see Figure 5-1 on page 164):

- Remove Cached Client (Autodetect Java 1 or Java 2)
- Remove Cached Client (Java 1 only)

In the first option the phrase "Autodetect Java 1 or Java 2" may confuse some users. In fact Autodetect refers to detecting the type of browser currently running, not to detecting the type of cached client that is currently installed.

Therefore the first option will always remove Java 1 cached client data (if any is present) if the user is running a Java 1 browser, and will always remove Java 2 cached client data (if any is present) if the user is running a Java 2 enabled browser.

In contrast, the second option will remove Java 1 cached client data, if the user is running a Java 1 browser or is running Internet Explorer with the Java 2 plug-in.

The following table summarizes the effect of these options for each browser:

Table 5-12 Effect of selecting a remove option

Browser:	-- Remove cached client (Autodetect Java 1 or Java 2):	-- Remove cached client (Java 1 only):
Netscape 4.x	Removes Java 1 cached client data.	Removes Java 1 cached client data.
Internet Explorer without the Java 2 plug-in	Removes Java 1 cached client data.	Removes Java 1 cached client data.
Netscape 6.x	Removes Java 2 cached client data.	No effect
Internet Explorer with the Java 2 plug-in	Removes Java 2 cached client data.	Removes Java 1 cached client data.

Scenario that may cause confusion

This subsection discusses a potentially confusing scenario that occurs when the user is running Internet Explorer and has the Java 2 plug-in installed, but uses an HTML file that has a client Java type of Java 1 and that launches a cached client.

- ▶ Because the client Java type is Java 1, the HTML file will launch Internet Explorer's default JVM, which is normally the Microsoft Java 1 JVM. See "Microsoft web browsers: Internet Explorer" on page 223.
- ▶ Therefore the Java 1 cached client will be installed, not the Java 2 cached client. The user may or may not realize this.
- ▶ Now suppose that the user wants to remove the cached client and chooses the first remove option, that is, Autodetect Java 1 or Java 2.
- ▶ Because the Java 2 plug-in is installed, the remove option detects the browser as a Java 2 enabled browser.
- ▶ Therefore the remove option tries to remove Java 2 cached client data, rather than the Java 1 cached client data that is actually installed.
- ▶ The Java 1 cached client data is not removed.

To avoid this problem, the user should be warned about this scenario if the HTML file's client Java type is Java 1, and the user is running Internet Explorer with the Java 2 plug-in in installed.

Removing the cached client

If the Java 2 plug-in is installed and the user chooses the first remove option, the following message is displayed:



Figure 5-24 Message for removal of Java 2 cached client

In other words, Host On-Demand will remove the HOD_CCR2.ccr file and do other cleanup, but the user must manually remove the Java 2 JAR files in the sticky cache.

The method of clearing the sticky cache may vary somewhat between Java 2 plug-ins. But for the IBM Java 2 plug-in version 1.3.1 runtime for Win32 the user must:

- ▶ Open the Java Plug-in Control panel. See section "The Java Plug-in Control Panel" on page 230.

- ▶ Select the Cache tab.
- ▶ Click Clear JAR cache.

The figure below shows the Cache tab of the Java Plug-in Control panel for the IBM Java Plug-in version 1.3.1.

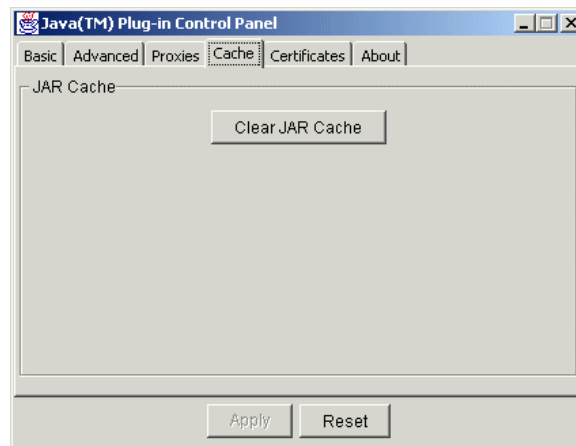


Figure 5-25 The cache tab with button for clearing the Java 2 plug-in's sticky cache

Note that this operation clears all the modules in the sticky cache, including modules that have been downloaded for other applications than Host On-Demand, and modules from other servers than Host On-Demand servers.

However, some Java 2 Plug-ins, such as the Sun Java plug-in version 1.4, have an additional button on the control panel that allows the users to selectively remove JAR files.

5.14 The Java 2 download client

Most of the material for the Java 2 download client has already been covered. Recall that:

- ▶ The Java 2 download client can be run without removing the Java 2 cached client. See 5.8.3, “Java 2 support with Host On-Demand 7.0” on page 189.
- ▶ The startup behavior of the Java 1 and Java 2 download clients is governed by the client Java type. See 5.11.2, “Startup behavior for Java 1 download client” on page 203 and 5.11.3, “Startup behavior for Java 2 download client” on page 204.

- ▶ The components for Java 2 download clients are downloaded as Java 2 JAR files and are stored in the Java plug-in's temporary cache. See "How Host On-Demand component modules are stored" on page 208.
- ▶ The following limitations exist in Host On-Demand support for the Java 2 download client (see "Limitations and workarounds" on page 194). For a discussion of the reasons for these limitations see "Reasons for three limitations on the Java 2 download client" on page 239.
 - No component can be downloaded that is not on the preload list.
 - The Function On-Demand client (HODThin.html) will not run correctly.
 - The default download clients (HOD_en.html, and so on) do not include some components.

5.15 Web browsers: Java 1 and Java 2 enabled

When a Host On-Demand HTML file with a client type of Java 2 or Auto Detect is run, it launches a detection applet in order to detect whether a Java 2 plugin-in is installed. A JVM must be present in order for this detection applet to be run.

Therefore you must ensure that a Java JVM as well as a supported web browser is installed on your Host On-Demand client machines.

Java 1 browsers, such as Netscape 4.x and Internet Explorer without the Java 2 plug-in, usually have a Java 1 JVM module included with them. Netscape 4.x includes a Symantec Java 1 JVM. Internet Explorer includes a Microsoft Java 1 JVM. However, be aware that in an early version of Windows XP the Java 1 JVM was not included with Internet Explorer. You had to download and install the JVM separately.

Java 2 enabled browsers, such as Netscape 6.x, need the Java 2 plug-in in order to launch a Java 2 applet. With Netscape 6.x you can install the Sun plug-in as part of the Netscape install, or you can install the IBM, Sun, or other vendor's Java 2 plug-in separately.

Internet Explorer can function as a Java 2 enabled browser or as a Java 1 browser when the Java 2 plug-in is installed. Either the IBM or the Sun Java 2 plug-in can be used.

Note: IBM recommends the IBM Java 2 plug-in for use with Host On-Demand.

Make sure that your client machines have a supported browser with a compatible JVM installed before rolling the machines out to your users.

For more on the Java 2 plug-in see 5.16, "The Java 2 plug-in" on page 227

5.15.1 Web browsers supported

The following list of browsers is supported. This list is taken from the online document *Planning, Installing, and Configuring Host On-Demand*. Check the latest documentation or the Host On-Demand web site for the most current list.

1. Netscape Navigator 4.6, 4.7, 6.1, 6.2
2. Microsoft Internet Explorer 4.01 with SP1, 5.0, 5.1, 5.5, or 6.0
3. Netscape Navigator 4.61 for OS/2
4. IBM Web Browser for OS/2 V1.2 (supports Java 2)

Also, the versions of Mozilla that correspond to the above versions of Netscape are supported.

Note: At the time this redbook was written, Netscape Version 7 was not supported by Host On-Demand 7.

5.15.2 Netscape web browsers

Netscape 4.x and Netscape 6.x

The table below summarizes the information in this subsection.

Table 5-13 Host On-Demand's use of Netscape Browsers

Browser:	JVM:	Type of applets that can be run:	Type of applets run by Host On-Demand:	HTML command to launch applet:
Netscape 4.x	Sun Java 1	Java 1	Java 1	APPLET
Netscape 6.x	Java 2 plug-in	Java 1 or Java 2	Java 1 or Java 2	APPLET

Netscape 4.x, which runs on several platforms, is a Java 1 browser. It executes HTML statements and uses an integrated Symantec Java 1 JVM to run applets. The Java 1 JVM can run Java 1 applets only. The HTML command APPLET is used to launch an applet.

Netscape 6.x, which likewise runs on several platforms, is a Java 2 enabled browser. It can execute HTML statements but needs a separately installed Java 2 plug-in to run applets.

The Java 2 plug-in can run Java 1 applets as well as Java 2 applets. The HTML command APPLET is used to launch an applet on Netscape 6.x, whether the applet is a Java 1 applet or a Java 2 applet.

With Netscape 6.x and the required Java 2 plug-in, the Host On-Demand HTML file usually launches the Java 2 version of the Host On-Demand applets (the HostOnDemand applet for the download client, or the CachedAppletSupport applet for the cached client). But in one situation the HTML file launches the Java 1 version of the Host On-Demand applets on the Java 2 plug-in. See “Startup behavior for Java 1 download client” on page 203.

For more information on how the Host On-Demand applets are launched see 5.17.4, “More information on launching the Host On-Demand applets” on page 241.

Installing Netscape 6.x

For information on installing Netscape 6.x on the Win32 platform see “Installing Netscape 6.x on the Win32 platform” on page 234.

5.15.3 Microsoft web browsers: Internet Explorer

Internet Explorer supports Java 1 and the Java 2 plug-in

The table below summarizes the information in this subsection.

Table 5-14 Host On-Demand's use of Internet Explorer

Browser:	JVM:	Type of applets that can be run:	Type of applets run by Host On-Demand:	HTML command to launch applet:	HTML command used by Host On-Demand:
Internet Explorer without the Java 2 plug-in	Microsoft Java 1	Java 1	Java 1	APPLET	APPLET
Internet Explorer with the Java 2 plug-in	Java 2 (Sun or IBM)	Java 1 or Java 2	Java 2	OBJECT (all Java 2 plug-ins) APPLET (newer Sun Java 2 plug-ins)	OBJECT

Unlike Netscape, which comes in either a Java 1 (Netscape 4.x) or a Java 2 enabled (Netscape 6.x) version, Internet Explorer is a Java 1 browser which can also use a Java 2 plug-in if one is installed on the system.

Internet Explorer has a setting called the default JVM that is normally set to be the Microsoft Java 1 JVM. This is the JVM that is included with Internet Explorer. However, the default JVM could be set to be any installed JVM.

Note: Host On-Demand always assumes that Internet Explorer's default JVM is set to the Microsoft Java 1 JVM.

The HTML command APPLET launches an applet on Internet Explorer's default JVM. The HTML command OBJECT launches an applet on the Java 2 plug-in. Also, with newer Sun Java 2 plug-ins, the APPLET command will launch an applet on the Java 2 plug-in.

The Host On-Demand HTML files always use the APPLET command to launch the Java 1 version of the Host On-Demand applets (the HostOnDemand applet and the CachedClientSupporter applet) and the OBJECT command to launch the Java 2 versions of the Host On-Demand applets. Note that Host On-Demand could use the OBJECT command to launch a Java 1 applet on the Java 2 Plug-in's Java 2 JVM, but it does not.

Note: Netscape 6.x also has a default JVM setting. The default JVM is normally set to the Java 2 plug-in when the Java 2 plug-in is installed.

For more information on how the Host On-Demand applets are launched see 5.17.4, "More information on launching the Host On-Demand applets" on page 241.

Default JVM for Internet Explorer must be MS Java 1 JVM

Some Java 2 plug-ins offer a checkbox that allows the user to switch the browser's default JVM to the Java 2 plug-in.

For Netscape 6.x, you should check this checkbox. However, IBM recommends that you do NOT set this checkbox for Internet Explorer.

Note: IBM recommends that for Internet Explorer running with the Java 2 plug-in, you do NOT set Internet Explorer's default JVM to the Java 2 plug-in.

The reason is that, as described in the previous section, the Host On-Demand HTML files always assume that the Internet Explorer's default JVM is set to Microsoft's Java 1 JVM. If you set the default JVM to the Java 2 plug-in, Host On-Demand may not run properly. However, there may be situations in which Host On-Demand will run properly with the default JVM set to the Java 2 plug-in.

The figure below shows a panel that appears during the installation of the Sun 1.4.1 Java 2 plug-in for the Win32 platform if the Custom install option is selected. IBM recommends that you set the checkboxes as shown in the figure below. That is:

- ▶ Do not check the checkbox for Internet Explorer.

The default JVM for Internet Explorer will be the Microsoft Java 1 JVM shipped with Internet Explorer.

- ▶ Do check the checkbox for Netscape 6.x

The default JVM for Netscape 6.x will be the Java 2 plug-in.

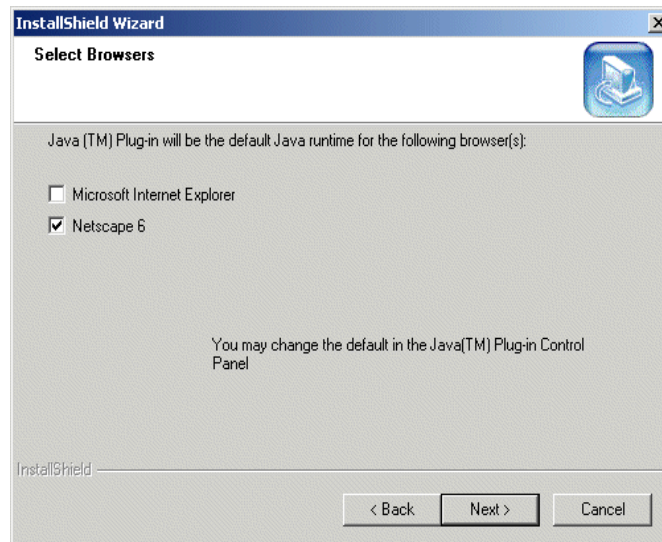


Figure 5-26 Install panel for Sun Java 2 plug-in 1.4.1

If you do not change the settings as shown above during the install, then IBM recommends that you change the settings after the install using the Java Plug-in Control panel.

The figure below shows the Browser tab in the Java Plug-in Control Panel of the Sun 1.4.1 Java 2 plug-in for the Win32 platform. IBM recommends that you set the checkboxes as shown in the figure below. That is:

- ▶ Do not check the checkbox for Internet Explorer.

The default JVM for Internet Explorer will be the Microsoft Java 1 JVM shipped with Internet Explorer.

- ▶ Do check the checkbox for Netscape 6.x.

The default JVM for Netscape 6.x will be the Java 2 plug-in.

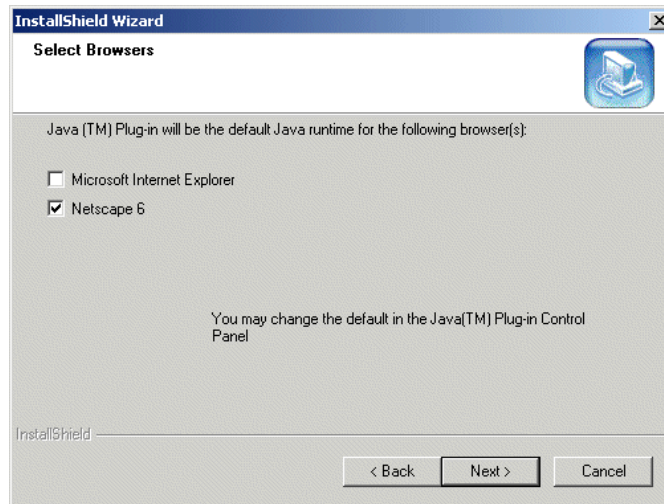


Figure 5-27 Browser tab of Java Plug-in Control Panel, Sun 1.4 for Win32

For more information about the Java Plug-in Control Panel see “The Java Plug-in Control Panel” on page 230.

Plug-ins that do not switch the JVM

The IBM Java 2 Plug-in version 1.3.1 runtime does not have this option to switch the JVM. Therefore you do not have to change the settings.

The install image for the runtime of this plug-in is distributed with Host On-Demand and is downloadable by Win32 Host On-Demand clients no matter what platform the Host On-Demand server is running on.

After the plug-in is installed, the Internet Explorer's default JVM remains set to the Microsoft Java 1 JVM, and Netscape 6.x's default JVM is set to the Java 2 plug-in. For more information about this plug-in see “Clients can download Java 2 runtime for Win32 platform” on page 228.

However, there is a message box that appears during the installation of this IBM Java 2 plug-in that might lead you to think that it is offering an option to set the Java 2 plug-in as the default JVM for Internet Explorer. Here is the message box.

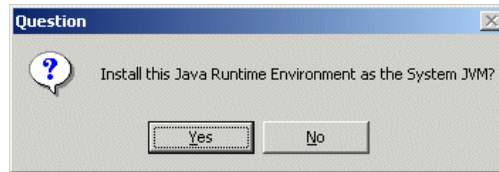


Figure 5-28 Message box during installation of IBM Java 2 plug-in runtime

In fact, the effect of clicking **Yes** on this panel is to copy the Java 2 versions of `java.exe` and `javaw.exe` to `\winnt\system32`. Your users may select either **Yes** or **No**. This will not affect the setting of Internet Explorer's default JVM.

5.16 The Java 2 plug-in

IBM recommends that you use an IBM Java 2 plug-in to run Host On-Demand, if an IBM Java 2 plug-in is available for the platform.

5.16.1 Java 2 plug-ins supported

Java 2 plug-ins supported for Win32

For Win32, Host On-Demand currently supports the following Java 2 Plug-ins:

Table 5-15 Java 2 plug-ins supported

Vendor	Version
IBM	1.3, 1.3.1
Sun	1.4

Java 2 plug-ins supported for non-Win32 platforms

For the Java 2 plug-ins supported for non-Win32 platforms, see the online guide *Planning, Installing, and Configuring Host On-Demand*.

New Java 2 plug-ins

When Host On-Demand has been tested with later versions of the Java 2 plug-in, the news will be posted on the Host On-Demand web site.

5.16.2 Clients can download Java 2 runtime for Win32 platform

Host On-Demand, for all platforms, includes an install image of the IBM Java 2 plug-in version 1.3.1 runtime for the Win32 platform. Therefore any client running on a Win32 platform can attach to a Host On-Demand server, no matter what platform the server is running on, download this install image, and install the Java 2 runtime for Win32 clients.

To download this installable version of the IBM Java 2 plug-in the user should connect to `HODMain.html` and click the following option:

IBM Windows Plug-in, Java 2 Technology Edition

If the client is running on a non-Win32 platform and this option is selected, the following message is displayed:

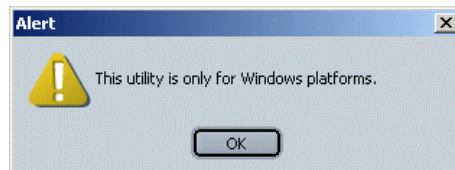


Figure 5-29 Attempt on non-Win32 platform to download Java 2 runtime for Win32

Do not use this plug-in for Win95 clients. Instead, download the Win95-compatible plug-in from the WebSphere Host On-Demand web page,

<http://www-3.ibm.com/software/webservers/hostondemand/support.html>

Installing the plug-in on the Win32 platform

See “Plug-ins that do not switch the JVM” on page 226 for a discussion of one of the install panels for the IBM Java 2 plug-in version 1.3.1 runtime.

A restricted user will not be allowed to install the plug-in.

Java plug-in for non-Win32 clients

For information on getting a Java 2 plug-in for a non-Win32 client see the online document *Planning, Installation, and Configuration Guide for Host On-Demand*.

Warning against having multiple Java 2 plug-ins installed

In our experience having multiple Java 2 plug-ins causes problems. If you have multiple Java 2 plug-ins installed, then:

- ▶ Uninstall all the Java 2 plug-ins
- ▶ Install the Java 2 plug-in that you want to use.

The Java 2 Java Console

The figure below shows the Java 2 Java Console.

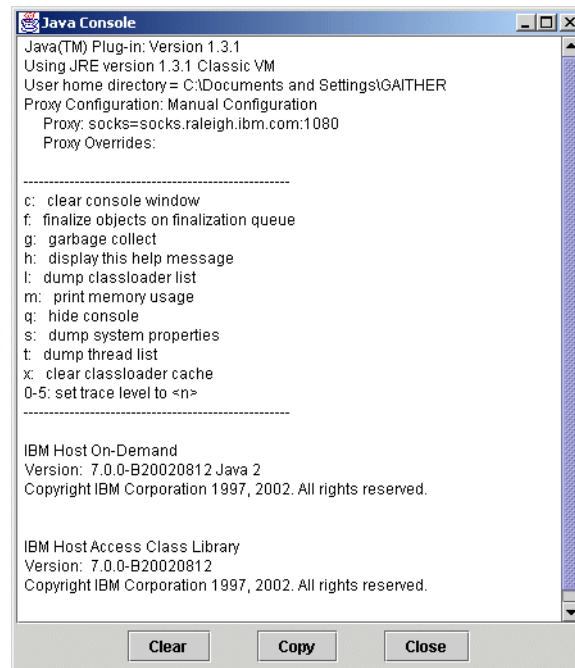


Figure 5-30 Java 2 console

Do not allow the console to appear immediately

The Java Console can be set to appear either as soon as the JVM is started, or at a later time. With Host On-Demand, you should set the Java Console to appear later. Otherwise problems can occur with Host On-Demand's Java 2 detection.

Having the Java Console appear later will not cause you to lose any debug output or error messages. After you make the Java Console appear, scroll the window back to view any debug output or error messages that have occurred.

To stop the Java Console from appearing immediately set the appropriate option on the Basic tab of the Java Plug-in Control Panel. See "The Java Plug-in Control Panel" on page 230.

To make the Java Console appear later:

- ▶ On Win32 platforms, wait until the Host On-Demand desktop panel appears, then click the Java 2 icon in the plug-in section of the Windows taskbar. The

icon will be a small version of either the Java coffee cup icon or the Duke (gnome) icon.

- On non-Win32 platforms, wait until the Host On-Demand desktop panel appears, then look for the icon in the system tray. If you do not find it, consult the Java 2 Plug-in documentation.

The figure below shows enlarged images of the Java coffee cup icon and the Duke (gnome) icon for popping up the Java console on the Win32 platform.



Figure 5-31 Icons for popping up Java 2 Java console on the Win32 platform

Using the console to determine vendor and version

To determine the vendor and version of the Java 2 Plug-in being run, pop up the Java 2 console and type the letter “s”. This causes the system information to be dumped to the console.

Scroll back through the system information and find the following 3 entries:
java.vendor, java.version, and java.vm.info.

Here is the information for these entries when the Java 2 Plug-in is the version 1.3.1 runtime for the Win32 platform. The install image for this runtime is distributed with Host On-Demand and is downloadable by Win32 Host On-Demand clients no matter what platform the Host On-Demand server is running on.

```
java.vendor=IBM Corporation
java.version=1.3.1
java.vm.info=J2RE 1.3.1 IBM Windows 32 build cn131-20020710 (JIT
enabled: jitc)
```

The Java Plug-in Control Panel

The Java 2 plug-in includes a Java Plug-in Control Panel for configuring the plug-in.

Starting the control panel

The following table shows how to start the control panel on the Win32 and Linux platforms. For other platforms consult the documentation distributed with the plug-in.

Table 5-16 How to start the Java 2 Plug-in Control Panel

Platform:	Plug-in Access
Win32	<div><div>- IBM Java 2 1.3.1 plug-in</div><div>Click Start, Programs, Java Plug-in Control Panel, or</div><div>Click the icon on the desktop</div><div> </div><div>- Sun Java 2 1.3.1 plug-in</div><div>Click Start, Settings, Control Panel, Java Plug-in, or</div><div>Click the icon on the desktop</div></div>
Linux	<div><div>- IBM Java 2 1.3.1</div><div><install>/jre/bin/JavaPluginControlPanel</div><div> </div><div>- Sun 1.4.0</div><div><install>/jre/bin/ControlPanel</div></div>

For Win32 the icon will be either a Java coffee cup icon or a Duke (gnome) icon. The following figure shows an enlarged image of the Java coffee cup icon on the Win32 desktop for launching the Java 2 Plug-in Control Panel.



Figure 5-32 Icon for launching the Java 2 Plug-in Control Panel from Win32 desktop

Control panel settings, IBM plug-in 1.3.1 for Win32 platform

The figure below shows the Java Plug-in Control Panel for the IBM Java 2 Plug-in version 1.3.1. The install image for the runtime of this version of the plug-in is distributed with Host On-Demand and is downloadable by Win32 Host On-Demand clients no matter what platform the Host On-Demand server is running on.

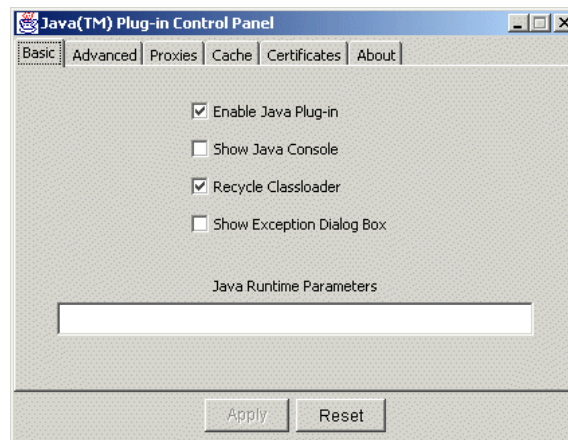


Figure 5-33 Java Plug-in Control Panel for IBM Plug-in 1.3.1 for Win32

The recommended settings for each tab are as follows:

► Basic tab

- Enable Java Plug-in.
Check this checkbox.
- Show Java console
Causes the Java console to be displayed as soon as the JVM is started. Do not check this checkbox. See “Do not allow the console to appear immediately” on page 229.
- Recycle Class Loader
Check this checkbox.
- Show Exception Dialog Box
Causes a popup window to appear if an exception occurs. Do not check this checkbox.
- Java runtime parameters
Additional runtime parameters to be set when the JVM is started. Leave blank.

► Advanced tab

- Java Runtime Environment
Enables the user to switch between several installed Java 2 Plug-ins. Leave this set to the first setting, Use Java Plug-in Default
- Enable Just In Time Compiler

Check this checkbox.

► Proxies tab

– Use browser settings

When this checkbox is checked, causes the Java 2 plug-in to read and use the browser's proxy settings. When not checked, causes the Java 2 plug-in to use the proxy settings set on this panel.

Normally this checkbox should be checked; that is, the plug-in should use the browser's proxy settings.

However, sometimes the plug-in cannot read the browser's proxy settings. This may cause a situation in which the Host On-Demand session cannot connect to the Host On-Demand server, even though the browser can connect to the Host On-Demand server. In this situation, uncheck this checkbox and try filling in the browser's proxy settings on this panel.

– Proxy settings

See the preceding item.

► Cache tab

This button clears the sticky cache. See Figure 5-25 on page 220.

► Certificates tab

This tab displays the certificates which the user has granted "always" while running a web browser, and allows the user to revoke the acceptance by deleting the certificate.

Settings for Sun Java 2 plug-in 1.3.1 for Win32 platform

The Java Plug-in Control Panel for the Sun Java 2 Plug-in Version 1.3.1 is similar to the control panel for the IBM Java 2 Plug-in Version 1.3.1. See the preceding section.

Additional settings for Sun Java 2 plugin 1.4 for Win32 platform

This section describes the settings for which the Java Plug-in Control Panel for Sun Java 2 Plug-in 1.4 is different from the IBM Java 2 Plug-in 1.3

► Basic tab

– Show console, Hide console, or Do not start console.

If set to Show or Hide, causes problems for Host On-Demand browser detection on Win32 platforms. See "Do not allow the console to appear immediately" on page 229. Set to Do not start console.

– Show Java in System Tray

Controls whether an icon is displayed for the Java console in the plug-in area of the Windows task bar when the JVM is started.

Check this checkbox.

► Browser tab

Controls whether the Java 2 plug-in functions as the default JVM for Internet Explorer and Netscape 6.

For the checkbox for Internet Explorer, IBM recommends that you do not check the checkbox. See “Default JVM for Internet Explorer must be MS Java 1 JVM” on page 224.

For the checkbox for Netscape 6, do check the checkbox. See “Default JVM for Internet Explorer must be MS Java 1 JVM” on page 224.

► Cache tab

The View button lets you view the names of the modules in the cache and selectively delete modules.

5.17 Additional information

Installing Netscape 6.x on the Win32 platform

First, see the warning in 5.16, “The Java 2 plug-in” on page 227 against having multiple Java 2 plug-ins installed.

Second, decide whether on the Win32 platform you want to run Netscape 6.X with the Sun Java 2 Plug-in or with the IBM Java 2 Plug-in.

Note: IBM recommends that you use the IBM Java 2 Plug-in with Host On-Demand.

If you want to run Netscape 6.x with the Sun plug-in, install the Sun plug-in as part of the Netscape 6.x install. In the directions below, choose Full or Custom. If you want to run Netscape 6.x with the IBM plug-in, do not install the Sun plug-in as part of the Netscape 6.x install. In the directions below, choose Recommended or Custom.

The Netscape 6.x install program has 3 main options:

► Recommended

This option does NOT install the Sun Java 2 plug-in.

► Full

This option installs the Sun Java 2 plug-in as part of the Netscape 6.x install.

► Custom

On the Additional Options panel, check the Sun Java 2 checkbox if you want to install the Sun Java 2 plug-in as part of the Netscape 6.x install. Do not check the Sun Java 2 checkbox if you plan to install the IBM Java 2 plug-in.

Sometimes Netscape 6.x on the Win32 platform has trouble finding the IBM Java 2 plug-in if the plug-in is installed first. If Netscape 6.x cannot find the Java 2 plug-in, follow these steps:

- ▶ Uninstall any Java 2 plug-ins that are installed.
- ▶ Install Netscape 6.x without the Sun Java Plug-in.
- ▶ Install the IBM Java 2 plugin.

Unexpected result with Internet Explorer

The fact that Internet Explorer has access both to its own internal Java 1 JVM and to a Java 2 JVM through the Java 2 plug-in can occasionally lead to an unexpected result.

For example, consider the following scenario on the client:

- ▶ The browser is Internet Explorer
- ▶ The client is a cached client.
- ▶ The client Java type in the HTML file is Java 1
- ▶ The Java 2 plug-in is installed

Because the Java 2 plug-in is installed, you might expect that Host On-Demand would refuse to run the Java 1 cached client. That is exactly what happens if you try this scenario on Netscape 6.x with the Java 2 plug-in. See 5.11.4, “Startup behavior for Java 1 cached client” on page 205.

However, in this case, because Internet Explorer has access to its default JVM, and because Host On-Demand assumes that the default JVM is set to the Microsoft Java 1 JVM, therefore Host On-Demand goes ahead and launches the Java 1 Host On-Demand cached client on the default JVM. See 5.11.4, “Startup behavior for Java 1 cached client” on page 205.

Minimum Microsoft JVM level for Internet Explorer

For Internet Explorer, Host On-Demand requires a minimum Microsoft JVM level of 3165. This is a fairly old level. Customers can raise this required level using the following session parameter:

```
<PARAMETER NAME=JVM_Minimum VALUE=xxxx>
```

An easy way to add this parameter is to use the Deployment Wizard, Additional Options page, Advanced Options panel, Additional Parameters tab.

5.17.1 More information on the Java 2 sticky cache

Java 2 has three options for caching the files needed by an applet:

- ▶ No caching, that is, download the files each time the applet is run.
- ▶ Browser caching. This is the default.
- ▶ Plug-in caching, that is, use the Java 2 plug-in's sticky cache.

Host On-Demand's Java 2 cached client uses the sticky cache. Host On-Demand's Java 2 download client uses the default setting, browser caching.

The cache is called sticky because files stored in it are not pushed out when the cache becomes full. Files are erased only when the user clears them. On the Win32 platform the sticky cache is located on the hard drive in a directory such as the following:

```
c:\Documents and Settings\JASmith\java_plugin_AppletStore\1.3.1\jar
```

where JASmith is the user name and 1.3.1 is the version of the Java 2 plug-in.

The sticky cache stores each file based both on the name of the file and on the identity of the server from which the file was downloaded. Consequently a separate set of components is stored in the cache for each Host On-Demand server visited, even if the components are identical. If a user revisits a Host On-Demand server, the previously cached components are used.

For example, the figure below shows part of the contents of a Java sticky cache after the user has connected to 2 different servers having the same version of Host On-Demand and has run the same cached client file on each server

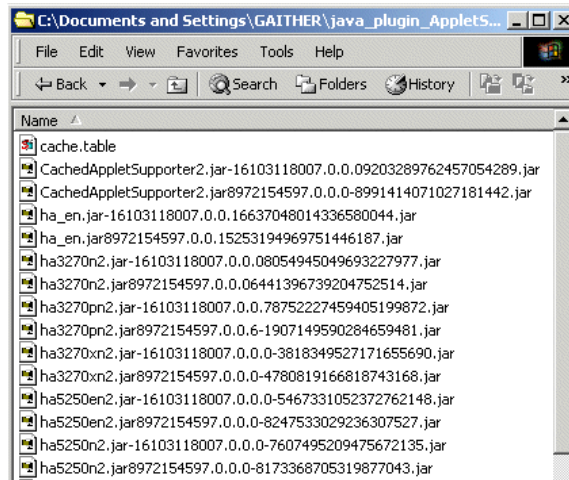


Figure 5-34 Example contents of Java 2 plug-in's sticky cache

Note that these modules are Java 2 Host On-Demand JAR files, because the JAR file names end in '2', such as ha3270n2.jar. (The JAR file ha_en.jar contains data, not code, and is used for both the Java 1 and the Java 2 version of Host On-Demand.) Note also that there appear to be two copies of every JAR file.

The example below shows more clearly that each similarly named JAR file refers to the same JAR file on a different server. This example displays some of the contents of a cache-management file called cache.table which is kept in the sticky cache.

Example 5-3 Partial contents of cache.table

```
#This File is Machine Generated.Please DO NOT CHANGE IT !! Changing it will BRE
#Mon Oct 14 17:28:29 EDT 2002
http://hodsrv1/hod/hodmacn2.jar=7.0.0.0\#hodmacn2.jar-16103118007.0.0.0-429069
http://hodsrv1/hod/hahostgn2.jar=7.0.0.0\#hahostgn2.jar-16103118007.0.0.0-1427
http://hodsrv1/hod/sccbase2.jar=2.0.2.7\#sccbase2.jar-16103118002.0.2.7-281836
http://hodsrv2/hod/hafntap.jar=7.0.0.0\#hafntap.jar8972154597.0.0.012806893857
http://hodsrv2/hod/hodimpn2.jar=7.0.0.0\#hodimpn2.jar8972154597.0.0.0-38733422
http://hodsrv2/hod/halumn2.jar=7.0.0.0\#halumn2.jar8972154597.0.0.022258443853
http://hodsrv2/hod/ha3270n2.jar=7.0.0.0\#ha3270n2.jar8972154597.0.0.0644139673
http://hodsrv2/hod/hac1taun2.jar=7.0.0.0\#hac1taun2.jar8972154597.0.0.0-676875
http://hodsrv2/hod/haftpn2.jar=7.0.0.5\#haftpn2.jar8972154597.0.0.5-2215966638
http://hodsrv1/hod/hassln2.jar=7.0.0.0\#hassln2.jar-16103118007.0.0.0-52905810
http://hodsrv2/hod/ha5250n2.jar=7.0.0.0\#ha5250n2.jar8972154597.0.0.0-81733687
http://hodsrv1/hod/hodappln2.jar=7.0.0.0\#hodappln2.jar-16103118007.0.0.032015
http://hodsrv1/hod/hacicsn2.jar=7.0.0.0\#hacicsn2.jar-16103118007.0.0.0-270400
```

You can see in the example above that each JAR file name is mapped to a server name and path, such as //HOD1/hod/hodmacn2.jar, and to a file version, such as 7.0.0.0.

5.17.2 More information on the cached client

Scenario requiring Java 1 cached client to be removed

Here is an example of an upgrade scenario that requires that the Java 1 cached client be removed.

- ▶ The user attaches to server HODSRV1, which is running a particular level of code, say HOD7 CSD1. The user runs an HTML file which installs a Java 1 cached client with a preload list. At this point the components on the preload list are installed on the user's machine.
- ▶ The user attaches to a second server HODSRV2 which is running a lower level of code, say HOD7 GM. Even though the user is attached to a different server, no addition components are needed, so far. The Java 1 cached client on the user's machine uses the components already installed from HODSRV1.
- ▶ Now the user tries to access a component that was not installed from server HODSRV1, because it was not on the preload list. In this scenario Host On-Demand would display the following message:

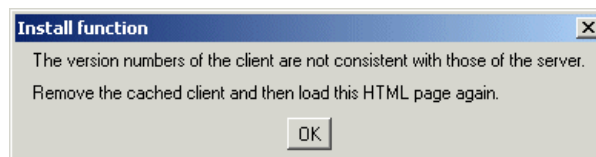


Figure 5-35 Version numbers not consistent

The problem is that the Java 1 components at the second server are older than the Java 1 components already downloaded by the cached client. Host On-Demand will not install an older Java 1 cached client over a newer one.

Therefore, in this scenario, the user has to remove the cached client, and then re-install the cached client at the second server.

5.17.3 More information on the download client

Reason for restriction on Java 1 download client

The reason that the Java 1 download client cannot be run while the Java 1 cached client is installed has to do with the order in which the Java 1 JVM for Internet Explorer looks for class files when the cached client is installed. Remember, in the following discussion, that we are talking about the Java 1 cached client and the Java 1 download client, not the Java 2 cached client and download client. Also, we are talking about Internet Explorer not Netscape 4.x.

When the Java 1 cached client is installed and the user is running Internet Explorer, Host On-Demand includes at the first of the system classpath the path of the HODCC directory where the user's cached client class files are installed. Consequently, when a Host On-Demand Java method is called, the JVM looks in the HODCC directory before looking in the JAR files downloaded by the browser in the browser cache.

Now suppose that the Java 1 download client is run without the Java 1 cached client being installed. This is the normal case. When the download client is run and a Java method is called, the JVM first looks in the HODCC directory for the class file. The HODCC directory does not exist, because the cached client is not installed. Therefore the JVM looks for, and finds, the class file in one of the Java 1 download client's JAR files in the browser cache.

Now suppose hypothetically that:

1. The Java 1 cached client is installed and Host On-Demand allows the Java 1 download client to be run. (In fact, Host On-Demand does not allow this.)
2. The cached client and the download client are different levels of Host On-Demand code, for example, version 7.0.0 and version 7.0.1.
3. A function called `MyMethod()` has changed between version 7.0.0 and 7.0.1

When the Java 1 download client is run and `MyMethod()` is called, the JVM would look for the class file first, and find it, in the HODCC directory. Therefore the JVM would execute the Java 1 cached client version of `MyMethod()` instead of the download client version. Because the wrong version is executed in this situation, an error could easily occur. The error could be minor or possibly catastrophic.

To avoid this scenario, the Java 1 version of Host On-Demand does not allow the download client to be run while the cached client is installed.

Reasons for three limitations on the Java 2 download client

The three minor limitations on the Java 2 download client that are described in 5.14, "The Java 2 download client" on page 220 are the result of two restrictions that affect the Java 2 download client:

1. JAR/CAB file restriction

For both the Java 1 and Java 2 download clients, all JAR or CAB files that are downloaded have to be specified in the APPLET command that launches the download client.

Unlike the cached client, the download client does not have a CachedAppletSupport applet to download JAR or CAB files after the client is launched.

The Java 1 download client gets around this restriction by downloading components not on the preload list as loose class files rather than as JAR or CAB files.

2. Signing restriction

The Java 2 JVM requires that all modules belonging to the same Java package must be signed in the same way.

This requirement means that the class files belonging to a Java package may be:

- A collection of loose class files (unsigned).
- A collection of class files in an unsigned JAR file.
- A collection of class files in one or more JAR files signed with the same certificate.

Likewise the class files belonging to a Java package may NOT be:

- A collection of JAR files signed with different certificates.
- A collection of signed JAR files and unsigned JAR files.
- A combination of a signed JAR file and loose class files (unsigned).

Now we can explain the limitations.

The reason for the first limitation described in 5.14, “The Java 2 download client” on page 220, that no component can be downloaded that is not on the preload list, is that after being launched the Java 2 download client is prevented from downloading additional components either as JAR or CAB files (because of the JAR/CAB file restriction described above) or as loose class files (because of the signing restriction described above). Therefore the Java 2 download client cannot download an additional component.

The reason for the second limitation, that the Function On-Demand client will not run correctly, is that the Function On-Demand client is a download client with a preload list. It contains a core of function that is intended to be augmented by components downloaded after the applet is launched. Therefore this limitation is really a particular instance of the first limitation above.

The third limitation, that the default download clients do not contain some components, is yet another particular instance of the first limitation above. In order to reduce the startup time of the default download clients, some components are omitted. Therefore the default download clients are download clients with preload lists, and fall under the first limitation.

5.17.4 More information on launching the Host On-Demand applets

This section provides more information on how the Host On-Demand applets (the HostOnDemand applet and the CachedAppletSupport applet) are launched.

Download client on Java 1 browser

The example below shows how a Host On-Demand HTML file launches the download client on a Java 1 browser. The command is APPLET for both Internet Explorer and Netscape 4.77, and the parameters are exactly the same for Internet Explorer and Netscape 4.77.

Example 5-4 Download client on Java 1 browser

```
<APPLET
  ARCHIVE="habasen.jar,hodbasen.jar,hodimg.jar,hacp.jar,hodsignn.jar,
    hamacrtn.jar,hacлтаun.jar,hodh11n.jar, havtn.jar,haslpn.jar,
    hakeypdn.jar,ha3270n.jar,hamacuin.jar,hodmacn.jar,haprintn.jar,
    halumn.jar,ha3270xn.jar,hodcfgn.jar,sccbase.jar,ha5250xn.jar,
    hodsslн.jar,ha3270pn.jar,hassln.jar,hacicsn.jar,haftpn.jar,
    ha5250pn.jar,hahostgn.jar,haxfern.jar,ha5250n.jar,hodapplн.jar,
    hakeympn.jar,hacolorn.jar,hodimpn.jar,ha5250en.jar"
  NAME="HODApplet"
  CODE="com.ibm.eNetwork.HOD.HostOnDemand"
  WIDTH="80%"
  HEIGHT="80%">
<PARAM NAME="Cabinets"          VALUE="habasen.cab,hodbasen.cab,hodimg.cab,
    hacp.cab,hodsignn.cab,hamacrtn.cab,
    hacлтаun.cab,hodh11n.cab,havtn.cab,
    haslpn.cab,hakeypdn.cab,ha3270n.cab,
    hamacuin.cab,hodmacn.cab,haprintn.cab,
    halumn.cab,ha3270xn.cab,hodcfgn.cab,
    sccbase.cab,ha5250xn.cab,hodsslн.cab,
    ha3270pn.cab,hassln.cab,hacicsn.cab,
    haftpn.cab,ha5250pn.cab,hahostgn.cab,
    haxfern.cab,ha5250n.cab,hodapplн.cab,
    hakeympn.cab,hacolorn.cab,hodimpn.cab,
    ha5250en.cab">
<PARAM NAME="ParameterFile"    VALUE="HODData\augj1d1\params.txt">
<PARAM NAME="JavaScriptAPI"    VALUE="false">
```

In the above example note that:

- ▶ In the first line the command is APPLET.
- ▶ In the ARCHIVE attribute JAR files are listed, while in the Cabinets parameter CAB files are listed. If the browser is Netscape then the JAR files are downloaded; if the browser is Internet Explorer then the CAB files are downloaded.
- ▶ In the ARCHIVE attribute and in the Cabinets parameter the modules are Java 1 modules. This is evident from the fact that the module names do not end with '.2'. Examples: habasen.jar, hodbasen.jar .
- ▶ In the CODE parameter the class to be invoked is com.ibm.eNetwork.HOD.HostOnDemand. This is the entry point for the HostOnDemand applet.

Cached client on Java 1 browser

The example below shows how a Host On-Demand HTML file launches the cached client on a Java 1 browser. The command is APPLET for both Internet Explorer and Netscape 4.77, and the parameters are exactly the same for Internet Explorer and Netscape 4.77.

Although the cached client is invoked in two different circumstances, to install the cached client and to start the installed cached client, the applet is launched in the same way in both circumstances. The applet itself determines whether the cached client needs to be installed.

Example 5-5 Cached client on Java 1 browser

```

<APPLET
  ARCHIVE="CachedAppletSupporter.jar"
  MAYSCRIPT
  NAME="CachedAppletSupporter"
  CODE="com.ibm.eNetwork.HOD.cached.appletsupport.CachedAppletSupportApplet"
  WIDTH="2"
  HEIGHT="2">
<PARAM NAME="Cabinets"                VALUE="CachedAppletSupporter.cab">
<PARAM NAME="DebugComponents"          VALUE="false">
<PARAM NAME="PreloadComponentList"     VALUE="HABASE;HODBASE;HODIMG;HACP;
                                           HAFNTIB;HAFNTAP;HAMACRT;
                                           HACTAU;HODHLL;HAVT;HASLP;
                                           HAKEYPD;HA3270;HAMACUI;
                                           HODMAC;HAPRINT;HALUM;HA3270X;
                                           HODCFG;SCCBASE;HA5250X;HODSSL;
                                           HA3270P;HASSL;HACICS;HAFTP;
                                           HA5250P;HAHOSTG;HAXFER;HA5250;
                                           HODAPPL;HAKEYMP;HACOLOR;
                                           HODIMP;HA5250E">
<PARAM NAME="DebugCachedClient"        VALUE="false">

```

```

<PARAM NAME="CachedClientSupportedApplet"
VALUE="com.ibm.eNetwork.HOD.HostOnDemand">
<PARAM NAME="InstallerFrameWidth"      VALUE="550">
<PARAM NAME="InstallerFrameHeight"     VALUE="250">
<PARAM NAME="UpgradePromptResponse"    VALUE="Prompt">
<PARAM NAME="UpgradePercent"           VALUE="100">
</APPLET>

```

In the above example note that:

- ▶ In the first line the command is APPLET.
- ▶ In the ARCHIVE attribute a JAR file is listed, while in the Cabinets parameter a CAB file is listed. If the browser is Netscape then the JAR file is downloaded; if the browser is Internet Explorer then the CAB file is downloaded.
- ▶ In the ARCHIVE attribute and in the Cabinets parameter the modules are Java 1 modules. This is evident from the fact that the module names do not end with '2'. Examples: CachedAppletSupporter.jar, CachedAppletSupporter.cab.
- ▶ In the CODE parameter the class to be invoked is com.ibm.eNetwork.HOD.cached.appletSupport.CachedAppletSupportApplet. This is the entry point for the CachedAppletSupport applet.
- ▶ The PreloadComponentList contains a list of the components that are to be downloaded initially. These are component names, not module names. A component may consist of one or more modules.
- ▶ The DebugCachedClient parameter is set to false.
- ▶ The CachedClientSupportedApplet parameter specifies the name of the applet to be launched if the cached client is installed. This is the HostOnDemand applet.

Download client on Java 2 browser

The example below shows how a Host On-Demand HTML file launches the Java 2 download client on Internet Explorer with the Java 2 plug-in. The command is OBJECT. On Netscape 6.x the command would be APPLET but the other information would be almost exactly the same.

Example 5-6 Download client on Java 2 browser

```

<OBJECT
  classid="clsid:8AD9C840-044E-11D1-B3E9-00805F499D93"
  WIDTH=80%
  HEIGHT=80%
  ID="HODApplet">
<PARAM NAME=CODE VALUE="com.ibm.eNetwork.HOD.HostOnDemand">

```

```

<PARAM NAME=ARCHIVE VALUE = "habasen2.jar,hodbasen2.jar,hodimg.jar,
                                hACP.jar,hamacrtn2.jar,hac1taun2.jar,
                                hodh11n2.jar,havtn2.jar,has1pn2.jar,
                                hakeypdn2.jar,ha3270n2.jar,hamacuin2.jar,
                                hodmacn2.jar,haprintn2.jar,halumn2.jar,
                                ha3270xn2.jar,hodcfgn2.jar,sccbase2.jar,
                                ha5250xn2.jar,hodss1n2.jar,ha3270pn2.jar,
                                hass1n2.jar,hacicsn2.jar,haftpn2.jar,
                                ha5250pn2.jar,hahostgn2.jar,haxfern2.jar,
                                ha5250n2.jar,hodappln2.jar,hakeympn2.jar,
                                hacolorn2.jar,hodimpn2.jar,ha5250en2.jar">
<PARAM NAME="type" VALUE="application/x-java-applet;version=1.3">
<PARAM NAME="MAYSCRIPT" VALUE="true">
<PARAM NAME="scriptable" VALUE="true">
<PARAM NAME=RealDocumentBase
                                VALUE=http://localhost/hod/augj2d1.html?JavaType=java2>
<PARAM NAME="ParameterFile" VALUE="HODData\augj2d1\params.txt">
<PARAM NAME="ShowDocument" VALUE="_parent">
<PARAM NAME="JavaScriptAPI" VALUE="false">
<PARAM NAME="PreloadComponentList" VALUE="HABASE;HODBASE;HODIMG;HACP;HAFNTIB;
                                HAFNTAP;HAMACRT;HAC1TAU;HODHLL;
                                HAVT;HASLP;HAKEYPD;HA3270;HAMACUI;
                                HODMAC;HAPRINT;HALUM;HA3270X;
                                HODCFG;SCCBASE;HA5250X;HODSSL;
                                HA3270P;HASSL;HACICS;HAFTP;HA5250P;
                                HAHOSTG;HAXFER;HA5250;HODAPPL;
                                HAKEYMP;HACOLOR;HODIMP;HA5250E">

</OBJECT>

```

In the above example note that:

- ▶ In the first line the command is OBJECT.
- ▶ In the ARCHIVE parameter JAR files are listed. There is no Cabinets parameter, because the Java 2 plug-in uses JAR files.
- ▶ In the ARCHIVE parameter the modules are Java 2 modules. This is evident from the fact that the module names end in 2. Examples: habasen2.jar, hodbasen2.jar.
- ▶ In the CODE parameter the class to be invoked is com.ibm.eNetwork.HOD.HostOnDemand. This is the entry point for the HostOnDemand applet.
- ▶ The PreloadComponentList contains a list of the components that are to be downloaded initially. These are component names, not module names. A component may consist of one or more modules.

Cached client on Java 2 browser

The example below shows how a Host On-Demand HTML file launches the Java 2 cached client on Internet Explorer with the Java 2 plug-in. The command is OBJECT. On Netscape 6.x the command would be APPLET but the other information would be almost exactly the same.

Example 5-7 Cached client on Java 2 browser

```
<OBJECT
  classid="clsid:8AD9C840-044E-11D1-B3E9-00805F499D93"
  WIDTH=80%
  HEIGHT=80%
  ID="HODApplet">
<PARAM NAME="CODE"
  VALUE="com.ibm.eNetwork.HOD.cached.appletloader.CachedAppletLoader.class">
<PARAM NAME="MAYSCRIPT"      VALUE="true">
<PARAM NAME="SCRIPTABLE"     VALUE="true">
<PARAM NAME="cache_option"    VALUE="Plugin">
<PARAM NAME="CachedClient"    VALUE="true">
<PARAM NAME="Java2"           VALUE="true">
<PARAM NAME=PreloadComponentList VALUE="HABASE;HODBASE;HODIMG;HACP;HAFNTIB;
  HAFNTAP;HAMACRT;HACLTau;HODHLL;
  HAVT;HASLP;HAKEYPD;HA3270;HAMACUI;
  HODMAC;HAPRINT;HALUM;HA3270X;HODCFG;
  SCCBASE;HA5250X;HODSSL;HA3270P;HASSL;
  HACICS;HAFTP;HA5250P;HAHOSTG;
  HAXFER;HA5250;HODAPPL;HAKEYMP;HACOLOR;
  HODIMP;HA5250E;HA_EN">
<PARAM NAME="cache_archive"  VALUE="CachedAppletSupporter2.jar,
  haprintn2.jar,ha5250n2.jar,hodimg.jar,
  ha3270xn2.jar,hac1taun2.jar,
  ha3270pn2.jar,hacolorn2.jar,
  hodh1ln2.jar,hacicsn2.jar,
  hahostgn2.jar,hamacuin2.jar,
  hakeympn2.jar,hafntib.jar,
  hodcfgn2.jar,havtn2.jar,hodappln2.jar,
  haxfern2.jar,ha_en.jar,halumn2.jar,
  hACP.jar,hodbasen2.jar,hafntap.jar,
  ha3270n2.jar,hassln2.jar,hamacrtn2.jar,
  hodssln2.jar,ha5250xn2.jar,haslpn2.jar,
  ha5250pn2.jar,ha5250en2.jar,
  sccbase2.jar,hakeypdn2.jar,haftp2.jar,
  hodmacn2.jar,habasen2.jar,
  hodimpn2.jar">
<PARAM NAME="cache_version"  VALUE="7.0.0.0,7.0.0.6,7.0.0.0,7.0.0.2,
  7.0.0.0,7.0.0.0,7.0.0.6,7.0.0.1,
  7.0.0.0,7.0.0.0,7.0.0.0,7.0.0.2,
  7.0.0.0,7.0.0.0,7.0.0.2,7.0.0.1,
  7.0.0.0,7.0.0.1,7.0.0.1,7.0.0.0,
```

```

7.0.0.0,7.0.0.3,7.0.0.0,7.0.0.0,
7.0.0.0,7.0.0.1,7.0.0.1,7.0.0.1,
7.0.0.0,7.0.0.0,7.0.0.0,2.0.2.7,
7.0.0.0,7.0.0.5,7.0.0.0,7.0.0.6,
7.0.0.0">
<PARAM NAME=RealDocumentBase
    VALUE="http://localhost/hod/auj2cc.html?JavaType=java2">
<PARAM NAME=AppName                VALUE="com.ibm.eNetwork.HOD.HostOnDemand">
<PARAM NAME="ShowDocument"         VALUE="_parent">
<PARAM NAME="CachedClient"         VALUE="true">
<PARAM NAME="DebugCachedClient"    VALUE="false">
<PARAM NAME="ParameterFile"        VALUE="HODData\auj2cc\params.txt">
<PARAM NAME="JavaScriptAPI"        VALUE="false">
</OBJECT>

```

In the above example note that:

- ▶ In the first line the command is OBJECT.
- ▶ In the CODE parameter the class to be invoked is
com.ibm.eNetwork.HOD.cached.appletLoader.CachedAppletLoader.class
This is the entry point for the Java 2 CachedAppletSupport applet.
- ▶ In the cache_option parameter the value is set to Plugin. This instructs the Java 2 plug-in to use the sticky cache.
- ▶ The PreloadComponentList parameter contains a list of the components that are to be downloaded initially. These are component names, not module names. A component may consist of one or more modules.
- ▶ The cache_archive parameter lists the JAR files that are to be placed in the sticky cache. Host On-Demand specifies that all the JAR files should be placed in the sticky cache
- ▶ The cache_version parameter is a Host On-Demand parameter that tells the Cached Applet Support applet the versions of the modules in the sticky cache.
- ▶ In the ARCHIVE parameter the modules are Java 2 modules. This is evident from the fact that the module names end in 2. Examples: habasen2.jar, hodbasen2.jar.
- ▶ The AppName parameter specifies the name of the applet to be launched if the cached client is installed. This is the HostOnDemand applet.
- ▶ The DebugCachedClient parameter is set to false..

6

Database On-Demand

In this chapter we discuss the administration and client side of Database On-Demand. Database On-Demand is a Java applet that allows users to perform Structured Query Language (SQL) requests to iSeries databases through a Java Database Connectivity (JDBC) driver. Though Database On-Demand is shipped with a JDBC driver, other user-installed JDBC drivers can also be registered and used, although Host On-Demand does not provide support for other drivers.

6.1 Administering Database On-Demand

The first important point to know is that an administrator cannot create SQL statements for users. He or she can, however, create groups and users, define the database functions that users can perform, manage statements that users have created, and create groups and users.

The Database On-Demand users are based on the users and groups that are defined in Host On-Demand, with the Database On-Demand being one attribute of this.

For example, let's say that two people in your organization (Greg and Kelly) will need to use Database On-Demand for SQL queries to an iSeries. Follow the directions below to configure these users for Database On-Demand.

6.1.1 Creating Database On-Demand groups and users

Use the Host On-Demand administration utility to create a group called DatabaseAdmin and users called Greg and Kelly. Make each user a member of the DatabaseAdmin group. The group name is not important, and can be anything that would best describe the group.

6.1.2 Configuring database options

In the Host On-Demand administration utility, right-click the Database group then click **Database > Options...** (Figure 6-1 on page 249) and the Database On-Demand Group Options window (Figure 6-2 on page 250) will be displayed. Some options allow or restrict certain functions, while other options set default values. Database On-Demand provides the following two methodologies for granting users authority for options:

- ▶ Most permissive

If a user belongs to two or more groups, the most permissive authority is granted for that option.

For example, if a user is a member of DatabaseAdmin and this group allows deleting SQL statements, and the user is also a member of DatabaseUsers, but DatabaseUsers does not allow deleting SQL statements, the user is allowed to delete SQL statements.

► User override

When options are modified for a selected user, the new settings override settings for the group or groups to which the user may belong. Default user options (those not explicitly set by a user or administrator) do not override group settings. This gives the administrator the ability to allow or restrict certain functions at the user level.

For example, suppose a user belongs to DatabaseAdmin, which allows edit SQL statements, and the user level options do not allow edit SQL statements. The user cannot edit SQL statements, even though the DatabaseAdmin group allows it.

The Host On-Demand online help provides a table indicating how user authority for Database On-Demand is granted.

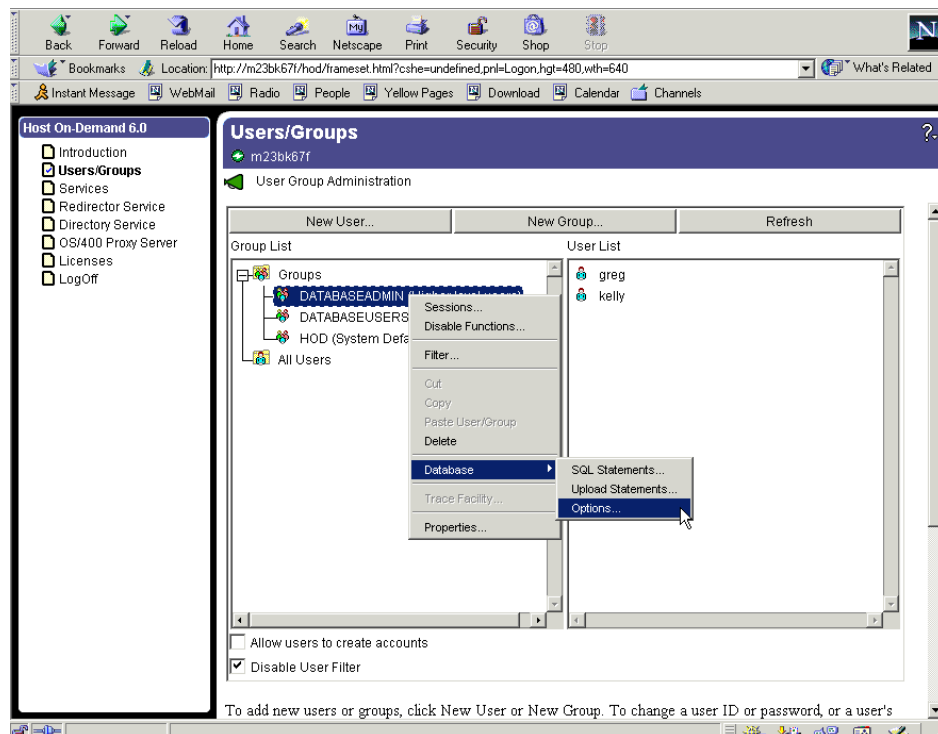


Figure 6-1 Opening the Database On-Demand administrator window

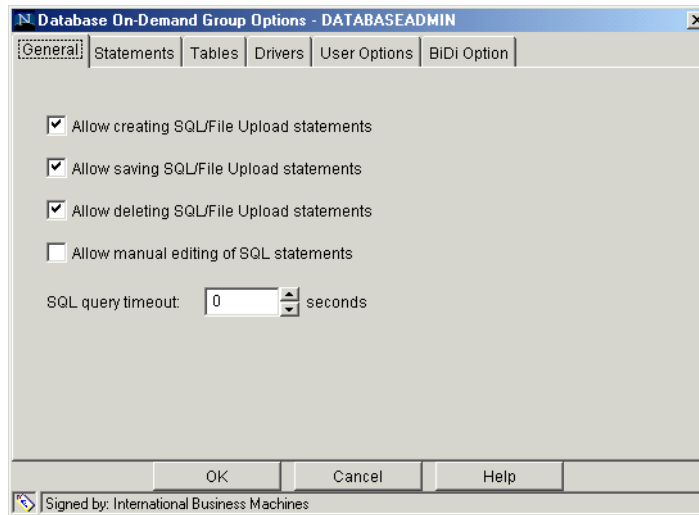


Figure 6-2 Database On-Demand Group Options window

The configurable options available from this window are self explanatory, and more information is available in the Host On-Demand online help if required.

6.1.3 Administering SQL statements

Database On-Demand allows the administrator to manage SQL statements that were previously saved by a group or user; the administrator can copy, rename or delete statements.

Let's say that Greg saved an SQL statement called List Employees and Kelly decides that access to the same database for the same information would be useful. The administrator must complete these steps:

1. Right-click the user with the saved SQL statement (in the example, Greg), and click **Database > SQL Statements....** See Figure 6-3.

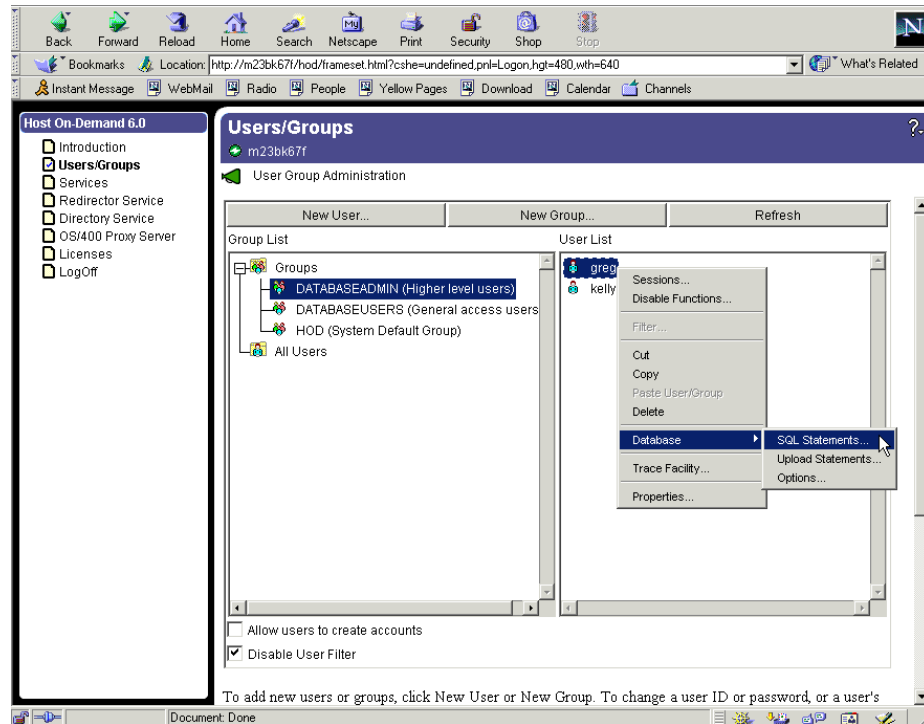


Figure 6-3 Preparing to copy a saved SQL statement from a user

2. When the Database On-Demand User Statements window is displayed as shown in Figure 6-4, select the saved SQL statement in the left-hand pane, in this example **List Employees**.
3. Select the user or group to which you wish to copy the SQL statement.

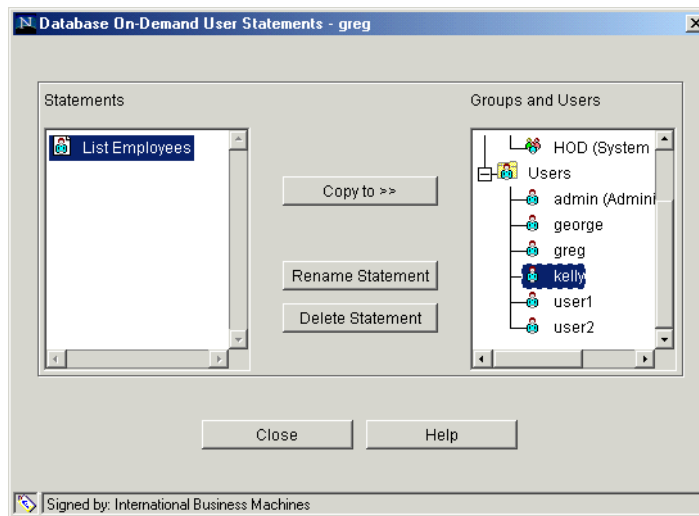


Figure 6-4 Copying saved SQL statements to a group

4. Click **Copy to >>** which will then show you any SQL statements already saved with that user, and gives you an option of renaming the SQL if desired. See Figure 6-5.

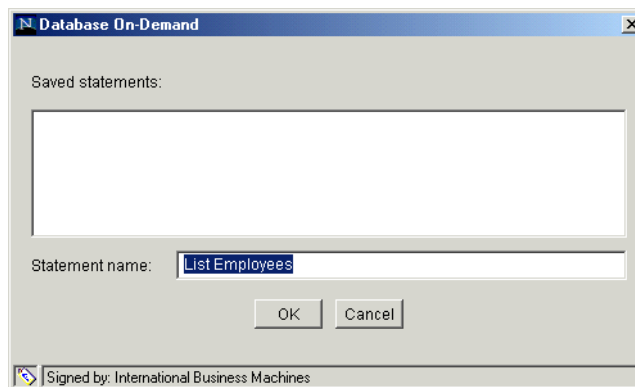


Figure 6-5 Saving the SQL statement to another user

5. Click **OK** to save the SQL to that user ID.

6.2 Using Database On-Demand

Start Database On-Demand from HODMain.html or directly with HODDatabase.html, and log in. On the Database On-Demand applet window, as shown in Figure 6-6, you will see two tabs:

- ▶ SQL Wizard tab

The SQL Wizard is the default view. It displays a view of previously saved SQL statements.

- ▶ File Upload tab

This tab displays a view of previously saved File Upload statements.

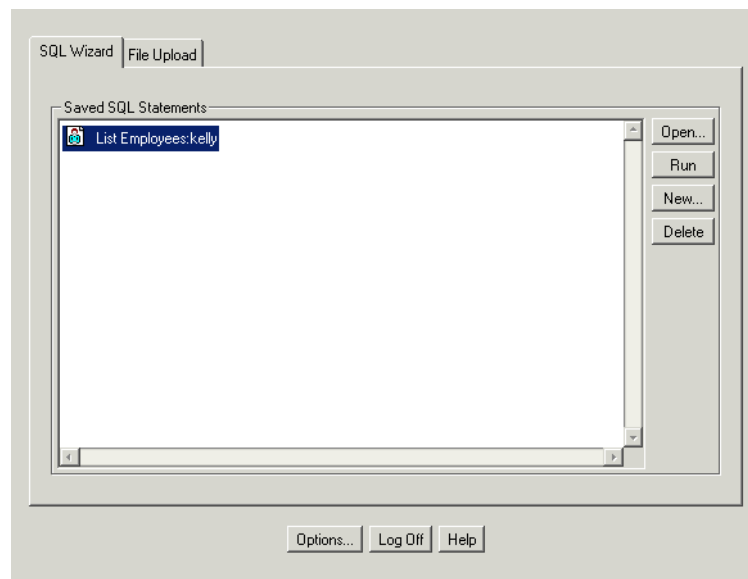


Figure 6-6 Database On-Demand applet

From this applet, you can do the following:

- ▶ Create a new SQL or File Upload statement
- ▶ Run an existing SQL or File Upload statement
- ▶ Open an existing SQL or File Upload statement
- ▶ Delete an existing SQL or File Upload statement

6.2.1 Creating a new SQL statement

You can create a new SQL query by performing the following steps. By way of example, we will create a query to list all the names of people at a certain zip code, using data stored in the Callup database hosted on an iSeries server.

Creating a new statement

From the Database On-Demand applet (Figure 6-6), perform these steps:

1. Click **New**

You will be presented with the Logon window shown in Figure 6-7 on page 255.

2. Type the database URL:

`jdbc:as400://iSeriesname`

To use SSL when connecting, type:

`jdbc:as400://iSeriesname;secure=true`

To use the OS/400 Proxy server without security:

`jdbc:as400://iSeriesname;proxy server=HODServername`

To use a secure connection via the OS/400 Proxy:

`jdbc:as400://iSeriesname;secure=true;proxy server=HODServername`

3. Type your user ID and password (if required).
4. Select the JDBC driver you want to use to access the database. In this case, the driver supplied, called AS/400 Toolbox for Java, is used.
5. Click **Connect**.

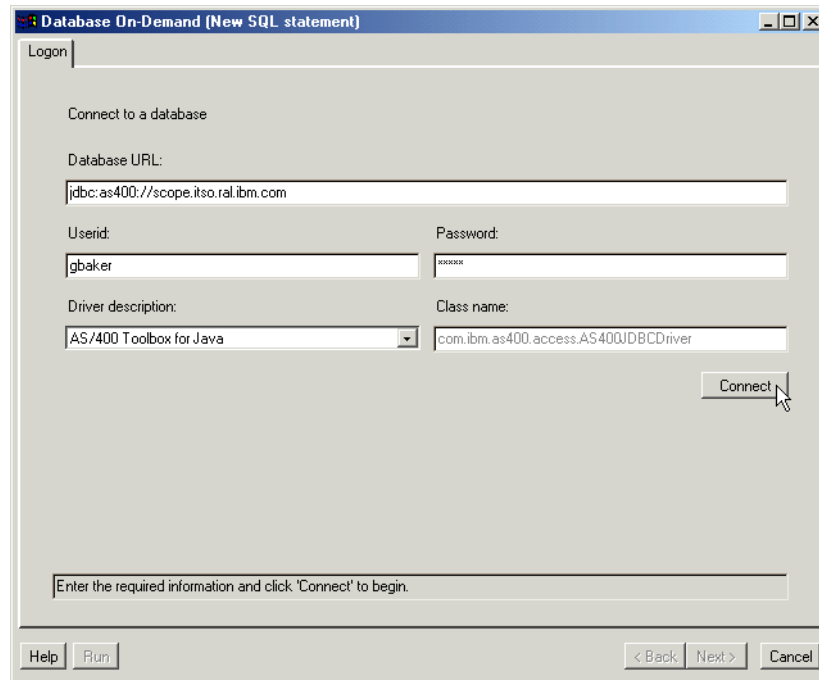


Figure 6-7 Database On-Demand Logon tab

When the connection to the iSeries has been made, the following tabs are added: Tables, Join, Condition 1, Columns, Sort, Output, SQL and Results as shown in Figure 6-8 on page 256.

Selecting a table

On the Tables tab there is a window in which to select tables from the host, and specify what type of SQL statement you want to use:

- Select
- Select Unique
- Insert
- Update
- Delete

You can select multiple tables when performing a select, or select unique.

In this example we chose the Callup table (QGPL.DATA):

1. Select the **QGPL.DATA** checkbox.
2. Since no join is required in this example, click **Condition 1** tab, or **Next** twice.

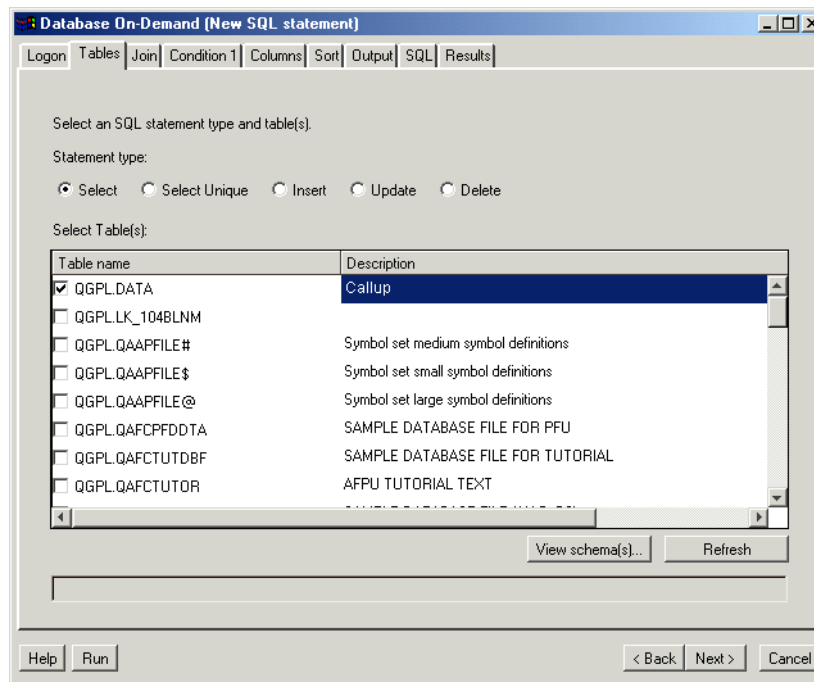


Figure 6-8 Database On-Demand Tables tab

Selecting Conditions

We want to find all the people at a certain zip code, in this case 27709. On the Condition 1 window (Figure 6-9 on page 257), the selected table should be displayed in the Selected table(s) field. If it is not, click the pull-down menu to select it.

1. Select **ZIP** in the Columns pane.

You will see that it displays the field type and field length, such as ZIP and DECIMAL(5) towards the bottom of the window.

2. In the Operator pane, select **is exactly equal to**.
3. In the Values pane, enter 27709.

If required, you can see all the values for ZIP by clicking **Find...** and leaving the Search for field blank, and all the values for ZIP will be returned. You can select **27709** and click **Use values** to use that value.

4. Click **Next** or the **Columns** tab to continue.

If you have more than one selection criteria, you can add further conditions by clicking **Find** on another column. This will also add another Condition tab.

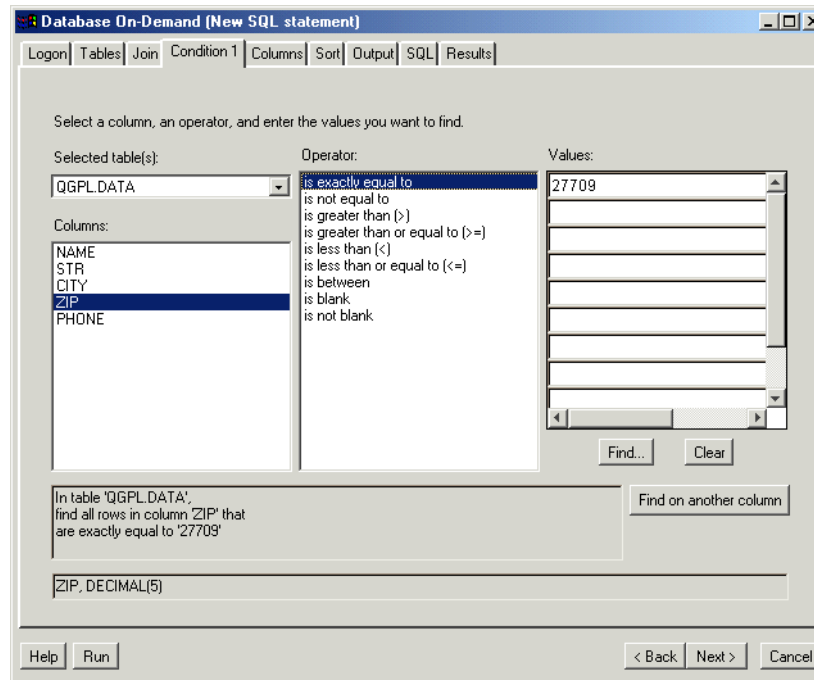


Figure 6-9 Database On-Demand Condition 1 tab

Selecting columns

The Columns window, shown in Figure 6-10, allows us to select which columns we want shown in the results. All available columns for the selected table, in this case QGPL.DATA, will be displayed in the left-hand pane labeled Columns. We want all columns displayed in the results, so we add them to the included columns by the following steps:

1. Click **Select all**.
2. Click **Add >>**.

You can change the order that the columns are displayed in the results by selecting the included column in the right-hand pane, and either clicking **Move up** or **Move down** as desired.

3. Click **Next** or click the **Sort** tab to continue.

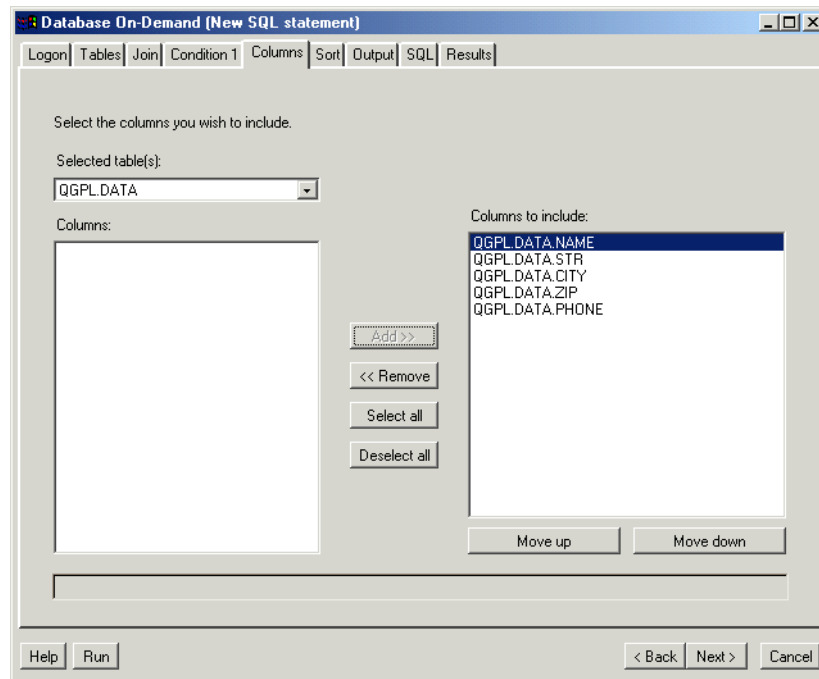


Figure 6-10 Database On-Demand column selection

Selecting a sort sequence

The Sort tab, shown in Figure 6-11 on page 259, allows you to specify the sort order for the columns you included in column selection. You can sort in either ascending (a->z) or descending (z->a) order. We will sort on ascending on the Name column, so the procedure is as follows:

1. In the left-hand pane labeled Column, click **NAME**.
2. Confirm the sort order is Ascending, which is the default.
3. Click **Add >>**.
4. Click **Next** or **Output** to continue.

If you are sorting on more than one column, you have the ability to change the order in which the columns are sorted by moving them up or down in the right-hand pane labeled Columns to sort on.

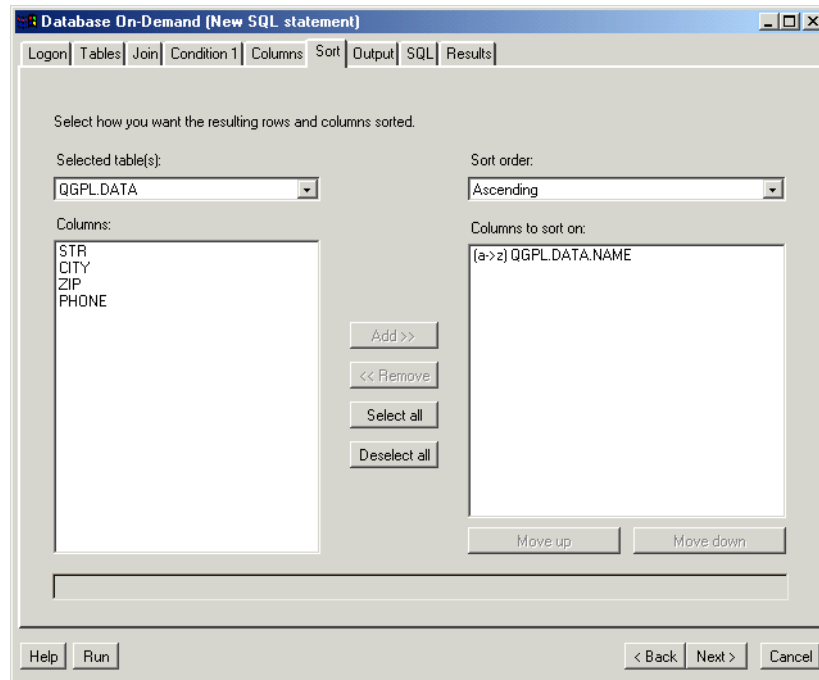


Figure 6-11 Database On-Demand Sort tab

Selecting the output target

The Output window, as shown in Figure 6-12, lets you send the query results to the display or to a file. Sending it to the display is quite simple, and generally the defaults provided on this window will be suitable. When sending the output to a file, there are various file types available, and in this case, we will send it to an HTML file. We will also use another HTML as a template for the results.

1. Under Output query results to, click **File**.

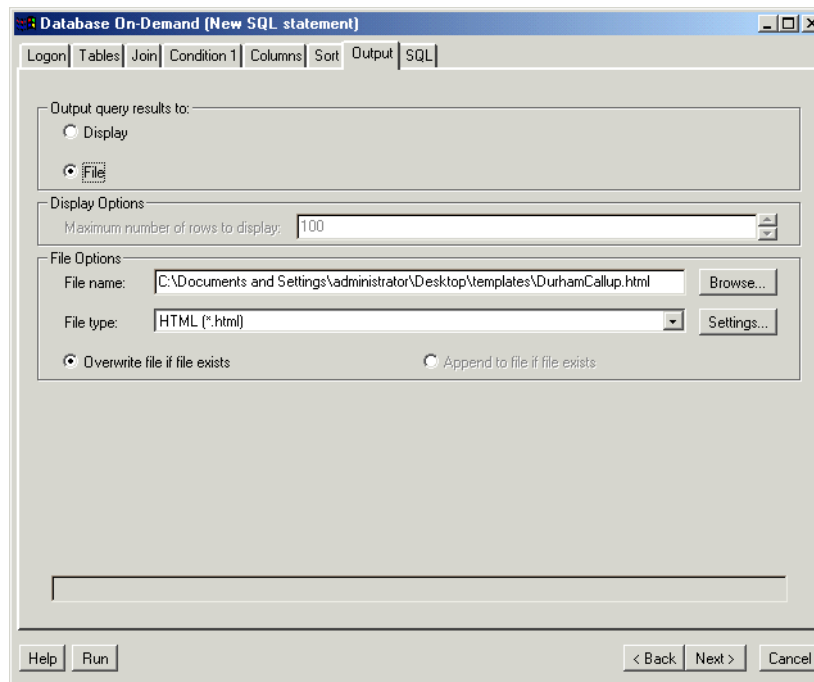


Figure 6-12 Selecting the output destination

2. Type in the file name.
3. Select **HTML** as the file type.
4. Click **Settings**.

You will be presented with the HTML Table Settings window (Figure 6-13 on page 261).

5. Click **Use HTML File as template**, and either type in the file name, or locate the file using the Browse button.

In this example we will use DurhamCallupTemplate.html, the contents of which can be seen in Example 6-1 on page 261. This HTML file was created earlier using an editor such as NotePad. When using an HTML template, the default is to have the HTML comment line `SQLTable` replaced with the query results. This can also be modified by changing the Template Tag field.

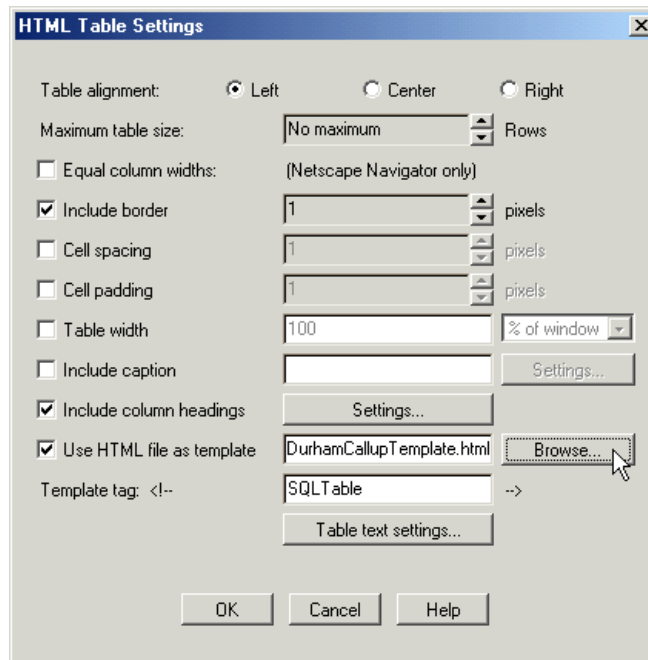


Figure 6-13 HTML table settings, specifying an HTML template file

6. Click **OK** to close the HTML Table Settings window.
7. Click **Next** or the **SQL** tab to continue.

Example 6-1 Contents of DurhamCallupTemplate.html

```
<HTML>
<HEAD>
<META content="text/html">
<title>Durham Callup Listing</title>
</HEAD>
<BODY>

<h3>Durham (27709) Callup Listing</h3>

<!-- SQLTable -->

<font size ="-1">This is a listing for all people with a 27709 zipcode</font>
</BODY>
</HTML>
```

Viewing, saving and running the SQL

The SQL window will appear as shown in Figure 6-14 as the default. If the administrator has checked **Allow manual editing of SQL statements** when configuring your user ID or group, it would appear as the window shown in Figure 6-15. Either window presented allows the user to copy the SQL to the clipboard, save the SQL, or run the query.

Executing the query can be done one of two ways:

- ▶ Click **Run SQL**.
- ▶ Click the **Run** button in the bottom left-hand corner of the window.

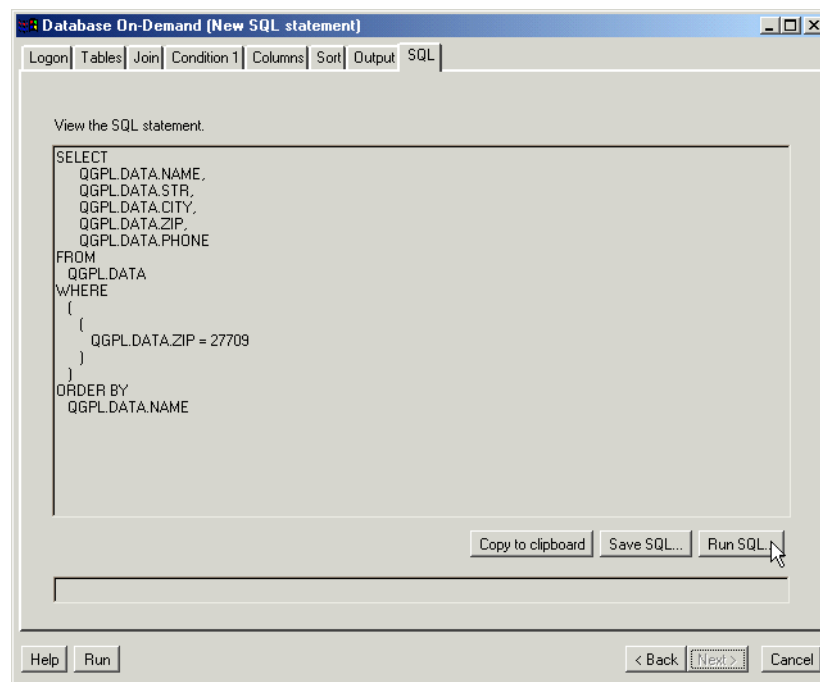


Figure 6-14 SQL tab as the default, with no SQL editing enabled

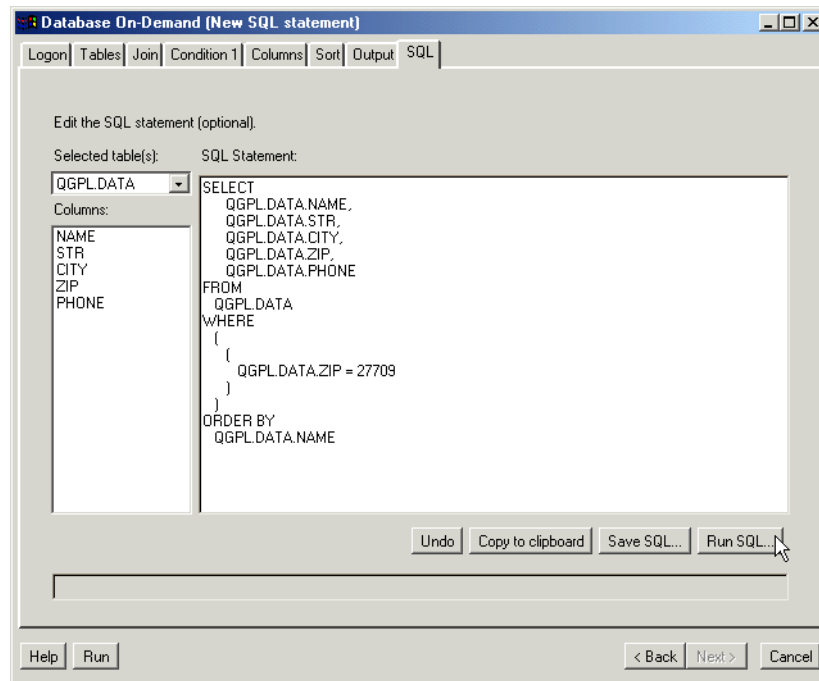


Figure 6-15 SQL tab showing manual editing of SQL enabled

Viewing the results

In this example, the results were written to an HTML file, and the output can be seen in Figure 6-16.

Durham (27709) Callup Listing

NAME	STR	CITY	ZIP	PHONE
Achim Tepper	700 Park Office Drive	Durham, NC	27709	919-543-5555
Achim Zorn	700 Park Office Drive	Durham, NC	27709	919-543-5559
Byron Braswell	700 Park Office Dr	RTP, NC	27709	919-543-4028
Carla Sadtler	700 Park Office Dr	RTP, NC	27709	919-543-4444
Carol Parks	700 Park Office Drive	Durham, NC	27709	919-543-5556
Casey Cooley	700 Park Office Drive	Durham, NC	27709	919-543-5557
David Skerrett	50 Melbourne Drive	Durham, NC	27709	919-544-1112
George Baker	700 Park Office Dr.	Durham, NC	27709	919-543-4005
Greg Mebberson	1 Harbour Drive	Durham, NC	27709	919-544-5555
Hari Balakrishnan	700 Park Office Drive	Durham, NC	27709	919-543-5558
Kelly Chen	1 Miller Street	Durham, NC	27709	919-544-7777
Tom Barlen	700 Park Office Dr	RTP, NC	27709	919-543-4244

This is a listing for all people with a 27709 zipcode

Figure 6-16 Results of SQL query when viewed in a browser

If instead it was decided to send the results to the display, it would appear as shown in Figure 6-17 on page 265. From this window, you can copy the results to the clipboard, save the SQL, and save the results.

If you click **Save SQL**, the statements are saved in a file called DBX.userid (where userid is your user ID) in the hostondemand\private directory on the server, with a name that you are asked to provide.

If you want to save the results of the query, click **Save Results**. The file types are the same as those provided on the Output tab, and it would be possible to write it to the same HTML file using a template, as we have done earlier.

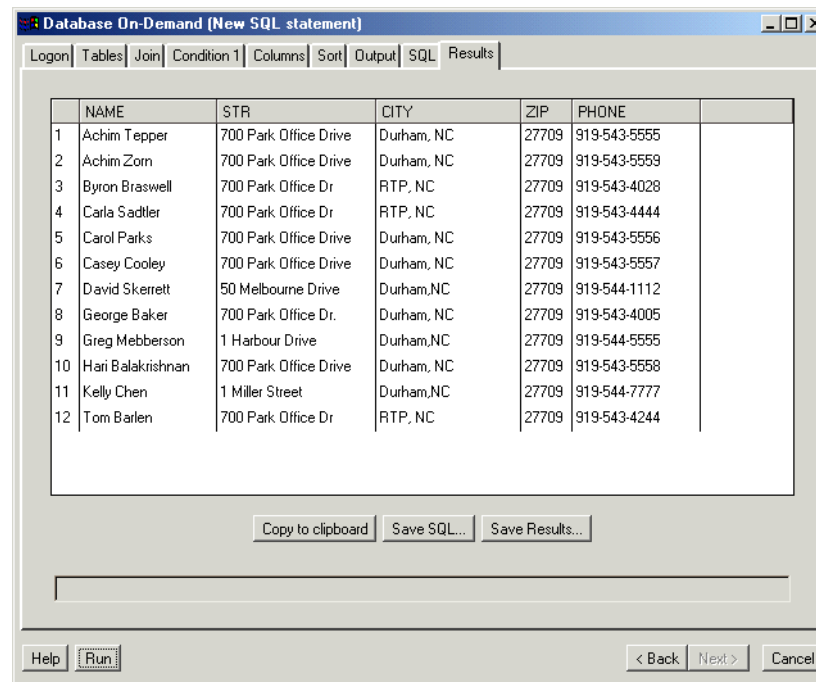


Figure 6-17 Results when sent to the display

When you have finished with this query, you can either modify the query and re-run it, save the SQL, or click **Cancel** to end the query.

6.2.2 Running an SQL statement

You can run any SQL statement that is listed on your Saved SQL Statements list.

To run a query, highlight the statement you want to use, then click **Run**. When the query has run, the results will be displayed.

If you did not check **Save password with statement** when saving the SQL, you will be prompted for your password prior to the SQL running.

6.2.3 Changing an SQL statement

You can view and edit the options used to create an existing SQL statement. This lets you make changes without having to recreate the SQL each time.

6.2.4 Deleting an SQL statement

You can delete SQL statements from your SQL Statements window, but once they are deleted, they cannot be recovered. SQL statements saved at the group level must be removed by the administrator. Icons alongside the SQL statement indicate whether it is at the group or individual level.

6.2.5 Customizing user options

Users can customize the behavior of Database On-Demand to suit their needs. For example, if you are always connecting to the same server, you can save the default logon values for the database name, user ID, password and driver to be used, as shown in Figure 6-18.

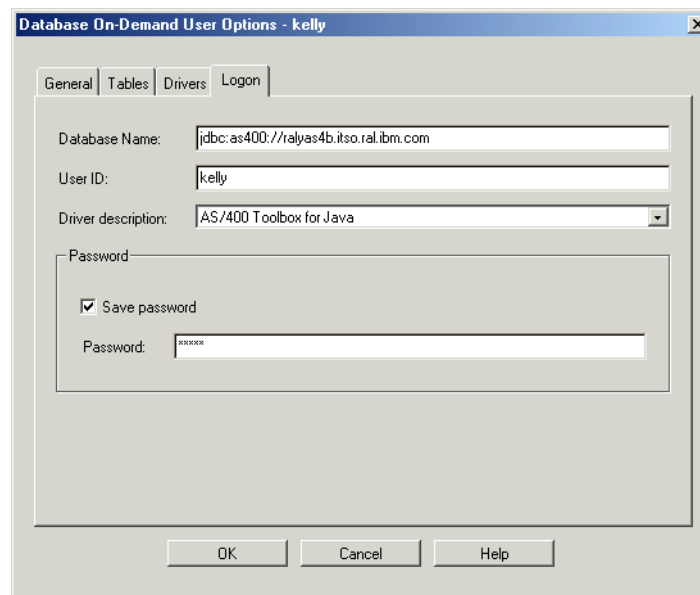


Figure 6-18 Setting default logon parameters

6.3 Installing and registering other JDBC drivers

There are now many JDBC drivers available, although only the iSeries driver is provided and supported with Database On-Demand.

By way of example, let's install the JDBC driver for IBM DB2 so that you can use Database On-Demand to access DB2 databases. The DB2 JDBC driver is in the db2java.zip file, which is located in [drive:]\SQLLIB\java\ after the installation of the CAECLIENT.

6.3.1 Installing a driver

New JDBC drivers must be installed in the same directory path as the AS/400 driver that is provided with Host On-Demand. For example, if the current directory for the default AS/400 JDBC driver is:

```
[drive:]\hostondemand\HOD\com\ibm\as400\access
```

Where drive is the drive letter where Host On-Demand is installed, you must unpack db2java.zip to [drive:]\hostondemand\HOD and the following directory structure will be created for the applet driver:

```
[drive:]\hostondemand\HOD\com\ibm\db2\jdbc\net\DB2Driver
```

6.3.2 Registering a driver

Either an administrator or user can register a driver. The procedures are as follows.

Registration by the administrator

Administrative registration begins by opening the Database On-Demand Options window by right-clicking either a group or user and selecting **Database > Options...** as shown in Figure 6-1 on page 249.

1. Click the **Driver** tab (see Figure 6-19).
2. Type in a description for the driver; in this example we entered:
DB2 Applet driver
3. Type in the class name of the driver, *ensuring the case is correct*:
COM.ibm.db2.jdbc.net.DB2Driver
The class extension is not specified.
4. Click **Register Driver**.
5. Click **OK** to save the new drivers.

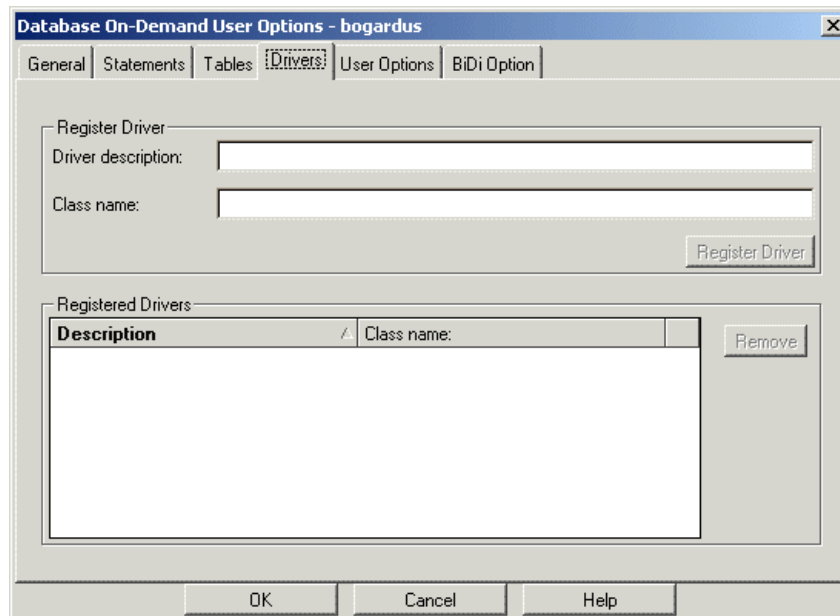


Figure 6-19 Registering DB2 JDBC driver by the administrator

Registration by a user

To register a driver, you must click **Options** on the initial Database On-Demand window. From this point you should follow the numbered steps in "Registration by the administrator" on page 267.

6.3.3 Using a new driver

Start a new SQL query from the Database On-Demand applet (see Figure 6-6 on page 253):

1. Click **New**.

You will then see the Logon tab similar to Figure 6-20 on page 269. This window is shown with all steps completed, and ready to connect to a DB2 database.

2. Type in the Database URL, such as:

```
jdbc:db2://bigtex.itso.ral.ibm.com/Sample
```

3. Fill in the Userid and Password fields.

4. In the Driver description, from the drop-down list select the DB2 driver that you just registered, DB2 Applet driver. You will see the Class name change to the correct driver.

5. Click **Connect**.

Once connected, generating an SQL statement and navigating the windows is the same as for the iSeries database, explained earlier in 6.2.1, “Creating a new SQL statement” on page 254.

Database On-Demand (New SQL statement)

Logon

Connect to a database

Database URL:
jdbc:db2://bigtex.itso.ral.ibm.com/Sample

Userid: kelly Password: xxxxxx

Driver description: DB2 Applet driver Class name: COM.ibm.db2.jdbc.net.DB2Driver

Connect

Enter the required information and click 'Connect' to begin.

Help Run < Back Next > Cancel

Figure 6-20 Connecting to a DB2 database

Note: The DB2 database must reside on the Web server; otherwise a Security Exception will occur. The JDBC driver will attempt to establish a separate network connection with the DB2 database through the JDBC applet server residing on the Web server. Java will trigger a security exception if the servers are different. More information can be found in the online *DB2 Application Building Guide*.

6.3.4 Common access problems

Sometimes, when you try to connect to a database, an SQL Assist Exception occurs. Such exceptions generally indicate that something on the Logon window is incorrect. Following are some examples of exceptions that could occur.

Application requester cannot establish the connection

Check that you entered the database name completely. For example, the error shown in Figure 6-21 can occur if the name is `as400.raleigh.ibm.com`, but you only typed `as400` and this host name could not be resolved by a name server.

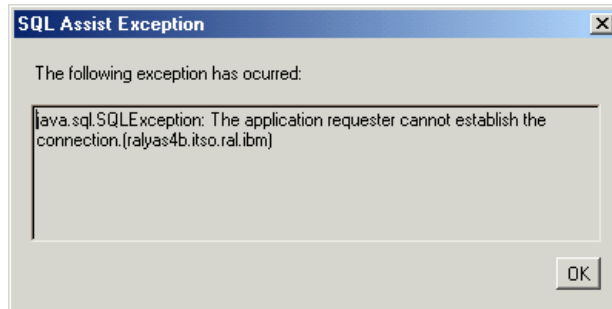


Figure 6-21 Error establishing database connection

No suitable driver

The no suitable driver error, shown in Figure 6-22, can occur if any part of the database name is incorrect. Below are some common errors.

- ▶ Correct syntax:
`jdbc:as400://bigtex.raleigh.ibm.com`
- ▶ Incorrect syntax:
`jdbc:as400//bigtex.raleigh.ibm.com`
- ▶ Incorrect syntax:
`jdbc:as40://bigtex.raleigh.ibm.com`

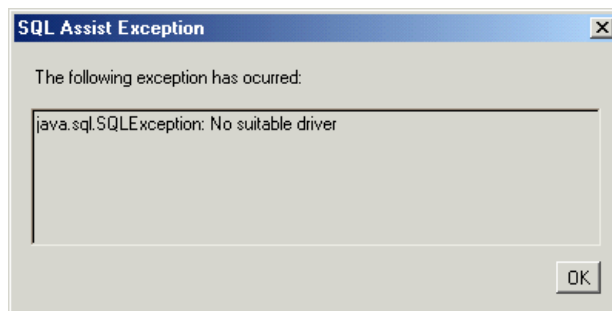


Figure 6-22 Error finding driver

Security exception

If you attempt to access a DB2 database that resides on a server different from the one that served the applet, you will receive a security exception as shown in Figure 6-23. This type of activity violates Java security. In our example, the DB2 database is on `bigtex.itso.ral.ibm.com`, and the Web server is on `mk23bk67f`. Internet Explorer and Netscape Navigator report the error differently. The error window shown at the top of Figure 6-23 is generated by Microsoft Internet Explorer, while the error window generated by Netscape Navigator is shown at the bottom.

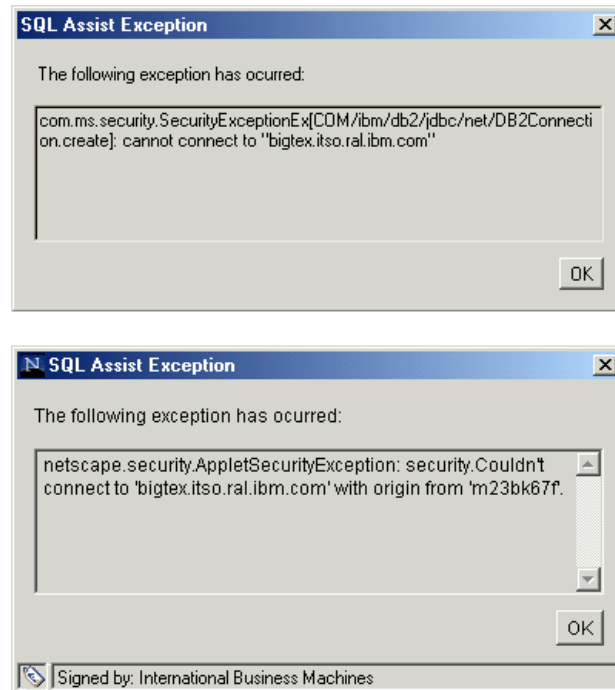


Figure 6-23 Error when attempting to connect to a remote DB2 database



Administration

The Service Manager is the component of Host On-Demand that controls the following Host On-Demand functions and its data stored on the configuration server:

1. Users/Groups management (Configuration Server)
2. Services management
3. Redirector Service management
4. Directory Service management
5. OS/400 Proxy Server management
6. Licenses management

Administration of a Host On-Demand server is done primarily through the administration applet, `HODAdmin.html`, which is loaded into your browser. If you are working on a Windows NT or Windows 2000 server, click **Start -> Programs -> IBM Host On-Demand -> Administration -> Administration Utility**. On all other platforms or at a workstation, load the applet by entering the following URL:

`http://[server_name]/hod/HODAdmin.html`

For access to the administrative functions, you must log on using an administrative user ID and password. The default user ID supplied by Host On-Demand is `admin` and the default password is `password`.

7.1 Manage users and groups

The Users/Groups task of the Administration Notebook, shown in Figure 7-1 on page 275, lets you:

- ▶ Manage groups.

If you are using an LDAP directory to store configuration information, groups may be hierarchical. The default data store does not allow for hierarchical groups.

- ▶ Manage users.

If you are using the default data store, users may belong to multiple groups; however, if you are using an LDAP directory to store preferences (see 7.4, “Directory Service” on page 357), a user may belong to only one group.

- ▶ Manage sessions for users or groups.

If you create a session for a group, all the users that you assign to that group inherit the session and all its settings.

- ▶ Copy, change or delete users, groups or sessions.

- ▶ Look at a trace file that has been created by a user and saved to the server.

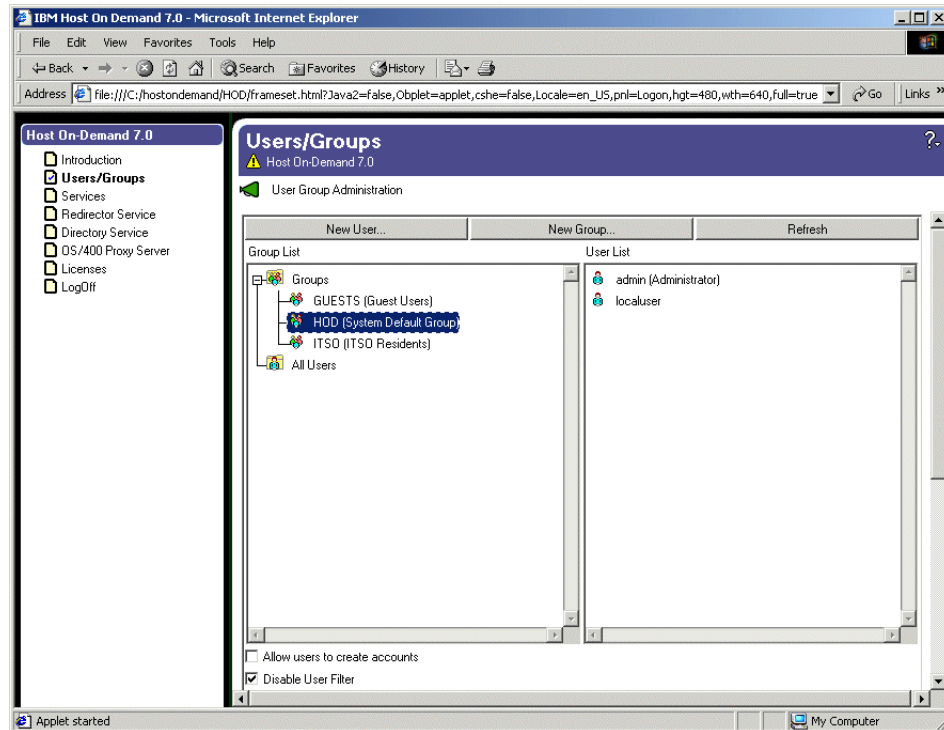


Figure 7-1 Users/Groups Administration window

When dealing with a large number of groups, users, or sessions, you may find it convenient to use Directory Utility. Directory Utility is a command-line Java application the administrator can use to manage user, group or session configuration information. This information is stored either in the Host On-Demand default data store, or in an LDAP directory. This utility is only useful in the environment where the Configuration Server-based model is in use. Directory Utility allows you to add, delete, or update large numbers of users, groups, or sessions in a batch mode environment instead of using the Administration client. Directory Utility reads an XML ASCII file that contains the following actions to be performed on users, groups, or sessions defined to the Configuration Server:

- ▶ Add, update, and delete groups
- ▶ Add, update, and delete users from groups
- ▶ Add, update, and delete sessions from users or groups

For details see 7.7, “Directory Utility” on page 367.

7.1.1 Planning

Host On-Demand Version 7 offers three different deployment models:

1. Configuration Server model

Initial configuration data is stored at the server and user modifications are stored on the server.

2. The HTML-based model

The configuration is deployed in a customized HTML file, and user modifications are saved locally at the client system.

3. The combined model

The client retrieves the group configuration from the server; however, user modifications are saved locally at the client system.

See Chapter 13, “Deployment strategies” on page 513 for a detailed explanation of these models.

7.1.2 Manage groups

Each user must be a member of at least one group. You may use the Host On-Demand provided default group, HOD, or you may create a group into which you will add users. When using the default data store, all groups are at the same level, and users may belong to multiple groups. However, if you store your preferences in an LDAP directory you may organize your groups hierarchically, but a user belongs to a single group and its hierarchy.

To add a new group, click **New Group**. If you are using the default data store, the window shown in Figure 7-2 will appear, and if you are using an LDAP directory, the window shown in Figure 7-3 on page 277 will appear.

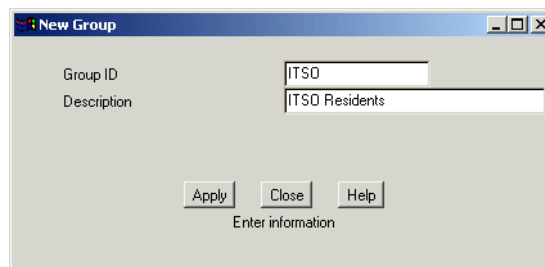


Figure 7-2 Configuring a group

You must enter a group ID. The first character must be a letter, and all other characters must be the English equivalent of A-Z, a-z, 0-9, . (period), and - (hyphen). Group IDs are always converted to uppercase unless you are using an LDAP directory server data store where mixed-cased characters are allowed.

A group description is optional. Any character is allowed except | (vertical bar) or # (number or pound sign).

If you are using the LDAP directory, you must select the hierarchy that the group will occupy. In the example shown in Figure 7-3, the group 6182book will be added under the group ITSO. The net result is that any user in the 6182book group will inherit sessions defined at the ITSO and the 6182books group level.

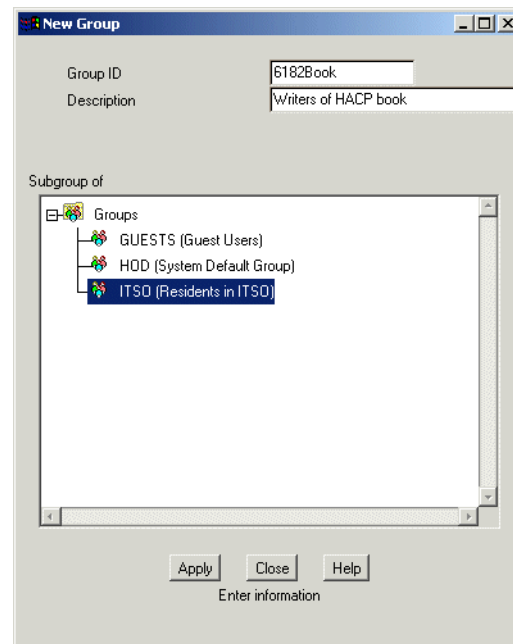
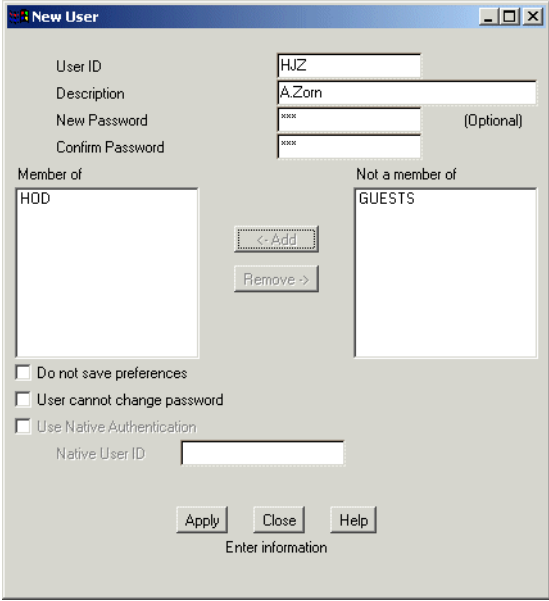


Figure 7-3 New group with LDAP

To store the group, click **Apply**. The window will remain open so that you may add another group. When you are finished adding groups, click **Close**.

7.1.3 Create a new user

To add a new user, click **New User**. If you are not using the LDAP directory to store preferences, the window shown in Figure 7-4 will appear.



The image shows a 'New User' configuration window. It has a title bar with a small icon and standard window controls. The main area is divided into several sections. At the top, there are four labels: 'User ID', 'Description', 'New Password', and 'Confirm Password'. To the right of these labels are input fields. The 'User ID' field contains 'HJZ'. The 'Description' field contains 'A.Zorn'. The 'New Password' and 'Confirm Password' fields are empty and have '(Optional)' written to their right. Below these fields, there are two list boxes. The left one is labeled 'Member of' and contains 'HOD'. The right one is labeled 'Not a member of' and contains 'GUESTS'. Between these two list boxes are two buttons: '<- Add' and 'Remove ->'. Below the list boxes, there are three checkboxes: 'Do not save preferences', 'User cannot change password', and 'Use Native Authentication'. The 'Use Native Authentication' checkbox is checked. Below the 'Use Native Authentication' checkbox is a label 'Native User ID' followed by an empty input field. At the bottom of the window, there are three buttons: 'Apply', 'Close', and 'Help'. Below these buttons is the text 'Enter information'.

User ID: HJZ
Description: A.Zorn
New Password: (Optional)
Confirm Password: (Optional)

Member of: HOD
Not a member of: GUESTS

<- Add
Remove ->

☐ Do not save preferences
☐ User cannot change password
☒ Use Native Authentication

Native User ID:

Apply Close Help
Enter information

Figure 7-4 Configuring a user

If you are using the LDAP directory server for preferences, you will see the window shown in Figure 7-5.

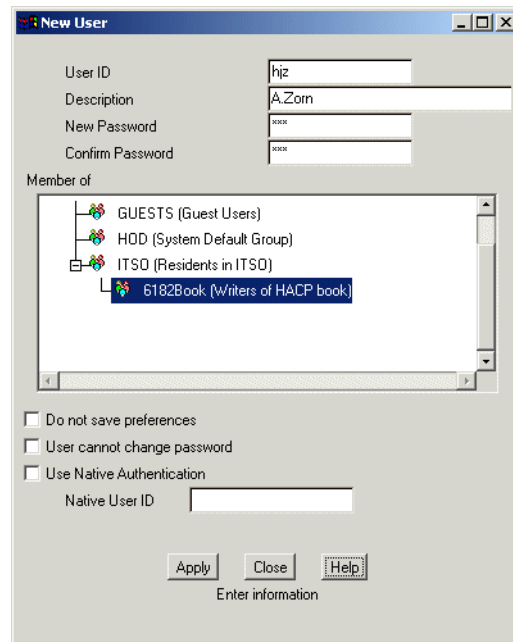


Figure 7-5 New user using LDAP directory

You must now complete the following elements of the window:

► User ID

The Host On-Demand user ID being created. The first character must be a letter. Valid characters are A-Z, a-z, 0-9, . (period), and - (hyphen). When using Host On-Demand to store configuration information, user IDs are converted to lowercase characters. User IDs must be unique, and must not match group IDs regardless of case.

Note: Windows users can define Host On-Demand user IDs that are identical to their corresponding Windows domain user IDs, users who log on to their Windows domain user IDs do not have to log in again to access their Host On-Demand sessions

► Description (optional)

A brief description of the user ID being created. Suggested contents: the full name of the user or a description of a group for a shared user ID. You can use any character except | (vertical bar) and # (number or pound sign).

► New Password (optional)

The user's password. Passwords are not required.

► Confirm Password (optional)

Repeat User's Password for confirmation. Not available when Native Authentication is selected.

► Member of

Each user must be a member of at least one group. If you are using LDAP, a user can be a member of only one group. Select the group that you want the user to be a member of. Unlike default Host On-Demand users, natively authenticated users cannot belong to multiple groups. The first character must be a letter. Valid characters are A-Z, a-z, 0-9, . (period), and - (hyphen). Group IDs are always converted to uppercase if the default Host On-Demand data store is used.

Note: When using the default data store, we recommend that you not place more than 1000 users in any one group.

► Do not save preferences

If selected, the user may be able to change items, such as emulator colors, but the changes will not be saved.

Users can be denied access to making preference changes. See 6.1.8, "Disabling emulator functions" on page 211 for details.

► User cannot change password

Prohibits a user from changing their password. When defining a natively authenticated user, this will be selected automatically.

► Use Native Authentication

Check this box to use the Native Authentication feature (only enabled when LDAP is used). Refer to 7.1.4, "Using Native Authentication" on page 281.

► Native User ID

This is the user ID that will be passed to the native operating system. This can be different from the Host On-Demand user ID. See 7.1.4, "Using Native Authentication" on page 281. If you are running on an AIX or UNIX operating system, ensure that this ID is set to the proper case, because IDs are case sensitive in these environments.

7.1.4 Using Native Authentication

The native platform authentication service allows users to logon to Host On-Demand using the same password as they would to logon to the operating system (Windows NT, AIX or z/OS) where Host On-Demand is active. When a user logs on to Host On-Demand, their password is validated against the system password, rather than a separate Host On-Demand password. This gives the Administrator a single point of control for password administration, and the user a single password to remember.

When a user logs on the following sequence as shown in Figure 7-6 on page 281 takes place:

1. The user ID and password are sent to the Host On-Demand service manager.
2. The service manager sends a request for logon information about the user to the LDAP server.
3. The LDAP server returns the requested user information and whether or not the user is configured for native authentication.
4. If the user is configured to use native authentication, the service manager sends the authentication user ID and the password to the operating system for verification. If the user is not configured for native authentication, the service manager compares the password that was entered by the user with the password returned by the LDAP server.

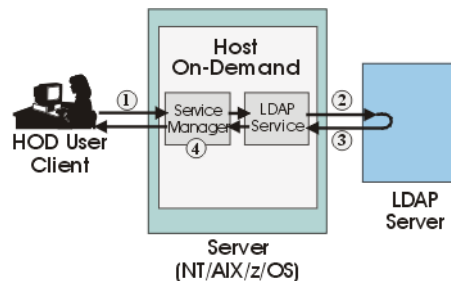


Figure 7-6 Process of native authentication

Use of the LDAP directory server must first be enabled as explained in 7.4, "Directory Service" on page 357 in order for the Use Native Authentication check box to be enabled. To enable a user for Native Authentication, select the **Use Native Authentication** check box, as shown in Figure 7-7.

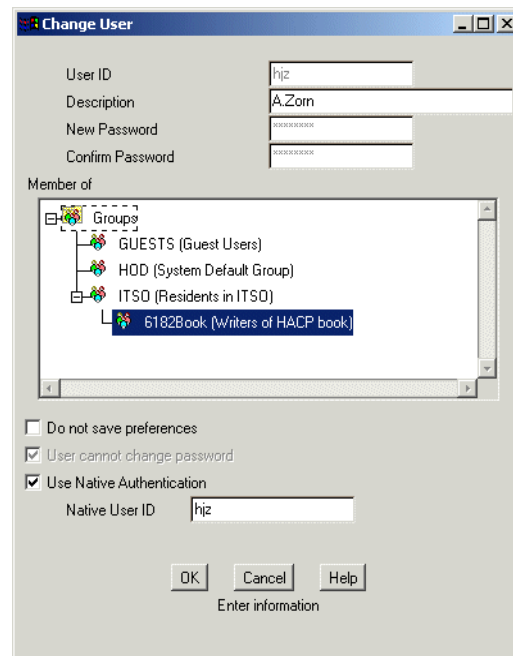


Figure 7-7 Configuring a user with Native Authentication

A bit more explanation is in order on the relationship between the Host On-Demand user ID/password and the native user ID/password. The rules are pretty simple in this relationship:

- ▶ The Host On-Demand user ID and password are a Host On-Demand administrative convenience. The user ID acts only as an index to the configuration data stored by Host On-Demand. It can be whatever you want it to be (within the Host On-Demand naming rules).
- ▶ If a password was previously specified, it will be ignored when you enable Native Authentication. It will remain in the database and if you ever disable Native Authentication for the user it will be reactivated.
- ▶ In Native Authentication mode, all password handling is done by the native operating system; therefore, the User cannot change password check box is disabled. If the native password expires and the user attempts to log on to Host On-Demand, the Host On-Demand logon will fail. The user must use an operating system interface to change the password before logging on to Host On-Demand.
- ▶ You must be careful with the native user ID if the Host On-Demand server is running on an AIX or UNIX system. On AIX and UNIX systems, the native user ID is case sensitive. Therefore, make sure the native user ID is specified

with the proper case. There is no translation of this field by Host On-Demand and case sensitivity is maintained.

- ▶ By default, Host On-Demand will translate all passwords entered at the logon window to lowercase before validating them, or forwarding them to the native system for authentication. Windows, AIX and UNIX servers all respect case sensitivity when dealing with passwords; therefore, if your Host On-Demand server is running on Windows NT, AIX or any UNIX server, you should insert the following parameter into the NSMprop file (found in the \hostondemand\lib subdirectory) to ensure proper processing of passwords:

```
LowerCasePasswords=false
```

Once you set this parameter, all passwords will be case sensitive, even for those users not using Native Authentication.

7.1.5 Administering groups, sessions and users

There is much that can be done by using standard GUI manipulation of the objects presented on the Users/Groups window shown in Figure 7-1 on page 275. All operations on Host On-Demand groups and users are performed by using the context (pop-up) menus. Operations using the context menus can be performed on only one group at a time. However, more than one user can be selected for a given operation using the mouse or the arrow key on the keyboard. Follow the standard Windows conventions. For users not familiar with Windows, the following tips will help:

- ▶ Clicking the user (or using the spacebar) with the mouse selects that user and deselects any other user(s).
- ▶ Clicking the user (or using the spacebar) while pressing the Ctrl key selects additional users.
- ▶ Clicking the user (or using the spacebar) while pressing the Shift key will select all users between one that is already selected up to and including the current user.

Note: The context menu is displayed when clicking the right mouse button. It will allow only those functions that are allowed in that context. For example, defining the host sessions available to a user can be done only at the individual user or at a single group level.

If you check the **Allow users to create accounts** check box in the Users/Groups window, you must provide an HTML file through which the users can create their own accounts. A sample file, `NewUser.html`, is located in the publish directory (the default is `/hostondemand/HOD`). You can use the sample file or create customized versions of it using the Deployment Wizard. Additional information is found at 5.4.1, “New user client” on page 175.

Note: For performance reasons, it is recommended that you place no more than 1000 users in any one group.

7.1.6 Filtering

If you have a large number of users, you may wish to use the Filter option to restrict the number of users displayed at one time. Filters are used to view the users within a group or the users in the All Users folder. There are two ways a filter can be used:

1. By using the filter option of the context menu for a group. This is a one-time use of filtering that allows the administrator to view a subset of a group. If another group is selected, the administrator will be shown an unfiltered view of that group, unless the filter context menu is used for that group also.

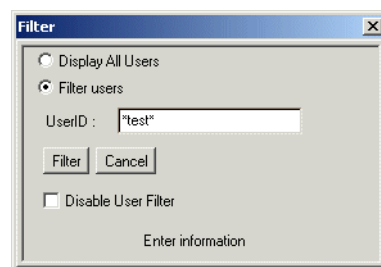


Figure 7-8 Selecting only the users containing “test” in their userids

2. Globally enabling the filter option. This is set by unchecking the Disable User Filter check box in the lower left-hand corner of the User/Groups administrative window. When this is done, every time the administrator views a different group, he will be asked for a new filter to use.

If you are using the same filter to view all groups, it is advisable to copy the filter into the system clipboard and paste it into the filter window.

Note: Host On-Demand stores all user IDs in lowercase and the filter engine is case sensitive. Therefore, you should not use uppercase letters in the filter. For example, a filter on G* will not return the same list as g*.

7.1.7 Configuring sessions

If several users need the same connection, you should put them in a group and define the sessions for that group rather than defining the sessions for each user. If there are users that have unique session requirements, the sessions must be defined separately for each such user.

There is no difference if you are configuring a session for a user or a group. Simply double-click the user or group entry where you wish to define the session and the window shown in Figure 7-9 will appear. To add a new session, click the button for the type of session you want to configure.

From an full administrator (with start session capability) an existing session can be run directly from the administrator by pointing to the session icon, right click and select **Start Session** from the context menu. By that administrators can easily test the defined session. However he can start only one session at a time!

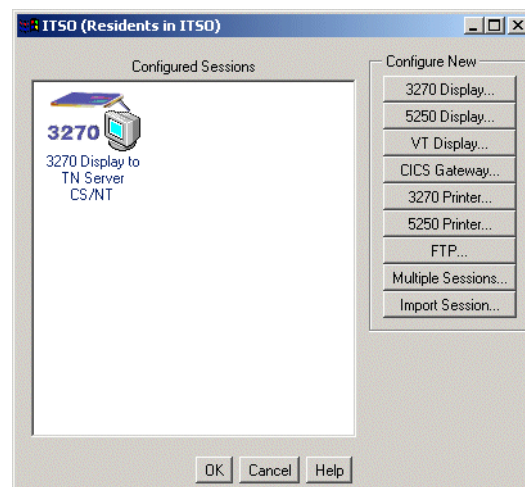


Figure 7-9 Session selection window

To copy a session display, open the context menu as shown in Figure 7-10 and click **Copy**, then find the group or user to whom you want to add the session and right-click again and click **Paste**.

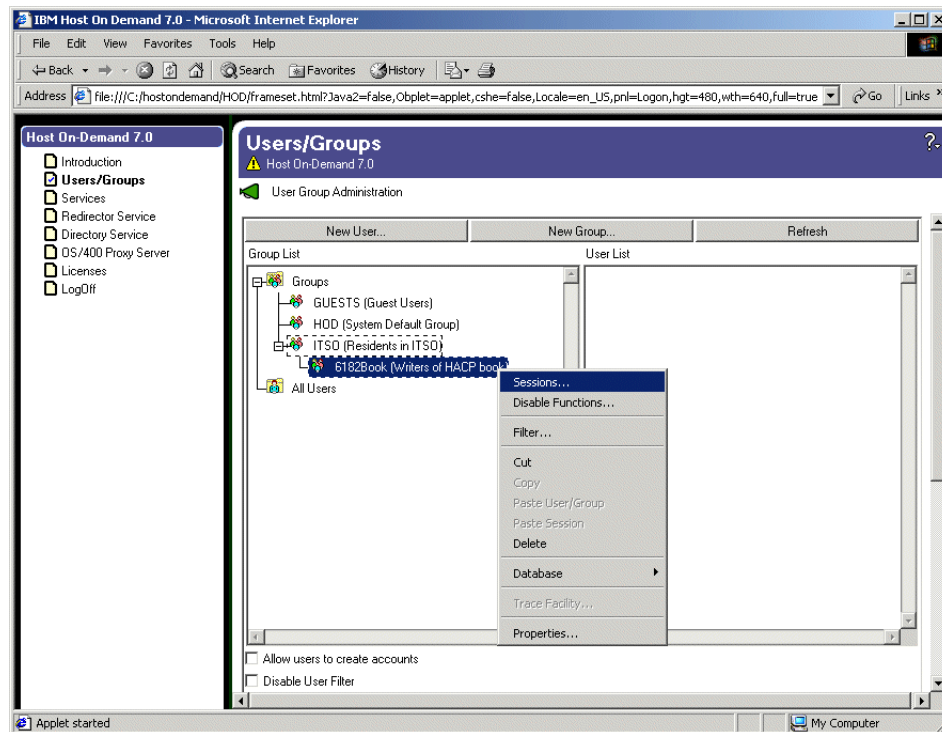


Figure 7-10 Administration context menu

If you would like to copy a session within the same user or group, click the right mouse button to display the context menu and click **Duplicate Session** rather than using copy and paste.

3270 and 5250 printer sessions

The setup of 3270 and 5250 printer sessions is similar to the display sessions. For parameters specific to print sessions please refer to chapter “Host printing” on page 661.

3270 and 5250 display sessions

The 3270 and 5250 sessions are very similar. The 3270 sessions will be used as an example except where there is a significant difference in the fields, then both will be explained. Fields unique to a specific emulator type are highlighted.

Selecting **3270 Display** from the window shown in Figure 7-9 brings up the window shown in Figure 7-11.

The screenshot shows a window titled "1:3270 Display" with a tabbed interface. The "Connection" tab is selected. The window contains the following fields and controls:

Field	Value	Lock
Session Name	3270 Display	<input type="checkbox"/>
Destination Address		<input checked="" type="checkbox"/>
Destination Port	23	<input checked="" type="checkbox"/>
Enable SLP	<input type="radio"/> Yes <input checked="" type="radio"/> No	<input checked="" type="checkbox"/>
TN3270E	<input checked="" type="radio"/> Yes <input type="radio"/> No	<input checked="" type="checkbox"/>
LU or Pool Name		<input checked="" type="checkbox"/>
Screen Size	24x80	<input type="checkbox"/>
Host Code-Page	037 United States	<input checked="" type="checkbox"/>
Associated Printer Session		<input type="checkbox"/>
Close Printer With Session	<input type="radio"/> Yes <input checked="" type="radio"/> No	<input type="checkbox"/>
File Transfer Type	Host File Transfer	<input type="checkbox"/>

Below the "File Transfer Type" field is a button labeled "File Transfer Defaults...". At the bottom of the window are buttons for "OK", "Cancel", "Keyboard...", and "Help".

Figure 7-11 3270 session Connection tab

Selecting **5250 Display** from the window shown in Figure 7-9 on page 285 brings up the window shown in Figure 7-12.

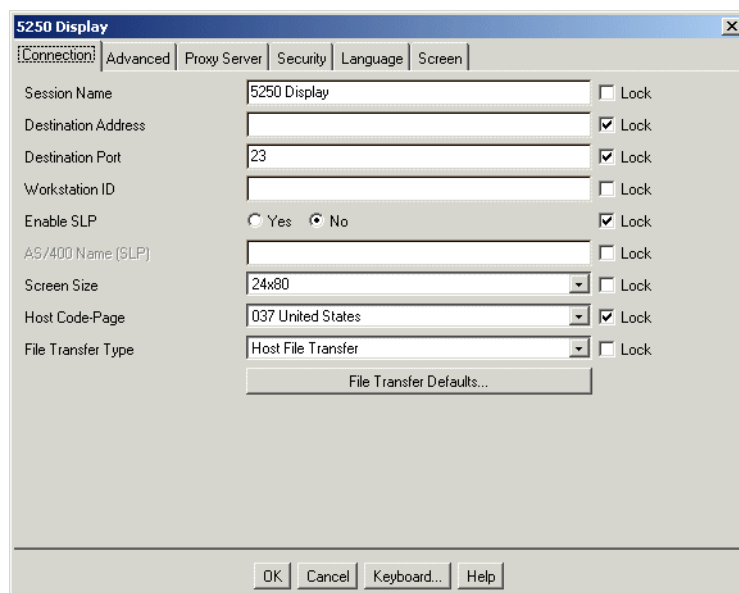


Figure 7-12 5250 session Connection tab

The remainder of this section will discuss the parameter on each of the tabs.

3270/5250 Connection tab

The parameters on this tab are connection oriented. Enter your data in the appropriate fields using the following descriptions:

► **Session Name**

This is the name you wish to assign to the session. It appears beneath the session's icon and at the top of the session window. Make sure that you do not give the same name to more than one session. If you do, you run the risk of a user having sessions of the same name if he or she is a member of more than one group.

► **Destination Address**

This is the host name or IP address of the Telnet server or gateway to which you want the session to connect. If the session will connect through the Host On-Demand Redirector, this must be the address of the Host On-Demand Redirector (see 6.1.10, "Redirector Service" on page 216).

If you are configuring a Host Printing session for use as only an associated printer session, you can leave this field blank. When the associated printer session starts, Host On-Demand will use the same address used by the display session.

► Destination Port

This is the port number on which the target server is listening for connections. If the session will connect to the Host On-Demand Redirector, this number must match the Redirector's Local Port number defined for this connection (see 6.1.10, "Redirector Service" on page 216).

The default port is 23 for 3270, 5250 and VT, and 2006 for CICS.

If you are configuring a Host Printing session for use as only an associated printer session, you can leave this field blank. When the associated printer session starts, Host On-Demand will use the same port used by the display session.

► Enable SLP

Enables you to dynamically find a service without knowing the destination port and address. Your network must be configured to support the Service Location Protocol (SLP). If SLP is supported in the network, Host On-Demand will connect to the server that responds with the least session load. If Enable SLP is Yes, there is no way to specify a specific LU name when establishing a connection, so you can use only pool names. For more details on SLP, refer to Appendix B, "Service Location Protocol" on page 1027.

► AS/400 Name (SLP) (5250 Session only)

Connects a session to a specific iSeries. Type the fully-qualified SNA CP name; for example, USIMBNM.RAS400B. If you do not specify an SLP iSeries name, the session connects to the default iSeries defined at the server. This field is available only when Enable SLP is Yes.

The first character must be A-through-Z, \$ (dollar sign), @ (commercial at sign), or # (number sign). The remaining characters can be A-through-Z, 0-through-9, \$, @, or #.

► TN3270E (3270 sessions only)

The extended TN3270 protocol is required if you want the session to connect to a specific LU or LU pool or if you want to use an associated printer.

► LU or Pool Name (3270 sessions only)

You might want to specify the name of an LU or LU pool, defined at a Telnet 3270E server, to which the session must connect. If you do not specify an LU or pool, the result depends on the type of server to which the session is connecting. Mostly, you will get a session from the server's default pool. If you are connecting to Communications Server/390 V2R10, you can enter the name of an LUGROUP for the POOL name.

When Enable SLP is Yes, this field can contain only a pool name. If a specific LU name is required (for example, for printing), do not use SLP.

If you are configuring a Host Printing session for use only as an associated printer session, this field can be left blank. When the associated printer session starts, Host On-Demand will determine the LU name from the Telnet server

► Workstation ID (5250 sessions only)

Defines the name of the workstation. Refer to 4.8.1, “5250 Workstation ID” on page 160 for a complete discussion of the options available. If you do not complete this field, a workstation ID is automatically defined by the host.

► Screen Size

The number of rows and columns in the session screen. The sizes are available from a drop-down list. The default is 24x80.

► Print-Buffer Size (3270 Printer only)

The size of the block of memory reserved for print data that is being sent to the printer. The valid values are in the drop-down list. This applies only to LU3 sessions.

The default Print-Buffer Size is 1920 (equals the 24 x 80 characters of the default screen size)

► Host Code-Page

Specifies the table used to map EBCDIC codes from the host to appropriate ANSI graphics on the workstation. The default is the code page that corresponds to the locale for which your workstation is configured, but you might need to change it, for example, if the session will connect to a host system in another country. You must set it to the code page supported by the host system to which the session will connect. For many countries, the default is the Euro version of the code page; only these will support the Euro currency symbol.

► Associated Printer Session (3270 sessions only)

TN3270E lets you associate a 3270 printer session with a 3270 display session. More information on associated printer sessions is found in 19.3, “3270 associated printer sessions” on page 682.

► Close Printer with Session (3270 sessions only)

This option is enabled only when an Associated Printer Session is selected. The default is No, enabling the printer session to be connected or disconnected independently of its associated display session.

If Yes is selected, the display session controls whether the printer session is connected or disconnected. The menu options to connect or disconnect are disabled in the printer session. When the display session is disconnected, the printer session is disconnected. When the display session is connected, the

printer session is connected. The printer window is closed when the display session window is closed. The printer window can also be closed independently, but the display window must be closed and restarted to restart the printer session.

When Yes is selected, the Session Inactivity Timeout for the printer session is ignored. When the display session times out, both the display and printer sessions are disconnected. If there is a job printing, the display and printer sessions are not disconnected until the print job completes.

When this option is set to Yes, the settings for the Auto-Connect and Auto-Reconnect on the printer session are derived from the associated display session, regardless of their configuration in the printer session. If Auto-Reconnect is set to No and the display session is disconnected by the telnet server or because of a network problem, the printer session will also be disconnected. If Auto-Reconnect is set to Yes and the display session is disconnected by the telnet server or because of a network problem, the printer session will remain connected until the display session is reconnected. If the display session LU is not associated with the running printer session LU, the printer session will be disconnected and reconnected to the display's associated printer LU.

It is recommended that you set this to Yes if the Telnet server is configured to use pooled associated LUs to ensure the printer and display LUs remain in sync.

► File Transfer Type

Determines if the file transfer type is Host File Transfer or FTP. If the File Transfer Type is FTP, a separate FTP session is started from the display session.

Note: Screen Customizer does not support the FTP client, so if the Screen Customizer session property is enabled in a display session's properties, then the supported File Transfer Type is Host File Transfer, not FTP.

Also note that FTP sessions do not support SSL even if the session is configured for SSL.

► File Transfer Defaults

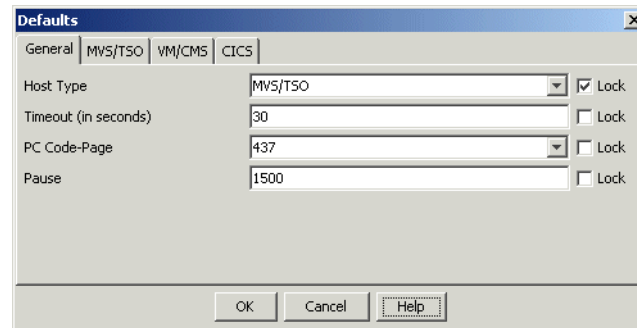


Figure 7-13 3270 file transfer

The 3270 host file transfer is implemented by using the IND\$FILE host program to transfer files via the 3270 data stream in DFT mode. In the General tab of the 3270 file transfer you select which is your default host type which is used for file transfer.

E.g. if you have in your company environment only hosts using VM, you select in the General tab for Host Type the VM/CMS from the drop down box. After that you configure your VM/CMS preferences using the VM/CMS tab. An example is shown in Figure 7-14 on page 292. Your users can start the file transfer from their 3270 session without needing to change / configure it.

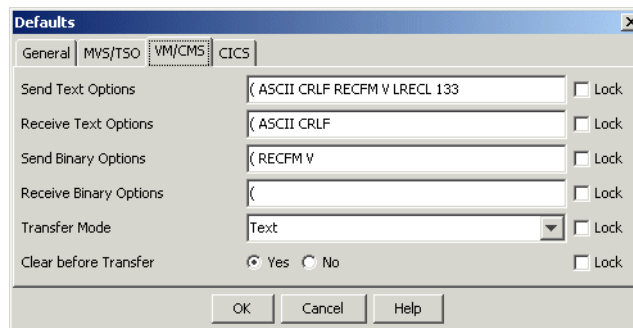


Figure 7-14 Example of VM/CMS file transfer settings

If you have more than one host type where users have to exchange data with, you can configure the preferences for each host type. But do not lock the Host Type in the General window, so that the user can change it. (The user has to switch the host type using Actions - File Transfer Defaults to select the host type to which he transfers data.) For more details refer to the online help.

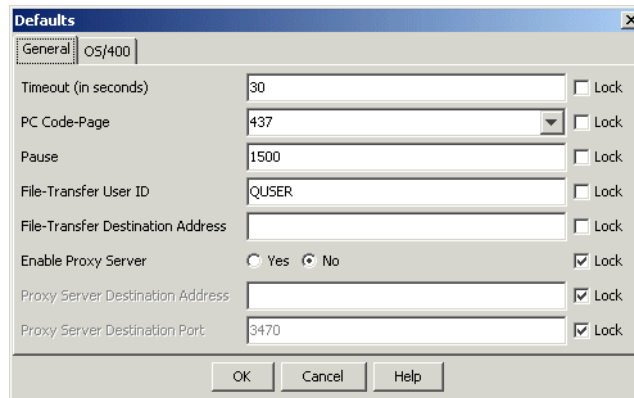


Figure 7-15 5250 host file transfer

In the 5250 file transfer window the defaults will vary depending upon what file transfer type is selected, Host File Transfer or FTP.

If Host File Transfer is selected you will see the window as shown in Figure 7-15 on page 293. For details refer to online help.

If FTP is selected, refer to **“.FTP session” on page 331** for an explanation of configuration.

► Keyboard button

Click at the keyboard button at the lower end of the session definition screen to get the keyboard remap window as shown in Figure 7-16 on page 294.

Using this feature, you can assign keys or key combinations as “shortcuts” to functions or applets. For example, you could assign Ctrl+m to execute a menu command or Alt+a to run an applet. Walk through the available Categories by clicking at the drop down box for that and view the assignments of the keys.

The example shows the Category Menu Commands, which contains e.g. the mapping for the cut and paste functions. Please view those examples and use the online help for this panel to understand to this utility.

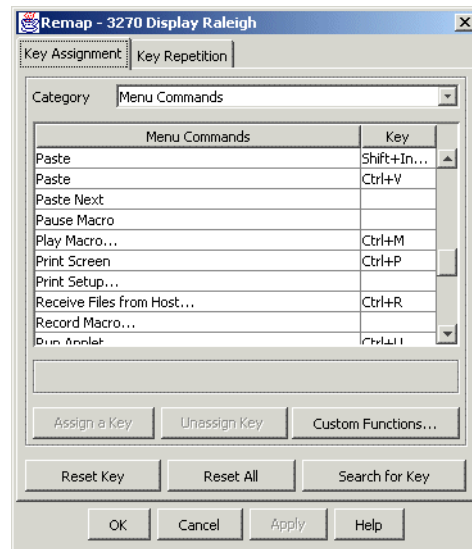


Figure 7-16 Keyboard remapping

► Lock

You can check **Lock** for any configurable parameter to prevent users from changing the associated startup value for that session. However, functions accessed from the session menu bar or tool bar can be changed. The lock option is available on all tabs of all sessions.

3270/5250 Advanced tab

The Advanced tab for 3270 and 5250 display sessions reflects operational characteristics of the session. The Advanced tab for a 3270 display session is shown in Figure 7-17.

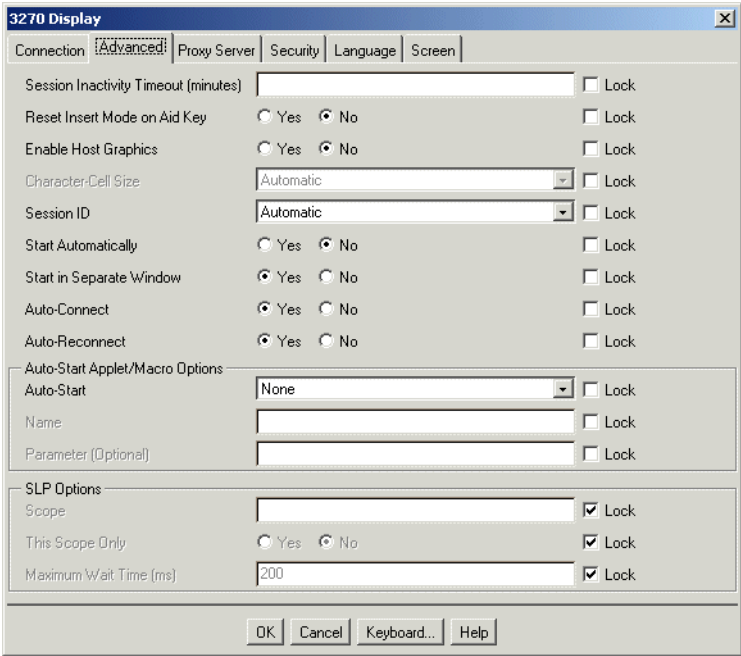


Figure 7-17 3270 session Advanced tab

The Advanced tab for a 5250 display session is shown in Figure 7-18.

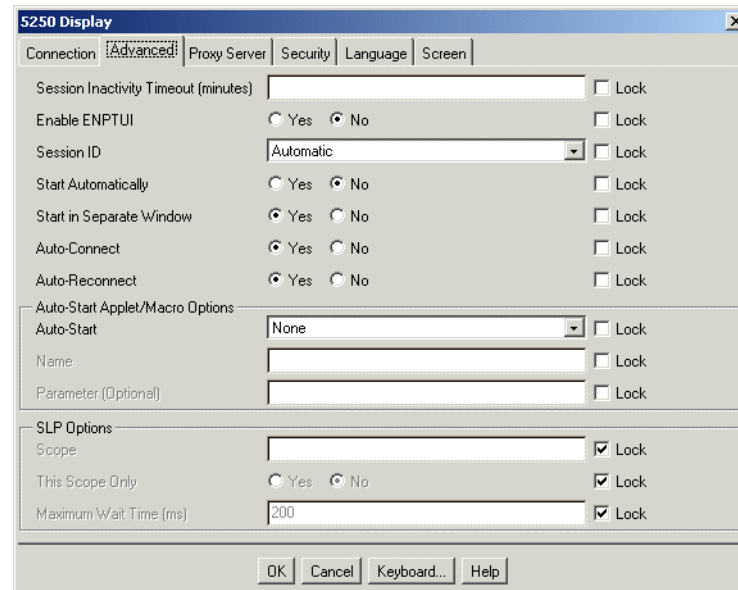


Figure 7-18 Advanced tab for 5250 display session

Use the following descriptions to complete the configuration:

► Session Inactivity Timeout (minutes)

Specifies the number of minutes that the Host On-Demand client will wait before terminating an inactive session connection. By default, the session connection is never timed out. The session will not automatically reconnect after timing out, even if Auto-Reconnect is set to true.

Terminating an idle connection may be useful for ensuring that resources such as LU names and workstation IDs are released when they are no longer being used. This option is available for 3270 or 5250 display/print sessions or VT sessions only

► Reset Insert Mode on Aid Key (3270 only)

If **Yes** is selected and you are in insert mode, then any aid key will turn the insert mode off. If enabled and you are not in insert mode, this function has no effect on the operation of the aid key. An aid key is an attention identifier key, any key that causes an interrupt to be sent to the host application, such as Enter, PF keys, or PA keys.

► Enable Host Graphics (3270 only)

Enables the 3270 host graphics function. If this is enabled, Character Cell Size will specify the cell characteristics used for host graphics. The default is No!

► Character-Cell Size

The cell characteristics used for host graphics.

Select Automatic if you use host applications that redraw graphical pictures according to a Usable-Area Query Reply or a Character-Set Query Reply sent by the terminal. This is the case when you use host applications under GDDM, and in some other graphical environments. Selecting Automatic causes the cell size returned by the Usable-Area Query Reply and the Character-Set Query Reply to change according to the Host On-Demand window size.

Select a fixed cell size if you use host applications that assume Terminal Usable Area to be a fixed coordinate space. Select the appropriate cell size for your application. For help selecting an appropriate cell size, refer to the character-cell-size table.

Automatic is the recommended selection. Choose a fixed cell size only if the application does not display graphics correctly for the automatically-chosen size

► Enable ENPTUI (5250 only)

Enables Enhanced Non-Programmable Terminal User Interface (ENPTUI) support. ENPTUI enables an enhanced user interface on nonprogrammable terminals (NPT) and programmable work stations (PWS) over the 5250 full-screen menu-driven interface. This option is not available if the session is enabled for Screen Customizer.

► Session ID

ID assigned to this session, A-Z. Sessions are started in alphabetical order. Automatic assigns the next available capital letter to the session. You may wish to assign a specific session ID if you have an HACL applet or EHLLAPI program that requires a specific session ID; otherwise, we recommend you use the Automatic setting.

► Start Automatically

If **Yes** is selected, the session is started and connected (if Auto-Connect is Yes) when the client is loaded.

► Start in Separate Window

Specifies whether the session is started in a separate browser window. If No, the session is started in the Client window with the session name and ID displayed on a tab. Each session started in the Client window is tabbed for easy access.

- ▶ Auto-Connect

Specifies whether the session should be automatically connected to the target Telnet server. If you set this to **No**, you must click Connect in the session menu every time you want to connect a session.
- ▶ Auto-Reconnect

Reconnects the session automatically if communication fails and later recovers.
- ▶ Auto-Start Applet/Macro Options
 - Auto-Start

Specifies whether an applet or macro should be run when the session is started.
 - Auto-Start name

Specifies the name of the applet or the macro to be run when the session starts.
 - Parameter (Optional)

The name of the parameter that is passed to the applet when the session starts. In order for an applet to receive parameters, the applet must implement the following method: `public void initParam(String param)`. The variable name, `param`, may be any valid variable name
- ▶ Host Code-Page (5250 printer session)

Specifies the table used to map EBCDIC codes from the host to appropriate ANSI graphics on the workstation. This option determines the list of available printers on the Printer tab. If you are using a DBCS language, you cannot select a host font. You must set it to the code page supported by the host system to which the session will connect.

The default is the code page that corresponds to the locale for which your workstation is configured.
- ▶ SLP Options

Service Location Protocol (SLP) enables a client to dynamically locate a TN3270 and TN5250 service, and to attach to the least-loaded server, making it unnecessary for you to know the destination address and port of any specific service. These fields control and manage the access by TCP/IP clients to servers that support SLP.

 - Scope

Controls and manages access by TCP/IP clients to servers that support SLP. Contact your administrator to get the correct value for this field.

If a server is not found within the specified scope, the session can be established only through an unscoped server; however, if This scope only is Yes, the session will connect only if a server with that scope is located.

The default is blank and this returns all scoped and unscoped services depending on the level of the Communications Server you are running.

We recommend that you configure all of your servers with scopes.

Special characters , (comma), / (forward slash), and : (colon) are not allowed in the Scope field.

- This Scope Only

Prevents a session from connecting to an unscoped server. If you set this to Yes and no server is found within the specified scope, the session will not connect.

The default is No.

- Maximum Wait Time (slp)

Sets the maximum time, in milliseconds, that the session waits to discover services or directory agents, or responses concerning load information. This value must be greater than zero and less than 3600000ms (1 hour).

The default is 200.

Directory Agent Discovery timeout and Service Agent Multicast timeout are both set with this value.

- ▶ Lock

Check Lock to prevent users from changing the associated startup value for a session. Users can not change values for most fields because the fields are unavailable. However, functions accessed from the session menu bar or tool bar can be changed.

- ▶ AS400 Options

Some specific parameters for a 5250 printer session are contained in the advanced tab for a 5250 printer session:

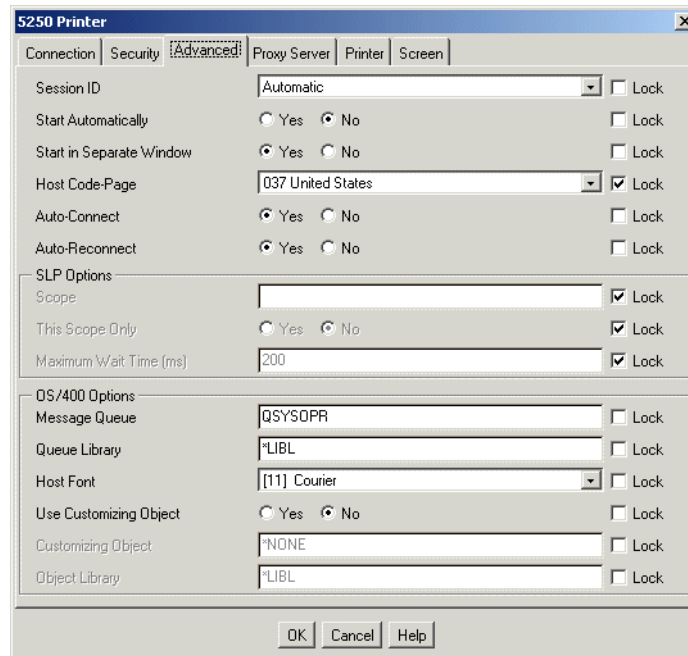


Figure 7-19 5250 printer advanced panel

- Message Queue

The name of the queue where operational messages for the printer device are sent.

If you specify the name of a display session here, messages relating to this printer device will be sent to that display session instead of to the default queue, QSYSQPR. A message queue exists for each display device and is assigned the same name as the device itself.

The first character must be A-through-Z, \$ (dollar sign), @ (commercial at sign), or # (number sign). The remaining characters can be A-through-Z, 0-through-9, \$, @, #, . (period), and _ (underscore).

- Queue Library

The name of the library where the printer message queue is located.

The first character must be A-through-Z, \$ (dollar sign), @ (commercial at sign), or # (number sign). The remaining characters can be A-through-Z, 0-through-9, \$, @, #, . (period), and _ (underscore).

- Host Font

The font used for a print file if a font is not specified by the application.

- Use Customizing Object

Click Yes if you want to use an object file to format print data instead of using the formatting provided by the application.

Customizing Object

The name of a user-defined iSeries file that can be used to format the data for this device.

The first character must be A-through-Z, \$ (dollar sign), @ (commercial at sign), or # (number sign). The remaining characters can be A-through-Z, 0-through-9, \$, @, #, . (period), and _ (underscore).

- Object Library

The name of the iSeries system library that contains the customizing object file.

The first character must be A-through-Z, \$ (dollar sign), @ (commercial at sign), or # (number sign). The remaining characters can be A-through-Z, 0-through-9, \$, @, #, . (period), and _ (underscore).

3270/5250 Proxy tab

The proxy tab was added to the HOD 7 session configuration window. The Proxy tab for 3270 and 5250 sessions are the same

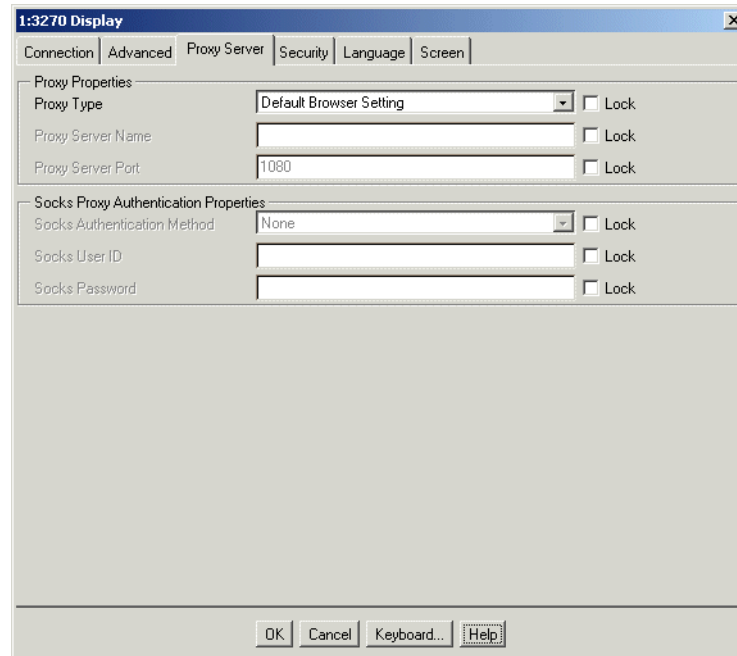


Figure 7-20 3270/5250 Proxy tab

- Proxy Properties
 - Proxy Type

A Host On-Demand session can connect to a host system through a proxy server. A proxy server enables applications to communicate transparently across a firewall. The proxy server connects to the host on behalf of the Host On-Demand session and relays data between the session and the host system.

This field allows you to specify what type of proxy server a host session uses. Select one of the following:

- Default Browser Setting

The session uses the proxy settings of the Web browser where the session runs.

The Java Virtual Machine (JVM) used in many Web browsers supports only Socks version 4 connections. If your proxy server supports Socks version 5 and you wish to use all of its features (such as authentication and enhanced IP address support), do not use the default browser settings. Instead, select Socks v5 as the Proxy Type.

- HTTP Proxy

The session only connects through an HTTP proxy server, overriding the proxy settings defined in the Web browser.

- Socks v4

The session only connects through a Socks version 4 proxy server, overriding the proxy settings defined in the Web browser. A Socks version 4 proxy servers connects to a host system on behalf of a Host On-Demand client and transmits data between the client and the host system.

- Socks v5

The session only connects through a Socks version 5 proxy server, overriding the proxy settings defined in the Web browser. Socks version 5 includes the complete functionality of Socks version 4; in addition, it supports authentication to the proxy server, IP version 6 addressing, domain names, and other networking features.

- Socks v4 if v5 unavailable

The session first attempts to connect using Socks version 5. However, if the proxy server does not support Socks version 5, the session connects using Socks version 4. In either case, the session overrides the proxy settings defined in the Web browser

The following fields are unavailable if Use Default Browser Setting is selected as the Proxy Type.

- ▶ Proxy properties

If the session is connecting to the host system through a Socks or HTTP proxy server, set the proxy server properties as follows:

- ▶ Proxy Server Name

Enter the hostname or IP address of the Socks or HTTP proxy server.

- ▶ Proxy Server Port

Enter the TCP port number of the Socks or HTTP proxy server.

The following fields are available only if Socks v5 is selected as the Proxy Type.

- Socks Authentication Method

Specify the authentication method that a Socks version 5 proxy server uses. Select one of the following:

- Clear Text

The user ID and password for authenticating to the Socks proxy server are sent as clear text over the connection.

- None

No authentication is used on the Socks proxy server

► Socks Proxy Authentication Properties

If you selected Clear Text as the Socks Authentication Method, enter the following properties:

- Socks User ID

Enter the user ID for authenticating to the Socks version 5 proxy server.

- Socks Password

Enter the password for authenticating to the Socks version 5 proxy server.

Note: If the Socks proxy server requires a user ID and password and you do not enter them here, the server will prompt you for them at connection time.

3270/5250 Security tab

The Security tab specifies the options required to establish secure sessions. The 3270 session has one additional parameter for telnet negotiation.

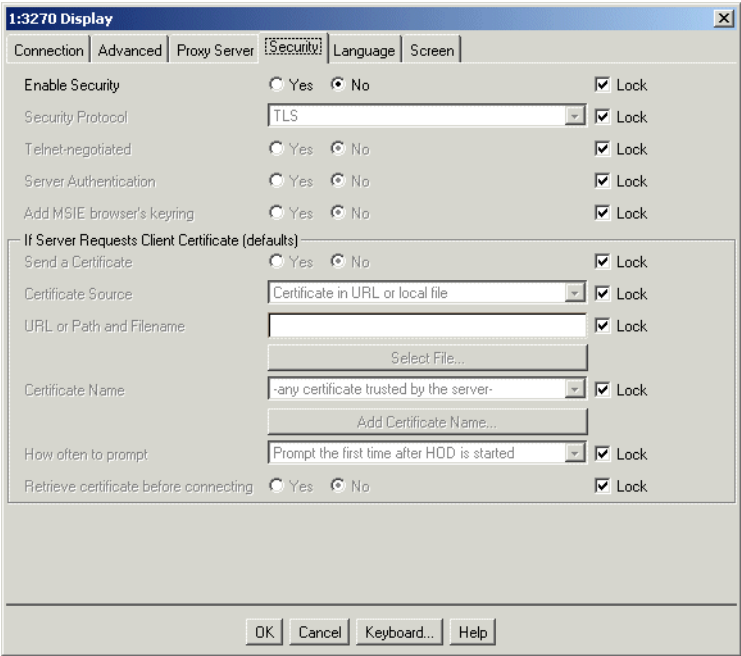


Figure 7-21 3270 session Security tab

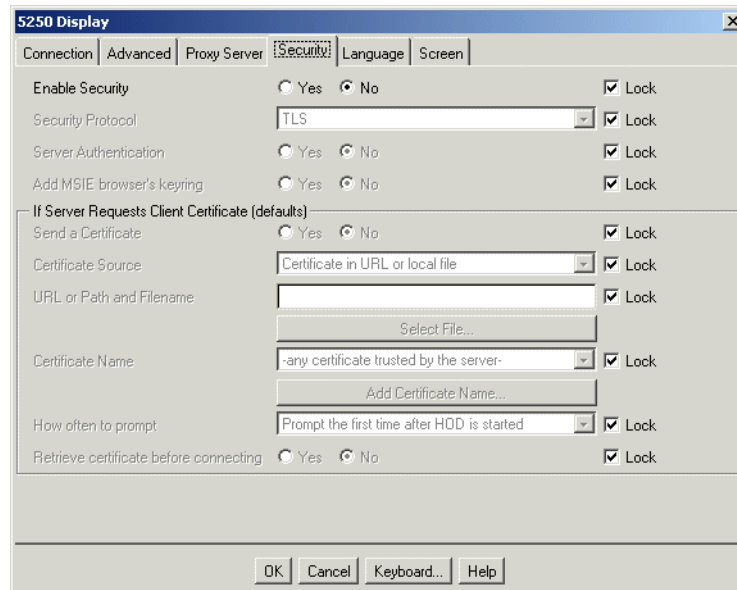


Figure 7-22 5250 session security tab

► **Enable Security**

Enables security between the workstation and either a telnet server that supports security or the Redirector

If Enable Security and Enable SLP are both set to **Yes**, a connection can be established only to servers that are enabled for security.

► **Security Protocol**

Select one of the following security protocols from the drop-down menu:

– **TLS**

Enables Transport Layer Security (TLS). TLS version 1.0 is the default security protocol for Host On-Demand clients. Note that TLS allows security negotiations from TLS version 1.0 to SSL version 3.0.

– **SSL**

Enables SSL version 3.0 security. Select this protocol only if the server cannot correctly negotiate a TLS connection.

► **Telnet-negotiated**

Determines if the security negotiations between the client and the Telnet server are done on the established Telnet connection or on a TLS connection prior to the Telnet negotiation. For the client to use this feature, the Telnet server must support TLS-based Telnet Security. The other options are valid regardless of whether Telnet-negotiated is set to Yes or No.

► Server Authentication

Ensures that a secure session is established only if the internet name of the server matches the common name in the server's certificate. This is effective only on a locally-installed client or a client downloaded via HTTPS

► Add MSIE Browser's Keyring

When this option is selected, the Host On-Demand client accepts Certificate Authorities trusted by the Microsoft Internet Explorer browser.

The following options are used to specify the handling of client authentication.

► Send a Certificate

Enables Client Authentication. If this option is turned off and the server requests a client certificate, the server will be told that no client certificate is available, and the user will not be prompted.

► Certificate Source

Once Send a Certificate is selected, this field is enabled. The client certificate can be kept in either a URL or local file or in the client's browser or a dedicated security device such as a smart card.

Alternatively, it can be kept in a local or network-accessed file, in PKCS12 or PFX format, protected by a password.

► URL or Path and Filename

Specifies the default location of the client certificate. The URL protocols you can use depend on the capabilities of your browser. Most browsers support HTTP, HTTPS, FTP, and FTPS.

► Select File

You optionally may click **Select File** and browse the file system available to the local system to locate the certificate.

► Certificate Name

This drop-down box is enabled if you indicate that the certificate is located in the browser or a security device. This drop-down box lists all personal certificates found in the Microsoft cryptographic database. Optionally, you may choose to accept “-any certificate trusted by the server-”.

► Add Certificate Name

This button invokes a dialog to specify the parameters for choosing a client certificate, including the common name, e-mail address, organizational unit, and organization used to define it. (This button is only available on the administrator's configuration panel.)

► How often to prompt

This drop-down box allows you to control the timing of prompts for client certificates. You can choose to prompt each time a connection is made to the server, or only the first time after starting HOD.

If your certificate is in a password-protected file and your client supports storing preferences locally, choosing **Prompt only once** causes HOD to prompt for the password the next time the connection is made, but never after that, unless the connection attempt fails.

If your certificate is accessed through the MSIE browser, **Prompt only once** can be chosen on any client, as well as **Do not prompt**, which will disable the prompt from HOD, but not from the browser or security device.

► .Retrieve certificate before connecting

If this button is turned on, the client will access its certificate before connecting the server, whether the server requests a certificate or not. If this button is turned off, the client will only access the certificate after the server has requested it; depending on other settings, this may force the client to abnormally terminate the connection to the server, prompt the user, and then re-connect.

3270/5250 Language tab

The following chapter describes the contents of the Language tab for 3270 and 5250 sessions. The fields on this page will not be enabled unless the code page of the system supports this page, such as Arabic, Hebrew, Thai, Chinese, Japanese or double-byte character set users.

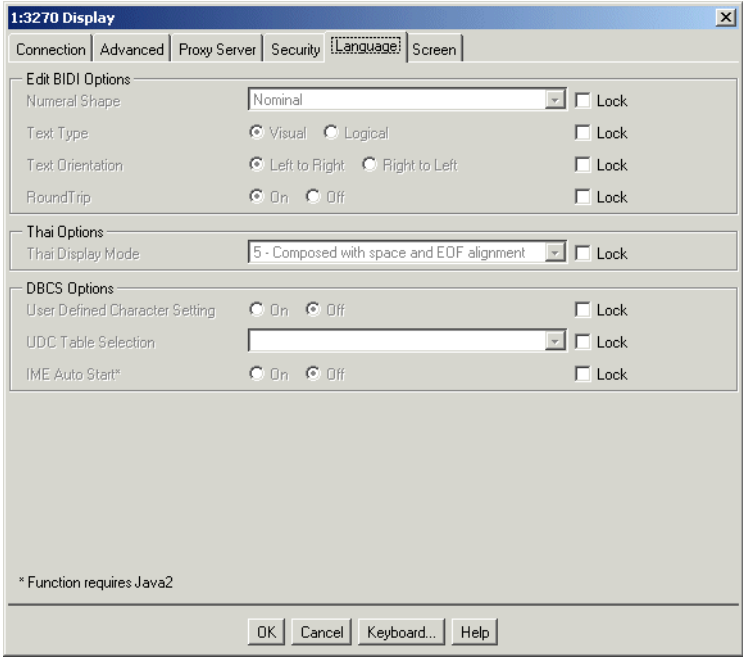


Figure 7-23 3270 Language tab

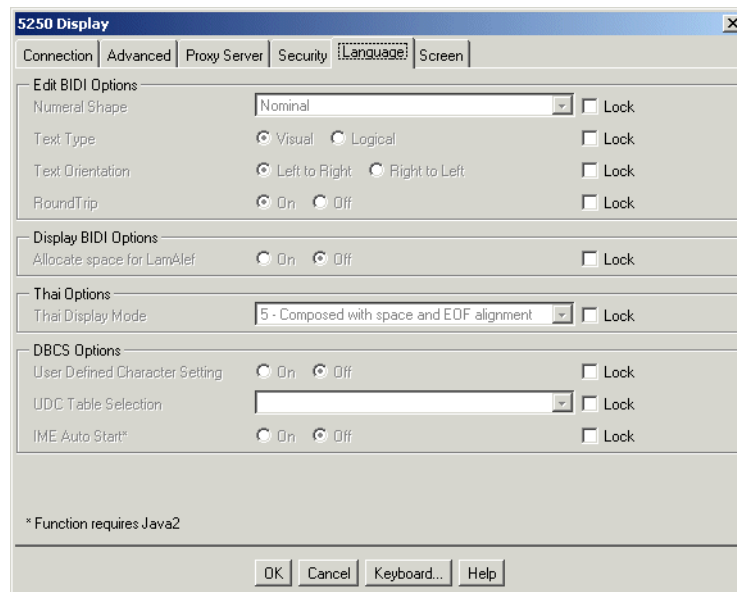


Figure 7-24 5250 Language tab

► Edit Bidi options

– Numeral Shape (Arabic only)

Determines the shape of numeric characters of the string copied to or pasted from the clipboard. Choose Nominal, National, or Contextual.

The default is Nominal.

– Text Type (Arabic and Hebrew only)

Determines the format of the text that is copied to or pasted from the clipboard. Select Visual or Logical.

The default is Visual.

– Text Orientation (Arabic and Hebrew only)

Determines whether the orientation of characters copied to or pasted from the clipboard is left to right or right to left.

The default is Left to Right.

– Round Trip (Arabic and Hebrew 5250/3270 only)

The Round Trip option disables the reversal of numerals if preceded by Bidi characters in the text copied to or pasted from the clipboard.

The default is On.

- ▶ Display Bidi options
 - Allocate space for LamAlef (Arabic 5250 only)

This option is to protect the LamAlef character at the Implicit file on iSeries systems. When this option is On, each LamAlef will allocate space at the end of the Arabic field.

The default is Off.
 - Numeral Shape (Arabic VT only)

Determines the shape of numeric characters on the screen. Select Nominal, National, or Contextual.

The default is Contextual.
 - Text Type (Hebrew VT only)

Determines the format of the text characters stored. Select Visual or Logical.

The default is Visual.
 - BIDI Mode (Arabic VT only)

Sets text display and cursor behavior to support VT display settings.

The default is On.
 - Cursor Direction (Hebrew Visual VT only)

Sets the cursor direction left-to-right (LTR) or right-to-left (RTL). When cursor direction is set RTL, all characters are displayed in the RTL direction because the cursor moves left by default after each displayed character. In general, only applications that are designed to receive input in a RTL direction will work properly when the Cursor Direction is set to RTL.

The default is LTR.
 - Smart Ordering (Arabic/Hebrew Logical VT only)

Determines whether segments of characters with different text attributes are ordered separately.

The default is Off.
 - Show Text Attributes (Arabic/Hebrew Logical VT only)

Enabled only when Smart Ordering is On.

The default is Yes.
 - Print RTL file (Arabic and Hebrew 3270 Printer session only)

Select Yes to print a file as it appears on a RTL screen. Print RTL file is available only for printing to Windows printers or Adobe PDF files. For more information, see Printing Right-to-left files.

The default is No.

- Thai options
 - Thai Display Mode

Select a display mode:

Table 7-1 Display Modes

Mode	Description
1 - Non-compose	No character composition occurs
2 - Composed	Thai characters are auto-composed. No column realignment is performed.
3 - Composed with space alignment	Three consecutive spaces cause column realignment. The realignment occurs whenever composing routine finds three consecutive spaces. If all fields have at least three trailing spaces, then all fields of all records will be properly aligned.
4 - Composed with EOF alignment	The EOF character (Hexadecimal 'EA') also causes column realignment. Whenever the composing routine finds a single EOF, it deletes the EOF and performs column realignment. If two consecutive EOFs are found, no realignment occurs, one EOF is deleted, and one EOF is treated as data.
5 - Composed with space and EOF alignment	Combines column realignment function of both mode 3 and mode 4.

- Lock

Check Lock to prevent users from changing the associated startup value for a session. Users can not change values for most fields because the fields are unavailable. However, functions accessed from the session menu bar or tool bar can be changed.

- ▶ DBCS options
 - User Defined Character Setting

Determine whether the session starts with a user-defined character (UDC) mapping table.

The default is Off. If you want to use a UDC mapping table, set this option to On.
 - UDC Table Selection

Select the UDC mapping table that will be applied in this session. You can create a new mapping table.

The default is none.
 - IME Auto Start

Select Yes to automatically start the Input Method Editor (IME) when the cursor is located on DBCS fields. IME is a front-end processor that generates DBCS strings. This function requires Java 2.

For DBCS sessions running on Java 2, the default is On. For all other sessions, the default is Off.

3270/5250 Screen tab

The parameters on this tab affect the visual appearance of the 3270 or 5250 display session. Both screen for 3270 and 5250 are the same. When using Java2 you will see 2 additional parameters the 3270 and 5250 screen tab for selecting fixed fonts and their font size.

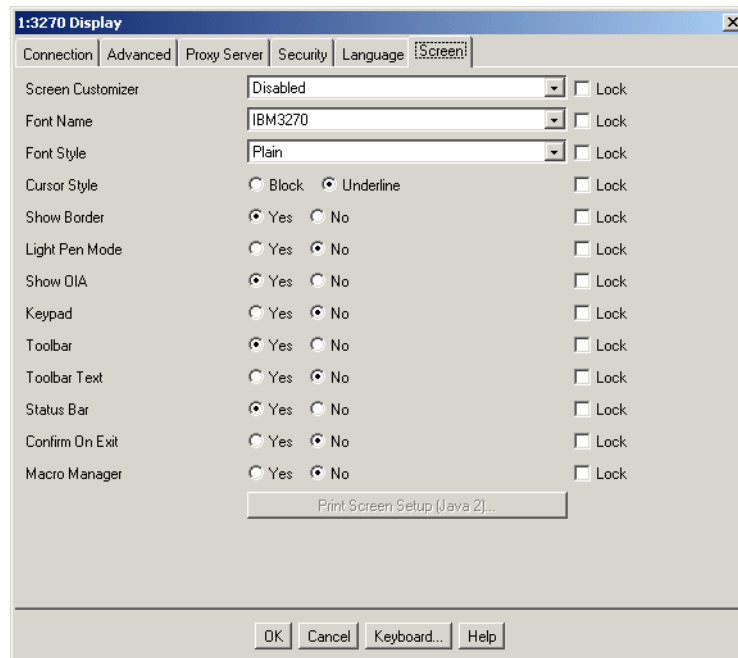


Figure 7-25 32705250 Screen tab under Java1

► Screen Customizer

Enables or disables Screen Customizer:

- Click Enabled or Client if you want to use a graphical interface.
- Click Administrator if you are using Screen Customizer Administrator and want to customize a graphical interface.
- Click Disabled if you do not want to use a graphical interface.

Two parameters Fixed Font and Fixed Font Size are only shown at Java2 administrators and will only apply to sessions at Java 2 clients (this functions depend on Java2). They have been added for enhanced accessibility. Using that a user can choose a rather large font which will keep its size and navigate with scroll bars in the host screen.

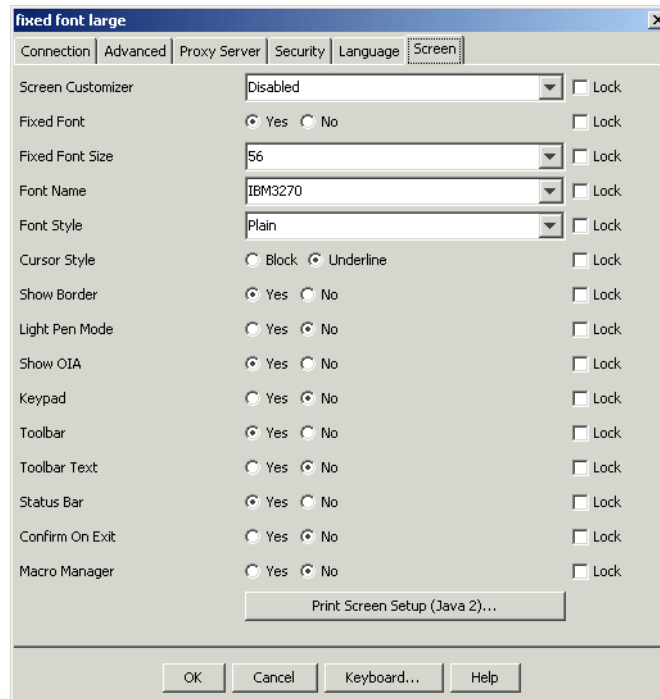


Figure 7-26 Fixed Font parameters in Screen window at a Java2 client

► Fixed Font (only onJava2)

When you select **Yes** the font size will not decrease or increase when you change the size of your emulator window. The font is fixed to its defined size. When the emulator screen does not fit in the browser window horizontal and vertical scroll bars will appear on the emulator window.

The default is No

Note: Fixed font parameters have only an impact to java2 clients. For java1 clients the emulator screen behaves always as if Fixed Font would be set to No

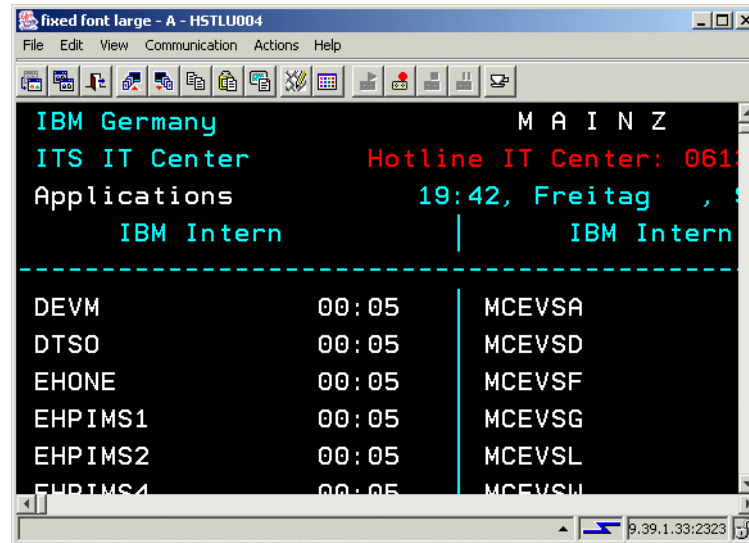


Figure 7-27 Fixed font emulator window on a Java2 client with scroll bars

- ▶ **Fixed Font Size (only on Java2)**
Select the desired font size from the drop down box
- ▶ **Font Name**
The type of font used to display characters on the screen. The choices are: Monospaced, Courier, IBM3270, ARB3270 (Arabic), HEB3270 (Hebrew) and THA3270 (Thai). The 3270 fonts are very similar to the sans-serif default font used by Personal Communications.
The default is Monospaced.
- ▶ **Font Style**
Choose Plain, Italic or Bold.
The default is Plain.
- ▶ **Cursor Style**
Determines whether the cursor displays as an underscore or a block while the keyboard is in Replace mode. You can also click Altcur on the session-window keypad to change the cursor style.

► Show Border

Determines whether a border around each protected and unprotected area is displayed. If Yes is selected the area between the spaces as used by the emulator window and the browser border is filled grey. If no is selected it is filled with the emulators background color (by default = black)

The default is Yes.

► Light Pen Mode

Choose to switch Light Pen Mode on or off when the session starts. Some host applications use a light pen as a pointer and operator. If you switch light pen mode on, you can use your mouse as a light pen.

The default is No (off).

► Show OIA

Determines whether the Operator Information Area (OIA) is visible on the screen.

The default is Yes (visible).

► Keypad

Determines whether the Keypad is visible on the screen. You can also turn this on or off from the View menu in the session screen.

The default is No (not visible).

► Toolbar

Determines whether the Toolbar is visible on the screen. You can also turn this on or off from the View menu in the session screen.

The default is Yes (visible).

► Toolbar Text

Determines whether the text that explains the purpose of each toolbar button is visible on the toolbar buttons. You can also turn this on or off from the View menu in the session screen.

The default is No (not visible).

► Status Bar

Determines whether the Status Bar is visible at the bottom of the screen when the session starts. The Status Bar displays connection status messages and toolbar button descriptions. You can also turn this on or off from the View menu in the session screen.

The default is Yes (visible).

► Confirm on Exit

Select Yes if you want a warning message to appear when a user attempts to close a session. If users select File > Exit, close a session window, exit from the toolbar, or right-click the left corner of the session window, a window appears asking if they really want to exit. If the user clicks OK, the session ends. If the user clicks Cancel or closes the window, the session remains open and unchanged. If the user closes the browser window, no exit warnings appear. If the user closes both a session and its associated printer session, the exit warning appears only once.

The default is No.

► Macro Manager

Determines whether the Macro Manager toolbar is visible on the screen. You can also turn this on or off from the View menu in the session screen

The default is No (not visible).

► Graphic Display (printer session only)

A printer session displays a window that includes several items of information, and shows the printer, workstation and host system as icons. If you turn Graphic Display off, the icons do not display but the other information remains; the window is therefore smaller.

► Show PA1 Key (3270 printer session only)

Choose whether to have a button on the screen for the Program Attention 1 key. The function of the key depends on the host application.

The default is No.

► Show PA2 Key (3270 printer session only)

Choose whether to have a button on the screen for the Program Attention 2 key. The function of the key depends on the host application.

The default is No.

► Lock

Check Lock to prevent users from changing the associated startup value for a session. Users can not change values for most fields because the fields are unavailable. However, functions accessed from the session menu bar or tool bar can be changed.

► Print Screen button

The window as shown in Figure 7-28 on page 319 allows to add headers and footers to a screen copy. The syntax for adding date, time etc. is the same as e.g. for a Windows Wordpad. This applies as well to the button Page Setup. For further detail refer to the online help of the window.

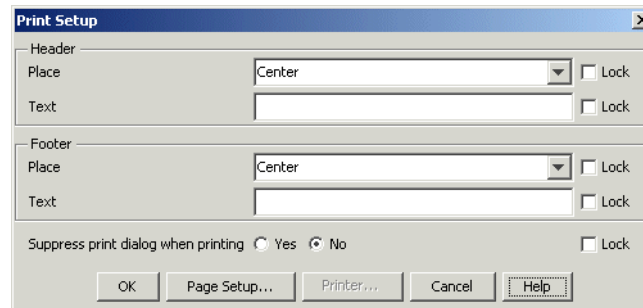


Figure 7-28 Print setup

VT Display session

Host On-Demand's VT support is fully compliant with accepted standards and includes features unique to the product.

To configure a VT session, click **VT Display**, which brings up the window shown in Figure 7-29.

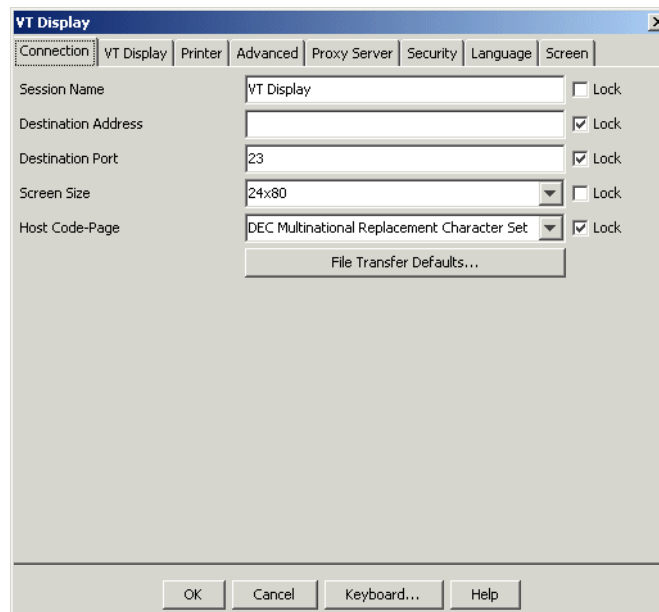


Figure 7-29 VT session Connection tab

VT Connection tab

The following fields have identical requirements as on the 3270 and 5250 settings. Refer to “3270/5250 Connection tab” on page 288 for details.

- ▶ Session Name
- ▶ Destination Address
- ▶ Destination Port
- ▶ Host Code-Page

The Screen Size parameter has many available screen sizes to choose from. You will probably find that most of them are very difficult to read; therefore, we recommend you use the 24x80 screen size unless your application requires a different size.

File transfer for VT host systems is performed with FTP; therefore, refer to “**.FTP session**” on page 331 for details on how to configure the file transfer defaults.

VT Display tab

VT displays have characteristics that are different from 3270 and 5250 sessions. See Figure 7-30.

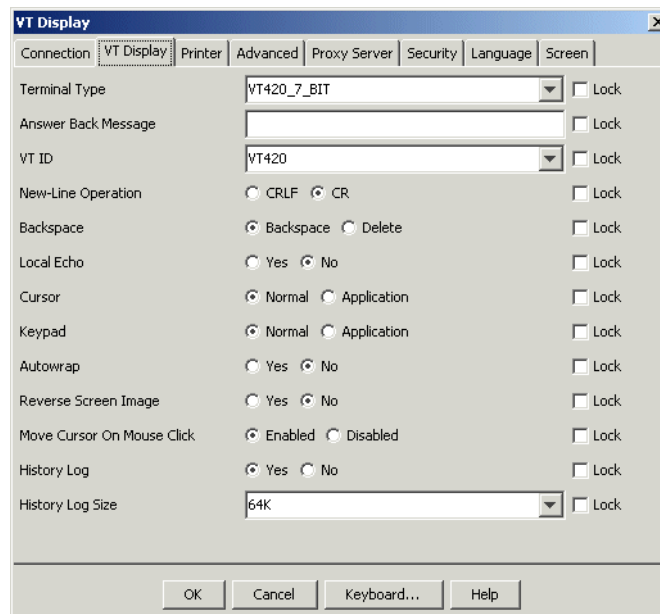


Figure 7-30 VT session display tab

- ▶ Terminal Type

Defines the type of VT emulation you want to use, which depends on the types supported by your host system. The types listed in the drop-down list box are:

- VT420 7-bit (default)
- VT420 8-bit
- VT100
- VT52

Note: If you require VT220 or VT320, specify VT420, which supports VT220 mode.

For VT emulation limitations see the online help for the display tab by clicking the Help button in the VT session display tab window.

► Answer Back Message

The Answer Back Message is used to return a message to the host when the host inquiry command is sent to the terminal. Enter into this field anything that you wish returned in response to a query command.

► New-Line Operation

Specifies where the cursor will move when you press the New Line key. The default is CR

- Click **CRLF** if the cursor must move to the left margin on the next line.
- Click **CR** if the cursor must move to the left margin on the same line.

► Backspace

Defines the default behavior of the backspace key; however, the actual action of the key is determined by the host application. The default is backspace.

- Click **Backspace** to send a standard ASCII backspace control-code (x'7F'). This moves the cursor backwards one position.
- Choose **Delete** to send a standard ASCII delete control-code (x'08'). This moves the cursor back one position and deletes the character in that position.

► Local Echo

Specifies where characters are sent when you type them.

- Click **Yes** to send characters to the host and to the display. Depending on how the host system behaves, you might get double characters on the screen.
- Click **No** to send characters to the host and depend on the host to send them back to the display.

The default is No, which means that characters will display only once.

- ▶ Cursor

Click **Normal** to use the arrow keys to move the cursor to different positions on the screen.

Click **Application** to use the cursor keys to send control-code sequences that can be read by host applications. To determine whether the Application option is required, refer to the application's documentation.

- ▶ Keypad

Click **Normal** to use the VT auxiliary keypad for typing numbers.

Click **Application** to use the VT keypad buttons to send control-code sequences that can be read by host applications. To determine whether the Application option is required, refer to the application's documentation.

- ▶ Autowrap

Specifies whether the text must automatically continue to a new line when it reaches the margin on the current line.

- ▶ Reverse Screen Image

Reverses the foreground and background colors.

- ▶ Move Cursor On Mouse Click

Specifies whether the cursor should move when you click the mouse on the screen.

- ▶ History log

The Host On-Demand history log can be turned on or off via the History Log radio button and its size is controlled by a drop-down control.

- ▶ History Log Size

The administrator can set the log size at increments between 16 KB and 512 KB. Once a VT session is configured to use the history log feature, the user will have access to the history log, which is highlighted in reverse video. You cannot use Host On-Demand's Copy function to get more than one screen's worth of data to the clipboard. By scrolling back and using the Copy Append function, it is possible to get as much of the history log as desired onto the clipboard before pasting it into another application. We found that the default of 64 KB for History Log Size was sufficient.

VT Printer tab

The screenshot shows the '1:VT Display' window with the 'Printer' tab selected. The window contains the following elements:

- Print to:** A dropdown menu set to 'Printer' with a 'Lock' checkbox.
- Print-to-File:** A section with 'Separate Files' set to 'Yes' (radio button selected) and 'No' (radio button unselected), with a 'Lock' checkbox.
- File Path and Name:** A text input field with a 'Lock' checkbox.
- Printer Definition Table:** A dropdown menu set to 'None' with a 'Lock' checkbox.
- Printer Name:** A text input field set to 'LPT1' with a 'Lock' checkbox.
- Buttons:** 'OK', 'Cancel', 'Keyboard...', and 'Help' at the bottom.

Figure 7-31 VT printer tab

A Printer tab, shown in Figure 7-31 on page 323, is actually a simplified version of the 3270 and 5250 printing setup. The Print Destination, Printer Name, Select Printer, File Path and Name and Separate Files options all work identically to their 3270 and 5250 counterparts. Refer to Chapter 19, “Host printing” on page 661 for a complete discussion of specifying printer information.

VT Advanced tab

The Advanced tab for the VT session, shown in Figure 7-32, is a simplified version of the 3270/5250 Advanced tab. Refer to “3270/5250 Advanced tab” on page 294 for details on how to complete these fields.

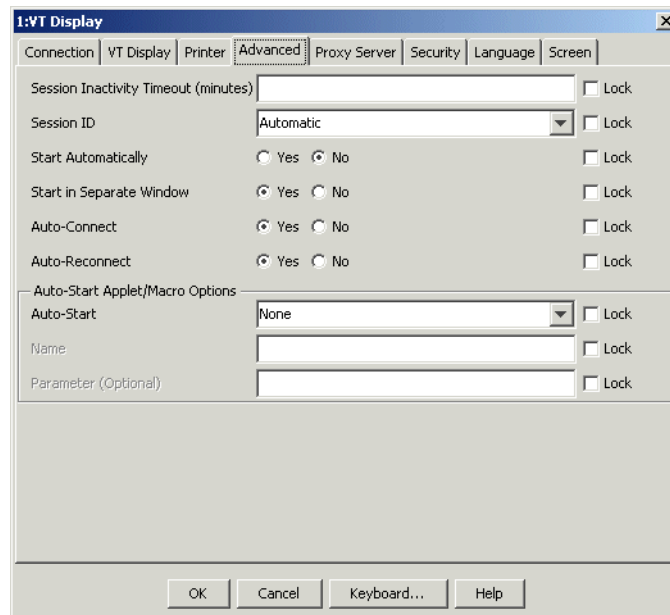


Figure 7-32 VT session Advanced tab

VT Proxy Server tab

The VT Proxy server tag is identical in content to the 3270 / 5250 Proxy server Tab. Refer to Figure 7-20 on page 302

VT Security tab

The VT Security tab is identical in content to the 3270/5250 Security tab. Refer to “3270/5250 Security tab” on page 304 for details on how to complete this tab.

VT Language tab

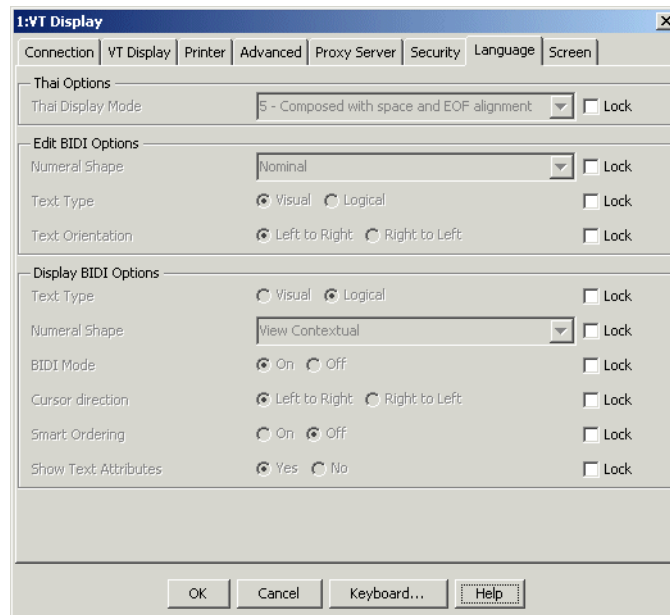


Figure 7-33 VT Display language tab

The Language tab is similar to the 3270/5250 Language tab. The fields on this page will not be enabled unless the code page of the system supports this page, such as Arabic, Hebrew, Thai, Chinese, Japanese or double-byte character set users. Refer to the online help for details in completing this tab (click Help button on this tab).

VT Screen tab

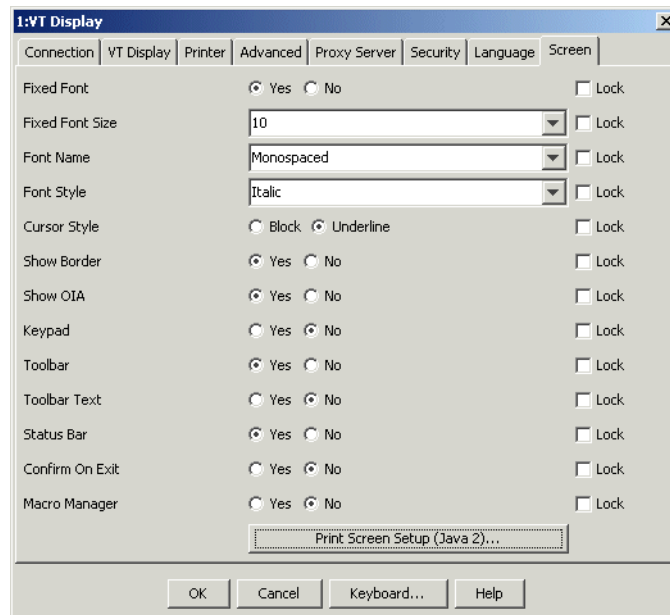


Figure 7-34 VT Display screen tab

The VT Screen tab, shown in Figure 7-34 on page 326, controls the appearance of the emulator window just as the 3270/5250 Screen tab does for 3270 and 5250 sessions. The VT session does not support Screen Customizer, so that option is missing. The Fonts field only supports Monospaced option. The light pen option is not available for VT screen. All other options are the same as the 3270/5250 session. This includes the dependency on Java2 for enabling the fixed font options. Refer to “3270/5250 Screen tab” on page 313 for details on completing this tab.

CICS Gateway session

The CICS Gateway client is a special client that communicates only to the CICS Gateway. It has limited functionality, such as it does not support SSL sessions. A more functional choice would be to use the 3270 emulator client.

The CICS Gateway client configuration, shown in Figure 7-35, consists of the following four tabs:

- ▶ CICS Connection tab
- ▶ CICS Advanced tab
- ▶ CICS Proxy tab
- ▶ CICS Screen tab

CICS Connection tab

The CICS Gateway client requires some unique configuration parameters:

► CICS Server

Specify the name of the CICS server to which the session connects. If this field is blank, the CICS Gateway's default server is used.

► CICS-Gateway Code-Page

Specify the code page in use by the operating system at the CICS Gateway. If this field is set to 000 Auto-detect, Host On-Demand normally retrieves the code page setting from the CICS Gateway; if that is not possible, you must get the correct code page from the network administrator.

The remaining parameters are the same as those specified for 3270/5250 sessions. Refer to “3270/5250 Connection tab” on page 288 for details.

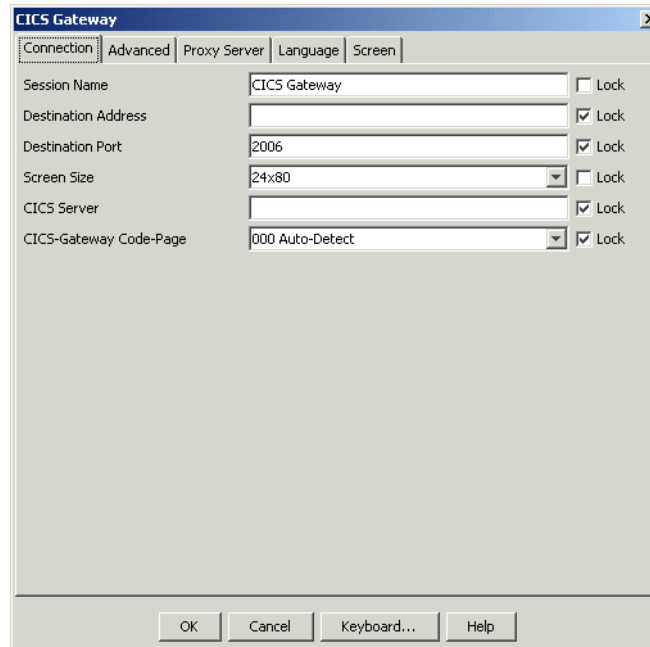


Figure 7-35 CICS session Connection tab

CICS Advanced tab

► Netname

The name of the terminal resource to be installed or reserved. If this field is blank, the selected terminal type is not predictable.

This option is available for CICS sessions only.

For the remaining parameters in the CICS Advanced tab, shown in Figure 7-36, refer to “3270/5250 Advanced tab” on page 294 for details on completing this tab.

The screenshot shows the 'CICS Gateway' dialog box with the 'Advanced' tab selected. The dialog has four tabs: 'Connection', 'Advanced', 'Proxy Server', and 'Screen'. The 'Advanced' tab contains the following settings:

- Reset Insert Mode on Aid Key:** Radio buttons for 'Yes' and 'No' (selected). A 'Lock' checkbox is to the right.
- Netname:** A text input field. A 'Lock' checkbox is to the right.
- Session ID:** A dropdown menu showing 'Automatic'. A 'Lock' checkbox is to the right.
- Start Automatically:** Radio buttons for 'Yes' and 'No' (selected). A 'Lock' checkbox is to the right.
- Start in Separate Window:** Radio buttons for 'Yes' and 'No' (selected). A 'Lock' checkbox is to the right.
- Auto-Connect:** Radio buttons for 'Yes' and 'No' (selected). A 'Lock' checkbox is to the right.
- Auto-Reconnect:** Radio buttons for 'Yes' and 'No' (selected). A 'Lock' checkbox is to the right.
- Auto-Start Applet/Macro Options:**
 - Auto-Start:** A dropdown menu showing 'None'. A 'Lock' checkbox is to the right.
 - Name:** A text input field. A 'Lock' checkbox is to the right.
 - Parameter (Optional):** A text input field. A 'Lock' checkbox is to the right.

At the bottom of the dialog are four buttons: 'OK', 'Cancel', 'Keyboard...', and 'Help'.

Figure 7-36 CICS session Advanced tab

CICIS Proxy tab

The CICS Proxy tab is similar to the proxy tab of the 3270 / 5250 session. So please refer to Figure 7-20 on page 302 for completing this tab.

The screenshot shows the 'CICS Gateway' window with the 'Proxy Server' tab selected. The window has a title bar with a close button. Below the title bar are five tabs: 'Connection', 'Advanced', 'Proxy Server' (selected), 'Language', and 'Screen'. The 'Proxy Properties' section contains three fields: 'Proxy Type' (a dropdown menu showing 'Default Browser Setting'), 'Proxy Server Name' (a text box), and 'Proxy Server Port' (a text box containing '1080'). Each field has a 'Lock' checkbox to its right. The 'Socks Proxy Authentication Properties' section contains three fields: 'Socks Authentication Method' (a dropdown menu showing 'None'), 'Socks User ID' (a text box), and 'Socks Password' (a text box). Each field also has a 'Lock' checkbox to its right. At the bottom of the window are four buttons: 'OK', 'Cancel', 'Keyboard...', and 'Help'.

Figure 7-37 CICS Proxy tab

CICS Language tab

The Language tab has only one pane to fill in. It will not be enabled unless the code page of the system supports this page. Refer to the online help for details in completing this tab.

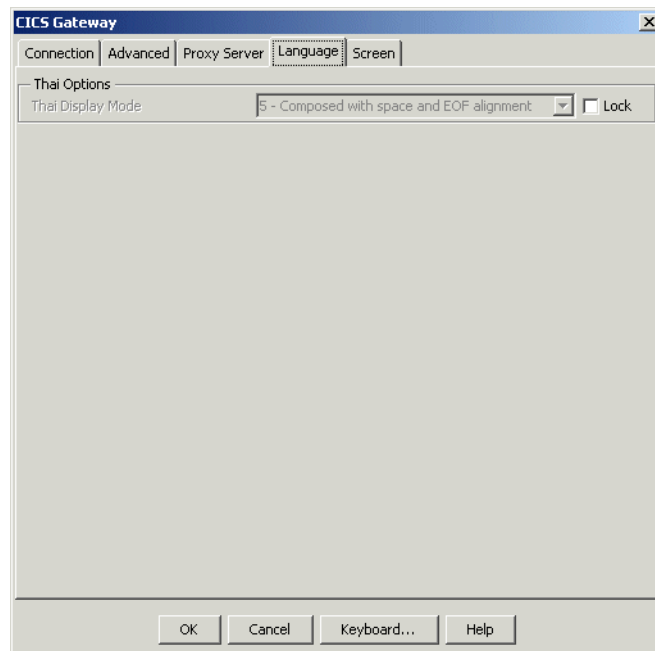


Figure 7-38 CICS session language tab

CICS Screen tab

The CICS Screen tab defines how the emulator window will appear to the end user. All configuration options are the same as for the 3270 session; therefore, refer to “3270/5250 Screen tab” on page 313 for details on how to complete this tab.

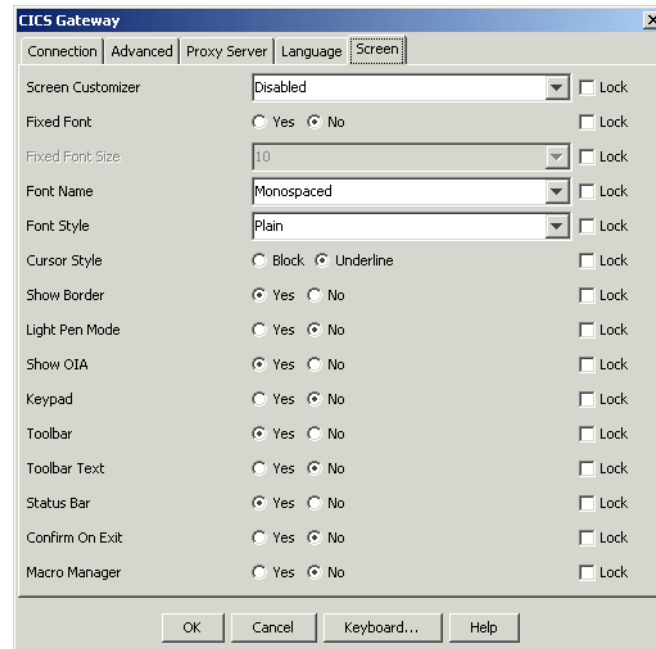


Figure 7-39 CICS session Screen tab

.FTP session

The FTP session lets you copy files and directories from or to another system that supports FTP.

FTP Connection tab

This tab, shown in Figure 7-40, specifies all connection information for this session.

The screenshot shows a dialog box titled "FTP" with a tabbed interface. The "Connection" tab is selected. The fields and their values are as follows:

Field	Value
Session Name	FTP
Destination Address	ftp.ibm.de
Destination Port	21
Anonymous Login	<input checked="" type="radio"/> Yes <input type="radio"/> No
E-mail Address	Testuser@ibm
User ID	zorn
Password	*****
Account	
Local Home Directory	C:\
Remote Home Directory	
Load Initial Remote Directory	<input checked="" type="radio"/> Yes <input type="radio"/> No

At the bottom of the dialog are three buttons: "OK", "Cancel", and "Help".

Figure 7-40 FTP session Connection tab

Use the following information to complete this window.

- ▶ Session Name, Destination Address, Destination Port
These parameters are the same as those for 3270/5250 sessions, so refer to "3270 and 5250 display sessions" on page 286.
The default port is 21 for FTP
- ▶ Anonymous Login
Enables the session to log in to an FTP server using anonymous as the user ID.
- ▶ E-mail Address
This field is enabled only when Anonymous Login is selected, and specifies the e-mail address to use as the password when connecting to the FTP server. If this field is blank, the session also sends anonymous as the password to the FTP server.
- ▶ User ID

Specifies the user ID the session uses when connecting to the FTP server. Anonymous Login must be disabled. If User ID is blank you will be prompted for a user ID and password when the session attempts to connect to the FTP server.

► Password

Specifies the password the session uses when connecting to the FTP server. If Password is blank, you are prompted for a password when the session connects to the FTP server. The password is not encrypted when it is sent to the FTP server. If Anonymous Login is set to Yes the e-mail address is sent to the FTP server as the password. If E-mail Address is blank and Anonymous Login is set to Yes, anonymous is also sent to the FTP server as the password.

► Account

Specifies the FTP server account name the session uses when connecting to the FTP server. Not all FTP servers use accounts. Check with the FTP server Administrator if you need an account.

► Local Home Directory

Sets the initial directory on your PC when the session connects to the FTP server. After the session connects, you can change directory by entering a valid directory in the Directory field, by clicking the directory button next to the Working Directory field and by double clicking on a directory folder in the local file list.

The default is c:\.

► Remote Home Directory

Sets the initial directory on the FTP when the session connects to the FTP server. After the session connects, you can change directory by entering a valid directory in the Directory field, by clicking the directory button next to the Working Directory field and by double clicking on a directory folder in the remote file list.

The default is the initial login directory set by the FTP server at login. If you enter a valid directory in this field it will over-ride the login directory set by the server.

► Load Initial Remote Directory

In some FTP sites a subdirectory is that large that it might take a considerable amount of time until the FTP session can display it. To avoid the initial display of such subdirectories select No. If you select No, the user must do one of the following for the host directory listing to appear:

- Type a path in the Directory field
- Refresh the remote listing

- Upload a file or directory
- The default is Yes, which shows the content of the remote subdirectory at session connect.

Figure 7-41 on page 334 shows the example of an established FTP session using the settings as filled in according to Figure 7-40 on page 332.

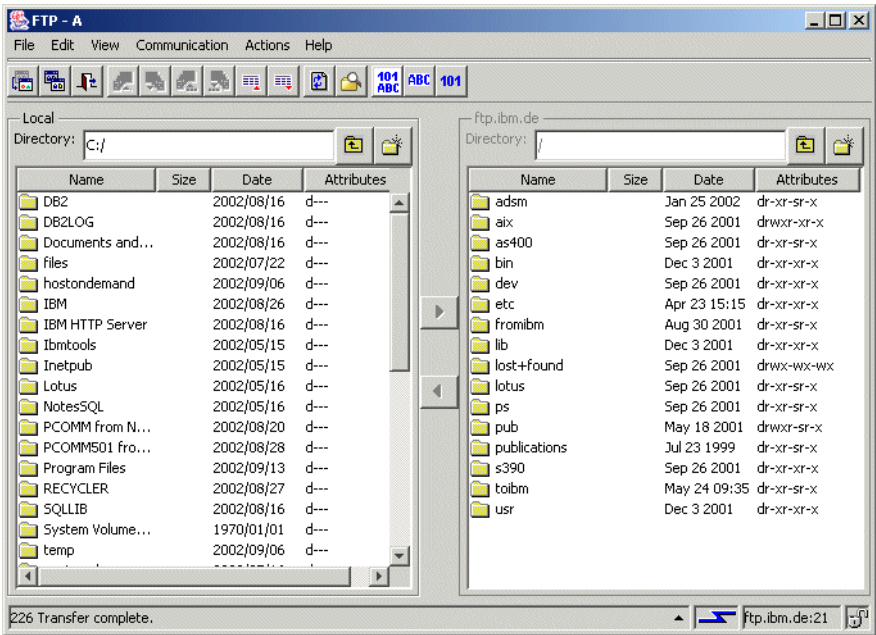


Figure 7-41 Example of established FTP connection

FTP Advanced tab

This tab controls file transfer parameters.

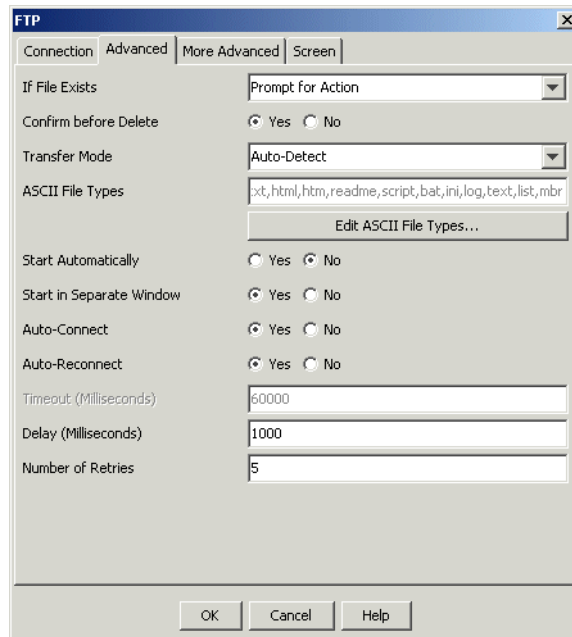


Figure 7-42 FTP session Advanced tab

Use the following information to complete this tab.

► If File Exists

Determines what action the session takes if the file already exists. Choices include Overwrite Existing, Prompt for Action, and Skip the File. When set to Prompt for Action, you can choose to Overwrite, Save As, Skip the File or Cancel.

The Transfer List Manager in the FTP client ignores this value.

The default is Prompt for Action.

► Confirm before Delete

Prompts for confirmation before a file is deleted.

The default is Yes.

► Transfer Mode

Sets the default file transfer mode. Valid values include ASCII mode, Binary mode and Auto-detect. ASCII files are typically plain text files, while binary files can be executable, graphics or a proprietary format (for example, database .dbf and MS Word .doc files). Auto-detect automatically selects the proper file

transfer mode for each file based on the file's extension. Files with an extension listed in the ASCII File Types list are transferred as ASCII files. Files with an extension not listed in the ASCII File Types list are transferred as binary files.

The default is Auto-detect

► ASCII File Types

Determines which files are transferred in ASCII mode instead of binary mode when Default Transfer Mode is set to Auto-detect. Files with an extension listed in this list are transferred as ASCII files. Files with an extension not listed in this list are transferred as binary files.

► Edit ASCII File Types

Allows you to add and remove entries in the list of ASCII File Types. To add an ASCII File Type, click the Edit ASCII File Types button. Enter the text to associate with ASCII mode in the dialog box that displays, and then click Add. To Add multiple entries, type each one into the dialog box, separated by commas (","), and click Add. To remove an ASCII File Type, click the Edit ASCII File Types button. Select the File Type from the list in the dialog box, and click Remove. To remove multiple entries, select entries by clicking on them and click Remove. Click OK when you are finished editing ASCII File Types.

► Start Automatically

Starts and connects the session (if Auto-Connect is Yes) when the client is loaded.

The default is No.

► Start in Separate Window

If Yes is selected, the session is started in a separate browser window. If No, the session is started in the Client window with the session name and ID displayed on a tab. Each session started in the Client window is tabbed for easy access.

The default is Yes.

► Auto-Connect

Automatically connects the session to the target ftp server. If you set this to No, you must click Connect in the session menu every time you want to connect a session.

The default is Yes.

► Auto-Reconnect

When set to Yes, automatically re-connects the session to the target ftp server if the connection fails. If you set this to No, you must click Connect in the session menu to re-connect the session.

The default is Yes.

► Timeout (milliseconds)

Sets the FTP connection timeout in milliseconds. To prevent the connection from ever timing out set this value to 0.

The default is 60000.

► Delay (milliseconds)

Sets the delay, in milliseconds, between connection retry attempts. Auto-reconnect must be set to Yes.

The default is 1000.

► Number of Retries

Sets the maximum number of connection attempts. A value of 0 will cause the session to try connect to the FTP server until a connection is made or you cancel the attempt. Auto-reconnect must be set to Yes.

The default is 5.

► Lock (Host On-Demand Administrator only)

Check Lock to prevent users from changing the associated startup value for a session. Users can not change values for most fields because the fields are unavailable. However, functions accessed from the session menu bar or tool bar can be changed.

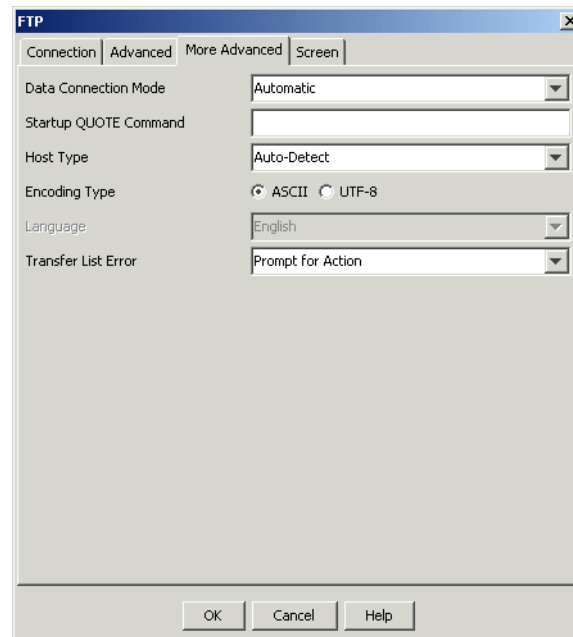
FTP More Advanced tab

Figure 7-43 FTP session More Advanced tab

This tab, shown in Figure 7-43, allows you to specify:

► Data Connection Mode

If you select Automatic, the FTP client attempts active mode transfer. If active mode transfer fails, the FTP client attempts passive (PASV) mode transfer. If PASV mode transfer works, the current session remains in PASV mode.

If you select Active, the FTP client always uses active/normal mode for data connection.

If you select Passive, the FTP client always uses passive mode for data connection.

The default is Automatic.

— Notes:

- By default, the FTP client checks the socks settings in your browser's proxy server settings to automatically enable PASV mode. This option is ignored when your browser's proxy setting is enabled.
- When the FTP client is within a firewall and it detects that the FTP server is outside the firewall, passive (PASV) mode is automatically activated, if your browser's socks setting is enabled.

- If you have a socksified client, you must enable this option, if the FTP server is outside the firewall.
 - If the FTP client is having trouble connecting to the FTP server, you may need to turn off the browser's proxy settings; if you have enabled this option, disable it.
 - If the FTP client session appears to hang after a file upload or download when PASV mode is enabled, increasing the Timeout value on the Advanced tab from 60000 milliseconds to 120000 milliseconds might resolve the problem.
- **Startup Quote Command**
- Sends an uninterpreted string of data to the FTP server as the session starts. The Startup Quote Command allows you to set FTP server supported options when starting the session. For example, to set the host translation table, type "site trans [translation table name]" in the dialog box.
- **Host Type**
- Defines the FTP server's directory/file format style. Valid values are Auto-Detect, MS-DOS, MVS, Novell, OpenVMS, OS/2, OS/400, OS/400-Unix, Unix, and VM.
- The default is Auto-Detect. You should only change this value if you are having trouble seeing the remote system's file list (for example, when the remote file list panel displays only file folders and no other data, or the format of the data is incorrect).
- To list the contents of MVS datasets and HFS directories without changing the Host Type or defining two separate FTP sessions, select Auto-Detect for the Host Type. You can list the contents of either PDS datasets or HFS directories using the working Directory field. For example, the path user.linklib lists the members of the user.linklib dataset, and the path /usr lists the contents of the usr HFS directory.
- When you select Auto-Detect, the FTP client uses SYST or PWD response string to determine the server directory style, and to parse the data in one of the Host Type formats. For default server OS and Host Type mapping refer to online help of this window.
- **Encoding Type**
- If you select UTF-8, the FTP client converts file names and path names to UTF-8 before sending them to the server and converts the file names and path names to the local client encoding when receiving the files from the server. For this to happen, the FTP server must support UTF-8 encoded path names. The default is ASCII when there is no conversion.
- **Language**

Enabled if you select UTF-8 encoding type. Select a language for FTP greetings and error messages. The default is the language of the Host On-Demand client. If the FTP server does not support the language you select or the default Host On-Demand client language, then the greetings and error messages appear in English.

► **Transfer List Error**

Allows you to choose what happens if an error occurs during the upload or download of a transfer list. If you select Prompt for Action, a window appears allowing you to continue or cancel the transfer. If you select Continue, the transfer of the remaining files or directories in the list continues. When the transfer is complete, a window appears with the number of errors during the transfer. Click Show Errors to view the errors.

► **Force BIDI reordering (BIDI only)**

Enabled only if you select UTF-8 as the Encoding Type and Arabic or Hebrew as the Language. If file names in the FTP client appear in reverse, select Yes to reorder them. For more information, see Bidirectional support for FTP client in UTF-8 mode.

► **Lock (Host On-Demand Administrator only)**

Check Lock to prevent users from changing the associated startup value for a session. Users can not change values for most fields because the fields are unavailable. However, functions accessed from the session menu bar or tool bar can be changed.

FTP Screen tab

The Screen tab shown in Figure 7-44 specifies the layout of the interactive FTP client window.

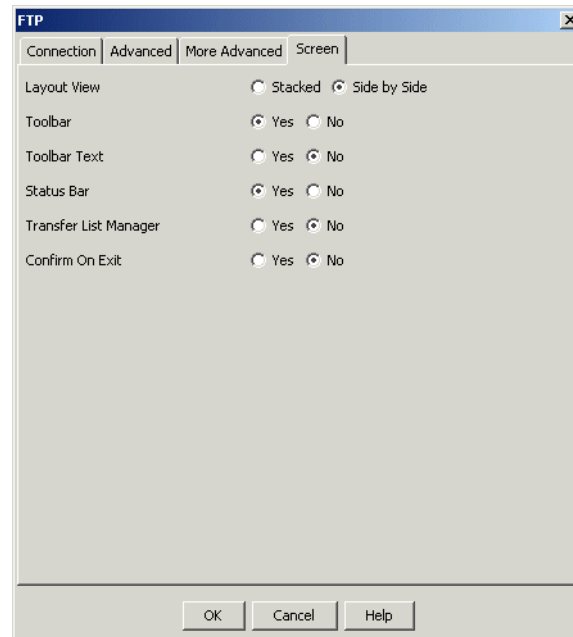


Figure 7-44 FTP session Screen tab

► Layout View

Determines which of two graphical views your FTP session has at startup. You can change the Layout View on the session menu under View. Both views list file name, size, date and attributes. Valid values are:

– Side by Side View

Provides a view of the remote and local file systems in two separate window panels adjacent to each other, with the remote file system on the left and the local file system on the right. When the FTP client window is resized, the file lists are automatically resized.

– Stacked View

Provides a view of the remote and local file systems in two separate panels on top of each other, with the local file system on top of the remote file system. When the FTP client window is resized, the file lists are automatically resized.

The default view is Side by Side.

► Toolbar

Determines whether the Toolbar is visible on the screen. You can also turn this on or off from the View menu in the session screen.

The default is Yes (visible).

► **Toolbar Text**

Determines whether the text that explains the purpose of each toolbar button is visible on the toolbar buttons. You can also turn this on or off from the View menu in the session screen.

The default is No (not visible).

► **Status Bar**

Determines whether the status bar is visible at the bottom of the screen when the session starts. The Status Bar displays connection status messages and toolbar button descriptions. You can also turn this on or off from the View menu in the session screen. The status bar displays all FTP commands and responses.

The default is Yes (visible).

► **Transfer List Manager**

Select Yes if you want the Transfer List Manager toolbar to be visible on the screen when the session starts. You can also display or hide the Transfer List Manager toolbar from the View menu.

The Transfer List Manager allows you to create and transfer a list of files or directories.

The default is No (not visible).

► **Confirm on Exit**

Select Yes if you want a warning message to appear when a user attempts to close a session. If users select File > Exit, close a session window, exit from the toolbar, or right-click the left corner of the session window, a window appears asking if they really want to exit. If the user clicks OK, the session ends. If the user clicks Cancel or closes the window, the session remains open and unchanged. If the user closes the browser window, no exit warnings appear.

The default is No.

► **Lock (Host On-Demand Administrator only)**

Select Lock to prevent users from changing the associated startup value for a session. Users can not change values for most fields because the fields are unavailable. However, functions accessed from the session menu bar or toolbar can be changed.

Multiple sessions

The multiple session object is a rather simple concept. A multiple session object is one that represents multiple Host On-Demand host sessions. When the user opens one of these objects, all of the sessions it represents are started. The configuration window is shown in Figure 7-45 on page 343.

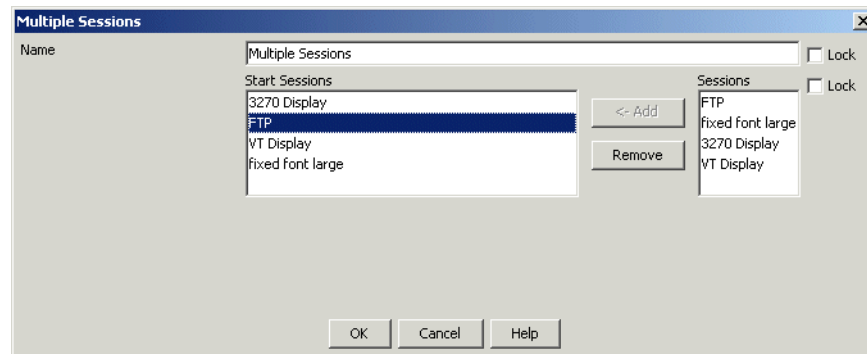


Figure 7-45 Multiple session configuration window

You must enter a unique name to identify the object. Next add sessions by highlighting a session from the right pane and clicking **Add**, repeating as necessary to place all desired sessions in the left pane. You may remove an unwanted session by selecting it from the left pane and clicking the **Delete** button. When completed you will have a session window similar to that shown in Figure 7-46 on page 343.

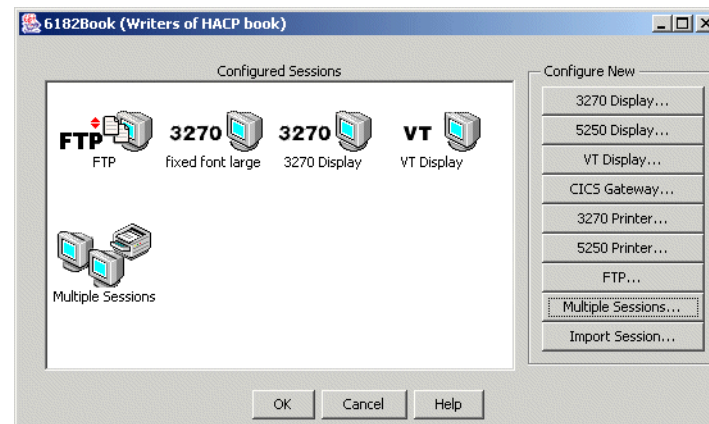


Figure 7-46 Configured Sessions window

When creating or modifying a multiple session object, keep in mind:

- ▶ To add more than one host session at a time, simply select additional sessions by using the Shift or Ctrl keys on the keyboard when selecting a host session. Then, when the desired sessions are selected, click the **Add** button.
- ▶ You can add the same session multiple times. When the object is opened, each of the sessions will open. So if you add the same host session twice, two emulator sessions to that host will open when the user opens that multiple session object.

Multiple session object behavior

This section is a summary of what to expect when using the multiple session object.

- ▶ A multiple session object can contain any Host On-Demand session object except another multiple session object.
- ▶ To add a Host On-Demand session to a multiple session object, the session must have been previously defined for that group or user.
- ▶ Launching a multiple session object will open all the sessions contained within.
- ▶ If a multiple session object is deleted, it does not delete the session objects it contains.
- ▶ If an administrator deletes a host session object that is contained within a multiple session object, a warning is displayed but the pointer to the object is not removed. The administrator must delete it separately.
- ▶ Multiple session objects can be exported and imported. However, when such an object is exported, it does not export the host session objects that are contained within. Those must be exported and imported separately.

Import Sessions

The Import Sessions window is used to import exported Host On-Demand (.hod) sessions or sessions from Personal Communications (.ws).

Click **Import Session** to bring up the window shown in Figure 7-47 on page 344.

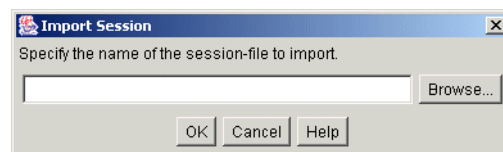


Figure 7-47 *Import Session window*

Simply type the name of the file or use the Browse button to locate the file on any file system accessible by the user. When you are finished, click **OK**.

You can import an existing session to create a new one. The existing session can be either a telnet session from Personal Communications v4.1 or later or a previously-exported Host On-Demand session.

If you are importing a session icon configured for multiple sessions, you must import all of the sessions contained within that configuration. For example, if the multiple session is configured to start a 3270 session, a 5250 session and a 3270 print session, then you need to import all 3 of those sessions along with the multiple session icon.

To import a session:

1. From the Client window, click Add Sessions, then Import.
2. Type the filename for the session you want to import, or click Browse. You can import:
3. Session files previously created by exporting Host On-Demand sessions
4. Telnet sessions from Personal Communications v4.1 or later
5. Click OK.
6. The session icon appears in the Configured Sessions area.

The sessions you import from other products (for example, Personal Communications) may not behave exactly as they did in the originating product. Features such as screen colors and key mappings may not be correct.

7.1.8 Disabling functions

When configuring users and groups, you can disable functions that you do not want users to access. You can disable any of the graphical interface items on pop-up menus and buttons in the Client window, the session menu, and the session toolbar. For example, you can remove items such as Copy, Export Session, or Properties from the pop-up menu in the client window or the macro button from the toolbar in the session window. When a function is disabled, it is removed from the toolbar or menus so users do not see it. Functions cannot be accessed using the shortcut keys either.

See Figure 7-48 on page 346 which shows the disable functions for the desktop. Use the left pane to navigate to the available group of options and select the behavior for the individual parameters on the right pane.

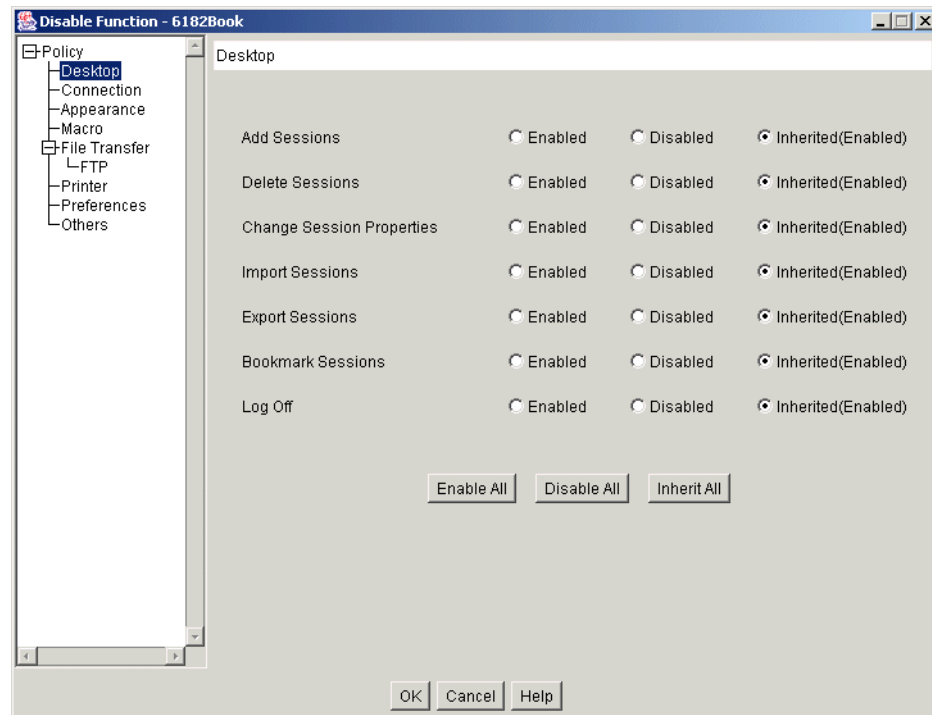


Figure 7-48 Disable function

You cannot reduce download size by disabling functions. Disabling functions is different from locking functions. You can lock the fields of a function when you are configuring a session. Locking fields locks the startup values for a session. In most cases, users cannot change values for those fields because the fields are unavailable. However, functions accessed from the session menu bar or tool bar can be changed.

You can enable or disable functions for a group or user in either one of two ways:

- ▶ From the Administration window:
 - a. Click Users/Groups.
 - b. Right-click a group or user and select Disable Function.
- ▶ From the Deployment Wizard:
 - a. Define a host session.
 - b. Click Disable Functions.

Changes made to a user's functions while the user is logged on do not become effective until the user has logged off and then logs back on.

Inheriting from a group (Administration client only)

Enabling and disabling functions can be set for each group and for each user within a group. A user or group can be configured to use, or inherit, the settings for functions from a higher (parent) group. However, settings for functions at the lower level groups or users take precedence over the higher level groups. In other words, if you disable a function at a group level but enable it for a user in that group, the function is enabled for only that user. Because of the inheritance factor, it is easier to set functions at the group level, and then disable or enable specific functions for the users or groups belonging to those higher level groups.

If you are using an LDAP server for storing configuration information, a user can be a member of only one group. If you select Inherit for a function, whatever is set for that group is applied to the user.

If you are using the Host On-Demand configuration server, a user can be a member of multiple groups. If you select Inherit, and all the groups to which the user is a member of have the function disabled, then the function is disabled for the user also. Or the other way around: If one of the groups to which the user belongs has a function enabled, the user inherits this function and is enabled for him.

See the online help for each group of options for detailed description of the individual parameters. In general you can select for each parameter:

- ▶ **Enabled**
Allows group or user to access this function.
- ▶ **Enable All**
Enables all the functions listed on the current screen.
- ▶ **Disabled**
Does not allow group or user to access this function.
- ▶ **Disable All**
Disables all the functions listed on the current screen.
- ▶ **Inherited (Enabled) (Administration client only)**
Uses the setting to which this user or group is a member of. The inherited value is enabled.
- ▶ **Inherited (Disabled) (Administration client only)**
Uses the setting to which this user or group is a member of. The inherited value is disabled.

- ▶ Inherit All (Administration client only)
Inherits all the functions listed on the current screen.

Note: If a function that is disabled is represented by a pull-down menu and an icon on the toolbar, both are hidden from the user.

7.2 Services

The Services window in Figure 7-49 will be presented after you click the **Services** task in the Administration Notebook. Use the Host On-Demand Services window to manage Host On-Demand services. This window shows the status of each service, the status of the trace option for each service, and lets you view the server's message and trace log. You can also refresh the view to see the current status of each service.

The following services are available:

- ▶ The Redirector gives clients access to telnet hosts other than the Web server from which they were downloaded. The Redirector supports TLS security. The Service status (started or stopped) and the trace status (started or stopped) remains the same across a Service Manager start or stop for the Redirector.
- ▶ The OS/400 proxy server enables all the data to flow through one configured port instead of multiple ports. The Service status (started or stopped) and the trace status (started or stopped) remains the same across a Service Manager start or stop for the OS/400 Proxy Server.
- ▶ Trace allows you to start and stop the server's trace facility, which lets you capture and view information that can help in resolving problems.
- ▶ Service Manager Trace allows you to stop, start, and set the trace level for the Service Manager's trace facility, which lets you capture and view information that can help in resolving problems with the Service Manager. The current service status and trace status remain the same across Service Manager stops and restarts.

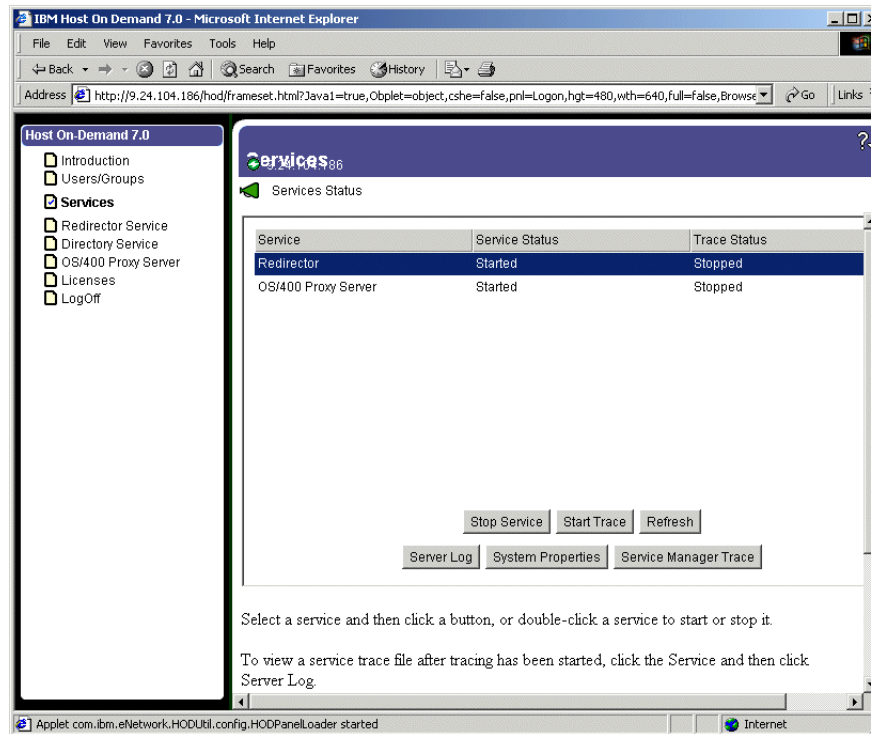


Figure 7-49 Services administration window

Starting and stopping a service or trace

To start and stop a service or trace:

1. Highlight the service.
2. Click the Start/Stop Service or Start/Stop Trace button (depending on which service you are starting or stopping). The Start service and Start trace buttons alternate between start and stop. For example, if you click the Start Service button to begin a service, this same button automatically changes to Stop Service.
3. To see message or trace information created by the Redirector, click Server Log. To make the log easier to read, copy the information to the clipboard and paste it into a file.

Tracing the Service Manager

1. Click Service Manager Trace to display the Service Manager Trace window.

- Trace Active

To turn tracing on, click Yes. To turn tracing off, click No.

- Trace Level

Set the Trace Level by selecting a value from 1 (collects only critical messages) to 3 (collects detailed trace information).

2. Click Apply to start the service manager trace. The trace status and trace level remain the same across Service Manager starts and stops.

The trace is saved in \private subdirectory as NCoDServices.RAS.txt. The trace settings are saved in \private subdirectory as NSMprop.

Refer to Chapter 22, “Problem determination” on page 801 for additional information regarding traces and log files.

7.3 Redirector service

The Redirector is a Telnet proxy that is able to accept connections from clients and pass them on, through a different port, to the next stage in the link. The Redirector can serve as a barrier between clients and the target Telnet server. If you do not want large number of clients connecting directly to your host system because of a security risk, you can have the clients connect to one or more redirectors. The redirectors pass the connection on to the host, allowing you to hide the address of the host from the client users. On Windows NT and AIX, the Redirector provides the support for Secure Sockets Layer (SSL) security between clients and the server.

The Redirector acts as a transparent telnet proxy that uses port remapping to connect Host On-Demand to other telnet servers. Each defined server is given a local-port number. Instead of connecting directly to the target telnet server, a Host On-Demand session connects to the Host On-Demand server. The Redirector maps the local-port number to the host-port number of the target and makes a connection.

Redirectors can be connected to each other (in a cascaded configuration). In that case, SSL security is also available between the Redirectors.

The following scenario shows how the Redirector works. Secure connections are possible between the client and Host On-Demand server.

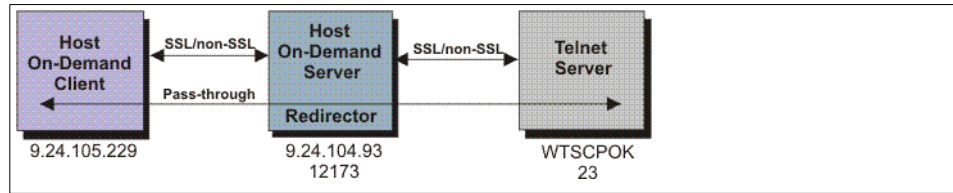


Figure 7-50 How the Redirector works

The Redirector gives Host On-Demand secure access to a wide range of hosts. Typically a Java applet, such as Host On-Demand, is made secure by preventing access to all local and network resources except the host that directly supports the applet.

The Redirector sets security for each host. Security choices are no data-stream modification (pass-through), client-side encryption, host-side encryption, and encryption on all data flowing between the Host On-Demand emulator session and the secure server (both).

Note: The Redirector service must be started at the Services tab.

7.3.1 Configuring the Redirector

If you are going to use the Redirector, you must create a definition in the Redirector for every destination host to which you want an emulator to connect. It is a good idea to configure it before you configure any sessions, since you must use your Redirector definitions (like destination port) when you configure your sessions. Each definition will consist of the address of the target Telnet server (usually a host system), the port on which that server will listen for Telnet connections, and the port (known as the local port) on which the Redirector will listen for connections (from clients) that are destined for the target server.

To configure the Redirector, follow these steps:

1. Log on as an administrator.
2. Click the **Redirector** task, shown in Figure 7-52 on page 353.
3. Click **Add** and Figure 7-51 appears.

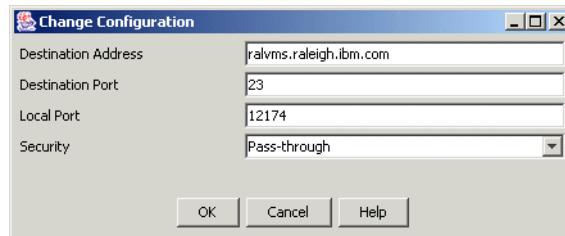


Figure 7-51 Redirector Add Configuration window

4. Type the values for this connection:

► Destination Address

The host name or IP address of the target Telnet server. If the IP address is likely to change, use the host name.

► Destination Port

The port number for the Telnet server through which it will communicate with the Redirector. Many hosts use the default, which is 23, for Telnet connections.

► Local Port

The port number through which the Redirector will communicate with clients.

Use the standard default port numbers or devise a new numbering scheme. When devising a new scheme, use port numbers that are not already defined for other TCP/IP applications. Because most well-known port numbers are lower than 5000, pick a port number between 5000 and 65535 to avoid conflicts.

► Security

Select a security level. Security through the Secure Sockets Layer (SSL) protocol must be set separately for each host configuration. The choices are:

- None
- Client-side
 - provides encryption of data transmitted between the Redirector and the emulator
- Host-side
 - provides encryption of data transmitted between the Redirector and a secure server (host)
- Both

provides encryption of data transmitted through the Redirector, between the emulator and a secure server (host)

Note that in all connections the data will be decrypted as it enters the Redirector and re-encrypted as it exits the Redirector. We recommend that if you require end-to-end encryption you use pass-through if both systems will support a direct negotiation of a secure session.

For more details on secure sessions and setting up the redirector refer to 11.5.2, “Configuring the Host On-Demand Redirector” on page 454.

See also Chapter 25.1.1, “Example of certificate management” on page 879 which shows an example how to use the HOD redirector as a proxy for a secure telnet session with Personal Communications Version 5.6.

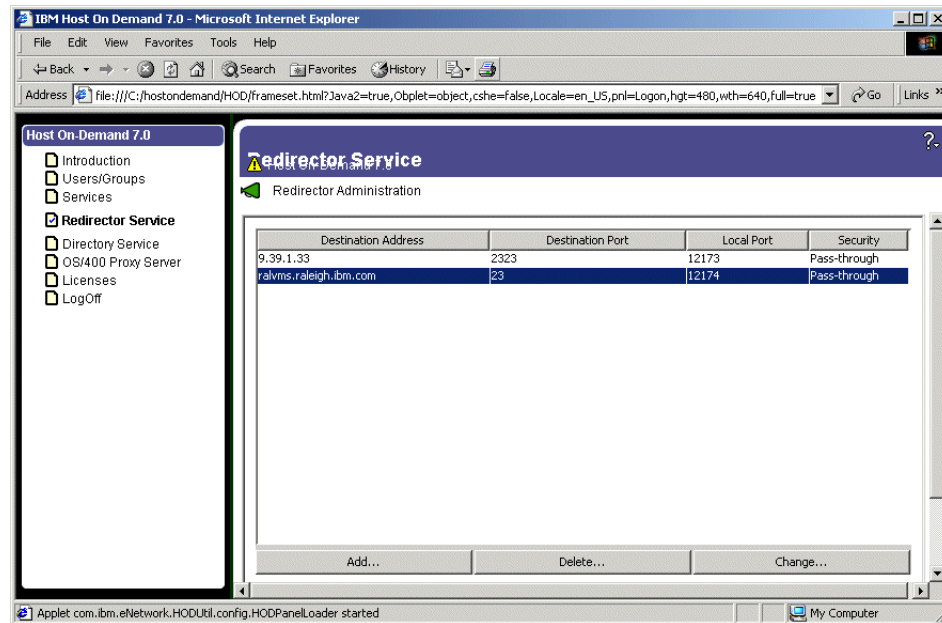


Figure 7-52 Redirector administration - two configured connections

Note: Do not use the Redirector if you don't really need it. It will perform adequately for relatively small numbers of sessions; however, because the Redirector is written in Java, it will not have the performance or capacity that can be obtained by using the Communications Server for AIX Redirector.

7.3.2 Configuring emulator sessions to use the Redirector

Configuring sessions for the Redirector is like configuring any other session, as described in 7.1.7, "Configuring sessions" on page 285.

Configure a redirected session to ITSO

- ▶ Click **3270 Display**. In the Configuration notebook, enter the following information as shown in Figure 7-53 on page 355:
 - Session Name: 3270 Display Raleigh Redir)
 - Destination Address: 9.24.104.186 (the address of the Host On-Demand server)
 - Destination Port: 12174 (the local port at the Redirector for this connection)
 - Click OK to save the session.

If you use one of the administration clients with start session enabled, you can point now at your session icon, click right mouse button and select Start Session. This will launch the selected session as shown in Figure 7-54 on page 355. In the lower right corner below the OIA you see the used IP address and port. This 5 digit port number makes it obvious that the session is not connected to a TN server but via a redirector. The open lock icon next to that indicates that this session does not use security.

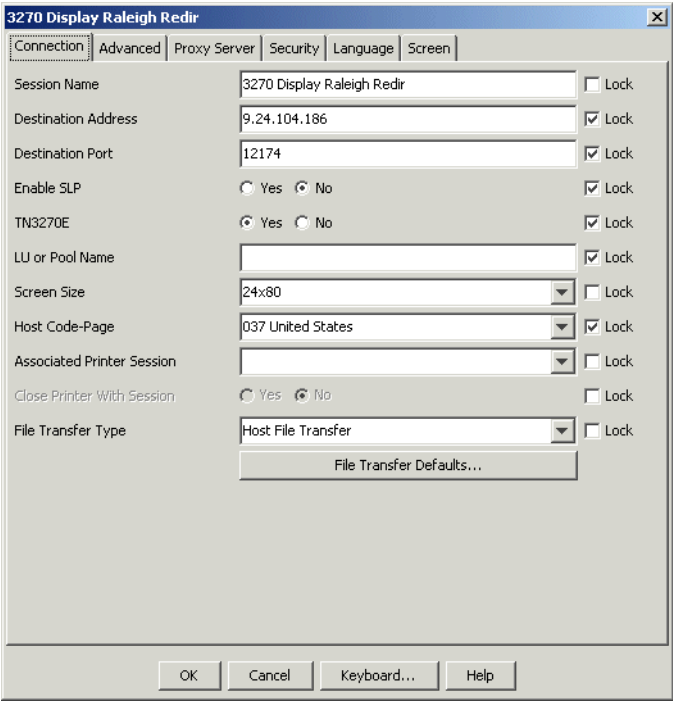


Figure 7-53 Session settings for use with redirector

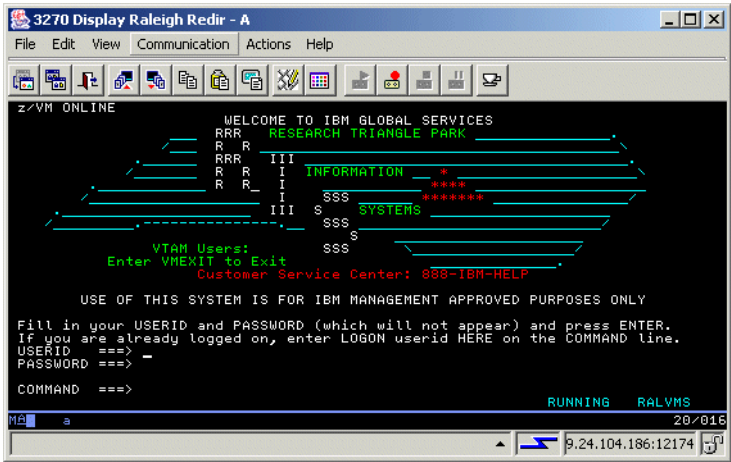


Figure 7-54 Session via redirector

Now we are using the redirector with a PCOMM client:

We set up a Personal Communications Version 5.6 telnet session using the IP address and port of our HOD redirector as shown in Figure 7-55 on page 356.

Telnet3270

Host DefinitionAutomatic Host LocationAdvanced Security Setup

	Host Name or IP Address	LU or Pool Name	Port Number
Primary	9.24.104.186		12173
Backup 1			23
Backup 2			23

Printer Association (only valid for TN3270E Display sessions)

Associated Printer Session

Browse...

☒ Start Associated Printer Minimized

☒ Automatically close the associated printer session with this session

☒ Auto-reconnect

☐ Enable Security

OKCancelApplyHelp

Figure 7-55 PCOMM settings for use with HOD redirector

Again we see in the status bar of the emulator the 5 digit port number, indicating the use of a redirector as seen in Figure 7-56 on page 357.

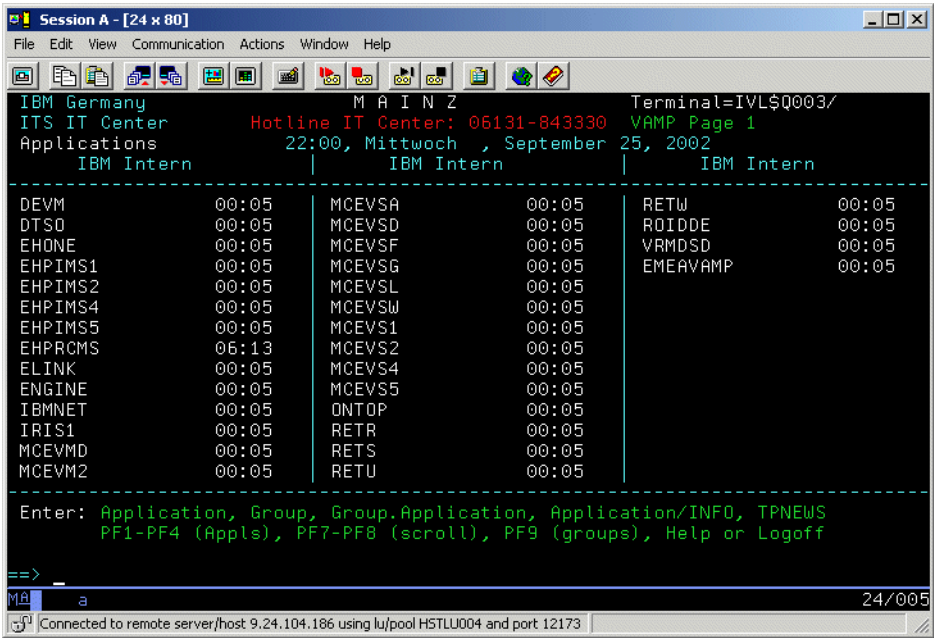


Figure 7-56 Personal Communications Version 5.6 uses HOD redirector

7.4 Directory Service

Enterprise customers often need to manage Host On-Demand user and group configuration information for a large number of users. For reasons of performance or administrative convenience, the information for these users may be distributed and managed across multiple Host On-Demand servers. Unfortunately, the user information is not shared among the Host On-Demand servers or among those servers and other applications.

However, a directory service, such as that provided by a Lightweight Directory Access Protocol (LDAP) server, can enable this kind of information sharing. For example, a single LDAP directory can store configuration information for multiple Host On-Demand servers. Configuration information is stored in directory entries in an LDAP directory; these entries are uniquely identified by a distinguished name (DN).

With Host On-Demand, you can use an LDAP directory instead of using the Host On-Demand server's private data store to store user, group, and session information. This option is available from the Directory Service in the Host On-Demand Administration window.

Note: Migrating to LDAP has significant implications for your group and user configuration information. Make sure you understand these implications before you migrate. See **“Implications of migrating to LDAP” on page 360**

When configuring a Host On-Demand server for use with an LDAP directory server, select the Directory Service task from the navigation area and you will be presented with the window shown in Figure 7-57 on page 358. This window will allow you to enable/disable LDAP directory service, identify the directory server and suffix that you want to use, and optionally let you migrate your existing Host On-Demand configuration data to the LDAP directory server. Regardless of whether or not you migrate, the default administrator ID will be created in the directory server with the default user ID/password of admin/password.

A detailed description of the LDAP directory and its usage can be found in Chapter 8, “LDAP directory server” on page 381.

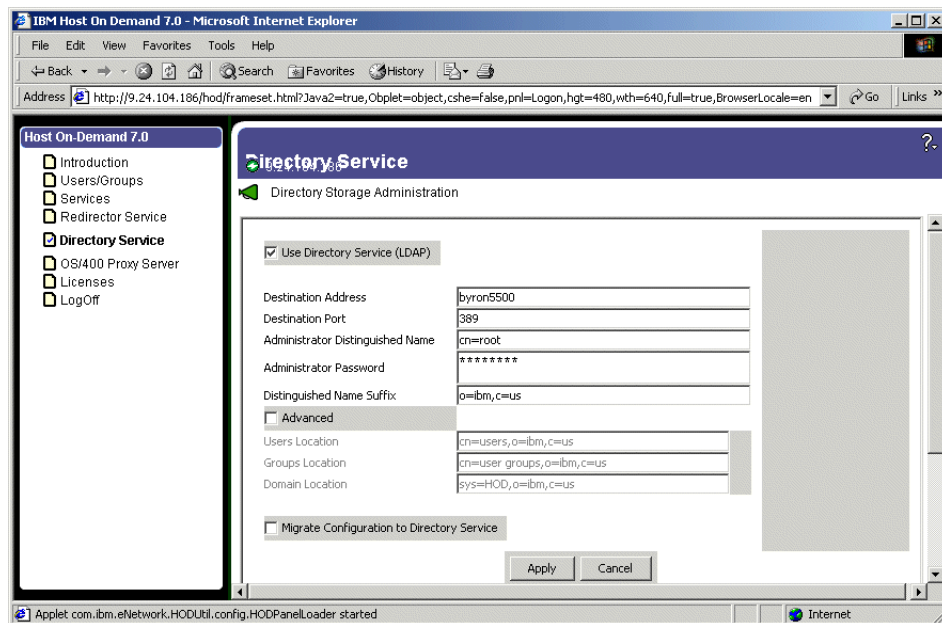


Figure 7-57 Directory Service Administrator

7.4.1 Use Directory Service (LDAP)

To configure Host On-Demand to use an LDAP directory, complete the following fields on the Directory tab in the Host On-Demand administration window:

► Destination Address

Type the IP address of the LDAP directory. Use either the host name or dotted decimal format. The default is the IP address of the Host On-Demand server.

► Destination Port

Type the TCP/IP port on which the LDAP server will accept a connection from an LDAP client. The default port is 389.

► Administrator Distinguished Name

Type the distinguished name (DN) of the directory administrator that allows Host On-Demand to update information. You must use the LDAP string representation for distinguished names (for example, cn=root).

► Administrator Password

Type the directory administrator's password.

► Distinguished Name Suffix

Type the distinguished name (DN) of the highest entry in the directory information tree (DIT) for which information will be saved. Host On-Demand will store all of its configuration information below this suffix in the DIT. You must use the LDAP string representation for distinguished names (for example, o=ibm,c=us).

Under normal circumstances it is recommended that you do not change anything in the Advanced section of this window. The only time you should consider enabling and modifying the Advanced section is if you were connecting to an LDAP directory server and trying to use previously installed directory entries that used the IBM ePerson schema.

7.4.2 Migrate configuration to LDAP directory

To migrate users and groups to an LDAP directory, click Directory Service in the Administration window, check the Migrate Configuration to Directory Service Box and click Apply.

If a group or user already exists in the LDAP directory, the information from the Host On-Demand data store is not written for that particular group or user. Also, if a user is a member of multiple groups in the Host On-Demand data store, the user will be assigned to only one of those groups in the LDAP directory.

During migration, log messages are written to standard output, which is typically the browser's Java console. Additionally the log messages are saved in a log file (hod1dap.log) in the private directory of the Host On-Demand server.

If the migration program ends prematurely, for example, because of a network failure, you can select this option and run the migration program again. After successful migration, the Migrate Configuration to Directory Service check box is automatically cleared. Simply select it and click Apply, and the migration process will begin again.

Once you have switched to the LDAP directory server, subsequent user-related changes will be made only in the LDAP directory, including administrative changes to groups, users, or sessions, and changes such as new passwords, macros, or keyboard changes, by either the administrator or a user.

Implications of migrating to LDAP

This section contains important information about using Host On-Demand with LDAP. You should read and understand this section before using LDAP.

LDAP enables you to manage Host On-Demand configuration information by arranging those users into a hierarchical tree of groups. A group can have one or more subgroups as children and each subgroup inherits all of the sessions defined by the parent group. A user can be an immediate member of any one group and inherits sessions from all the groups in its inheritance tree. This means that you can define sessions in a high-level group for a large number of users and subgroups and then customize them in lower-level groups for smaller numbers of users. It also means that no user can belong to more than one group.

- Will migrating to LDAP change my present group structure and user configurations?

Yes. Because your Host On-Demand private data store is not arranged hierarchically, migrating your configuration information to an LDAP directory changes the relationship between your users and groups. Specifically, all groups and their sessions become children of the root group of the LDAP directory and all users become members of one of the groups they were members of before migration (refer to the migration log for details). Also, because of this change, users that are members of multiple groups will lose configuration information as a result of migration.

- What happens if I choose not to migrate my configuration information?

None of the users, groups, and sessions that are defined in the private data store will be accessible from the logon window or the administration window. If it does not already exist, Host On-Demand will create a single administrator User ID named "admin" with a password of "password."

- ▶ What happens to the configuration information in the private data store when I migrate?

It is preserved and is not modified by the migration process. However, it does not reflect the latest updates either. When you use an LDAP directory, changes to configuration information will only be updated in that LDAP directory.

- ▶ Once I have migrated and started using LDAP, how do I switch back to using the Host On-Demand private data store?

Clear the Use Directory Service (LDAP) box on the Directory tab, and click Apply. This will disable use of the LDAP directory and Host On-Demand will begin retrieving user and group information from the private data store.

- ▶ Is there anyway to migrate my configuration back to the Host On-Demand private data store?

No, migrating from an LDAP directory to the Host On-Demand private data store is not supported.

7.5 OS/400 Proxy Server

On AS400 sessions a file transfer would open a new port and build up a new additional session for that. To funnel all those through only one port the OS/400 proxy server is used. Using one port reduces the security risk when transferring files through a firewall. For details on using OS/400 Proxy Server refer to “OS/400 Proxy” on page 419.

To enable/disable and configure the OS/400 proxy server, you must click the **OS/400 Proxy Server** task in the main HODAdmin.html window. This brings up the window shown in Figure 7-58.

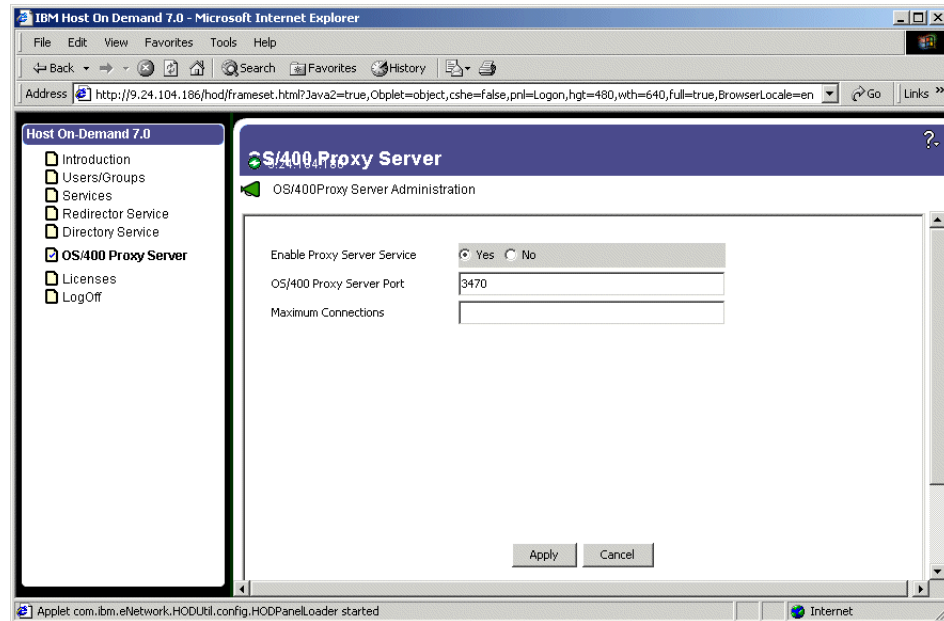


Figure 7-58 OS/400 Proxy Server Administration

To enable the OS/400 Proxy Service select the **Yes** radio button. This enables the remaining fields on this window. You may specify the port you wish the proxy to use (default is 3470). The Maximum Connections field allows you to limit the number of connections; however, unless you are experiencing problems we recommend you leave this blank. You must click **Apply** to make the selections active.

7.6 License Use Management

A Host On-Demand server keeps a count of the number of concurrent users at any given time. This enables you to determine and validate the number of Host On-Demand licenses that you need. A License Use Management (LUM) server enables you to manage and control licenses for Host On-Demand and other software products.

Choose the server that you want clients to report by clicking Licenses in the Administration window. Clients can be switched to report to either type of server at any time. However, the clients that are already connected are not switched until they have logged off or closed the browser and reconnected.

The number of concurrent users is based on a user's ID and IP address. Locally installed clients are not included in this count. Any of the following combination of sessions is counted as a single use:

- ▶ HACL or Beans sessions
- ▶ Emulator sessions
- ▶ Database On-Demand sessions

A license is considered to be in use from the time a session is started until it is closed, regardless of any pattern of usage during that period. If more than one session is active from the same combination of IP address and user ID, only one client is counted.

To take advantage of the license usage support with Host Access Class Library (HACL) and Host Access Bean programs, you must install a Host On-Demand server (from which the programs must be downloaded) and properties must be passed to the `ECLSession` constructor or `Session Bean`. Valid properties are:

- ▶ The type of server that will manage usage. The property name is defined by the constant `ECLSession.SESSION_LUM_LICENSING`, and the value must be LUM or HOD.
- ▶ The identity of the License Use Management server. The property name is defined by the constant `ECLSession.SESSION_LUM_SERVER`, and the value must be the host name or IP address of the License Use Management server.
- ▶ The port number of the License Use Management server. The property name is defined by the constant `ECLSession.SESSION_LUM_PORT`.
- ▶ The identity of the Host On-Demand server. The property name is defined by the constant `ECLSession.SESSION_SERVICE_MGR_HOST`, and the value must be the host name or IP address of the Host On-Demand server.
- ▶ The identity of the user. In multi-user environments, use the User ID property to further refine license-usage counting. This property name is defined by the constant `userid`, and the value must be a string that uniquely defines a user in a multi-user environment.

If you are using AIX as the License Use Manager (LUM) for Host On-Demand, the Basic License Tool (BLT) of the LUM might terminate abruptly, and you won't be able to restart it if there is an active Host On-Demand client with the user profile "default" using a Netscape browser. In order to restart the BLT, stop the Host on-Demand session on the client, close the browser, and delete the "Users" subdirectory in the Netscape directory. Then start the Netscape browser again on the client and define a user profile other than "default". You can then restart the BLT of the AIX LUM, and it will operate correctly.

The Licenses window is shown in Figure 7-59.

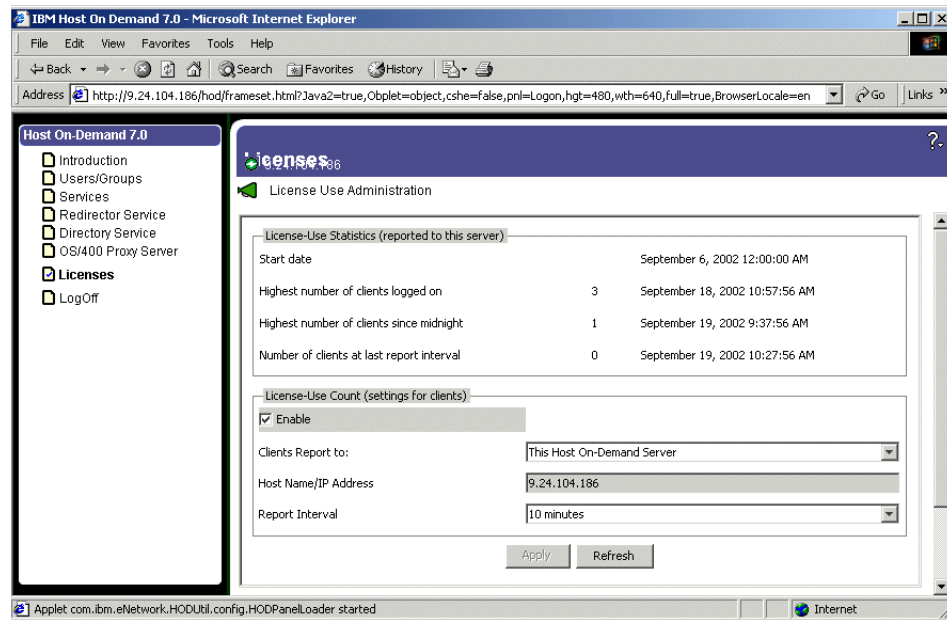


Figure 7-59 Licenses window

7.6.1 Enabling License-Use Count

Host On-Demand counts the sessions when the **Enable** check box in License-Use Counts (settings for client) is selected.

To view information about license usage or enable usage counting, click Licenses in the Administration window. Information from the latest count is displayed when you open the License window or when you click Refresh.

► License Use Statistics

This information applies only when clients are reporting to this Host On-Demand server. Refer to the License Use Management server documentation about reviewing statistical information for clients reporting to a License Use Management server.

► Start date

The date and time that the first check was performed.

► Highest number of clients logged on

The highest number of concurrent users logged on since the start date, and the date and time that this occurred. The overall information is saved in a file named `LicenseOverallHistory.txt` in the `\private` directory. This file contains one entry per day showing the highest number of users each day since the start date and is continuously appended until it is deleted or renamed.

- ▶ Highest number of clients since midnight

The highest number of users since midnight and the date and time that this occurred.

- ▶ Number of clients at last report interval

The number of users when the last count was performed and the date and time this occurred. The information is saved in a file named `LicenseRecentHistory.txt` in the `\private` directory. This file contains entries for the last 12 counts.

- ▶ License Use Count

Configure the Host On-Demand server so that clients downloaded from this server report to a Host On-Demand server or a License Use Management server. You must click **Apply** to activate any changes that you make.

- ▶ Enable

Enables clients downloaded from this server to report to a Host On-Demand server or to a License Use Management server. To stop clients from reporting to a server, clear the check box.

- ▶ Clients Report to

Select whether you want clients to report to a Host On-Demand or a License Use Management server. Use the drop down box to select the servers. If you select **Other Host on Demand Server** you can type over the address shown in the field Host Name/IP Address

- ▶ Host Name/IP Address

Type the host name of the Host On-Demand or License Use Management server that clients must report to. (select first **Other Host on Demand Server** in the drop down list from the **Clients Report to** parameter

- ▶ Report Interval

Select the amount of time for clients to wait between reports. Clients begin using the new interval once the previous interval has expired.

7.6.2 Disabling License-Use Count

You may disable the client from performing license use tracking activities in one of three ways:

1. For those users who log into the Host On-Demand configuration server for preferences, license use tracking may be disabled by clearing the Enable check box on the Licenses window shown in Figure 7-59 on page 364.
2. Clients created with the Deployment Wizard may disable license use counting by adding the HTML parameter on the Additional parameters (of the advanced options) window as shown in Figure 7-60 on page 366.

When using the Deployment Wizard, it is also recommended that you clear the License Use Management check box from the preload configuration window as shown in Figure 7-61 on page 367. This stops the downloading of class file(s) that perform license use tracking. This also makes for a smaller client.

3. Add the following parameter to the HTML file of any Host On-Demand client:
<Param Name=Disable Value=LUM>

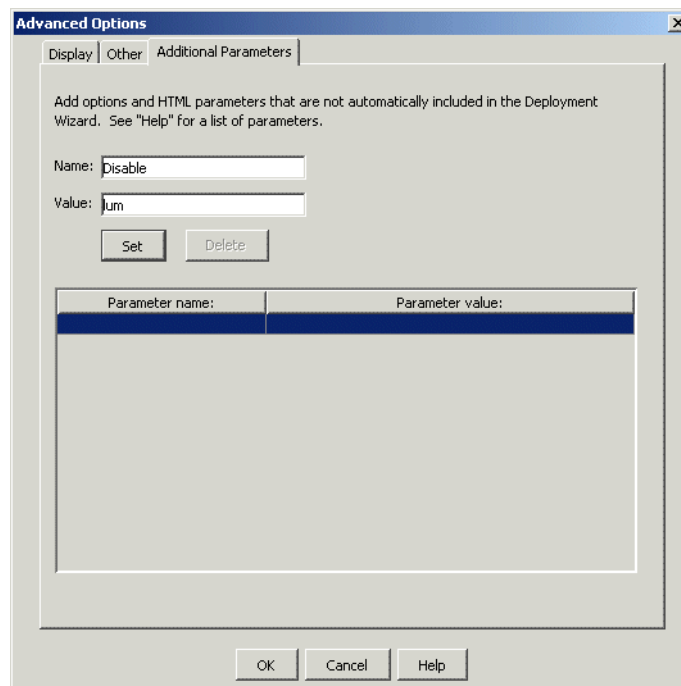


Figure 7-60 Disable License-Use Count with the Deployment Wizard

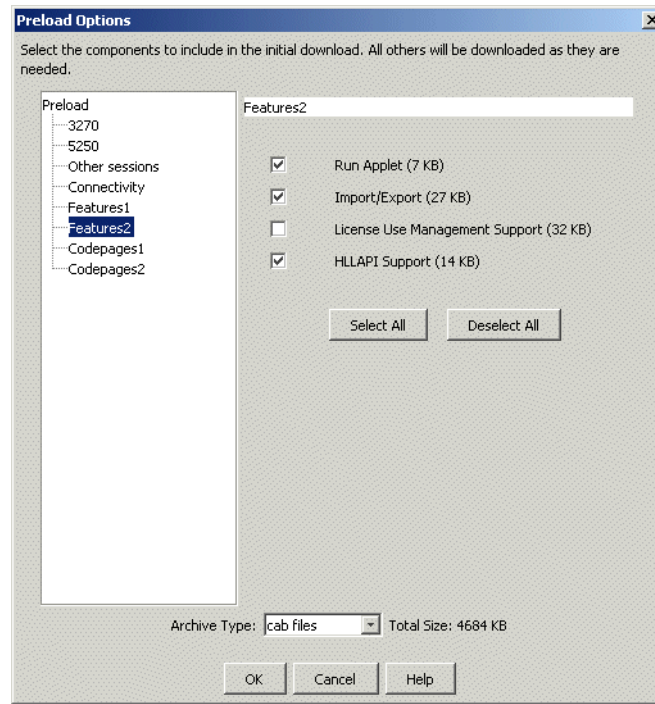


Figure 7-61 Licences management in pre-load options of DW

7.7 Directory Utility

Directory Utility is a command-line Java application the administrator can use to manage user, group or session configuration information. This information is stored either in the Host On-Demand default data store, or in an LDAP directory. This utility is only useful in the environment where the Configuration Server-based model or the combined model is in use. Directory Utility allows you to add, delete, or update large numbers of users, groups or sessions in a batch mode environment instead of using the Administration client. Directory Utility reads an XML ASCII file that contains the following actions to be performed on users, groups, or sessions defined to the Configuration Server:

- ▶ Add, update and delete groups
- ▶ Add, update, and delete users from groups
- ▶ Add, update and delete sessions from users or groups

7.7.1 Using Directory Utility

On Windows NT and Windows 2000, the command file to run Directory Utility and a sample XML file are located in the `hostondemand\lib\samples\DirUtil` directory. The command file is called `DirUtil.cmd` and the sample text file is called `Sample.xml`.

The command, or script, file for other operating systems and the sample XML file are located in the `hostondemand\lib\samples\DirUtilCommandFiles` directory. The sample text file is called `Sample.xml`. The command or script files are shown in Table 7-2.

Table 7-2 Directory Utility command files

Operating System	File name
Windows	DirUtil.cmd
AIX	DirUtil-AIX
Novell	DirUtil-Novell.ncf
OS/2	DirUtil-OS2.CMD
UNIX, HP-UX, Linux, and Solaris	DirUtil-UNIX
iSeries	DirUtil-AS400.sh
zSeries	DirUtil-S390

To run Directory Utility, type the following at the command line:

```
DirUtil-xxx filename.xml admin password [hostname] [port] [CON | FILE]
```

Important: The parameters are positional, not keyword; therefore, the order of the parameters is important.

Where:

`DirUtil-xxx` Directory Utility command or script file for your operating system; see Table 7-2.

`filename.xml` The file that contains the XML elements to manage users, groups and sessions. The text file must have an `.xml` extension, and be a valid XML file. Refer to 7.7.2, “XML file syntax” on page 369 for a description of the XML file commands. If the text file is not in the same directory as the Directory Utility command file, you must specify the path to the file. This parameter must be present on the command line.

admin	Is the Host On-Demand administrator's user ID. This parameter must be present on the command line.
password	Is the Host On-Demand administrator's password. This parameter must be present on the command line.
[hostname]	The Host On-Demand Service Manager's host name or IP address. The default host name is localhost (127.0.0.1). This parameter is optional.
[port]	The Host On-Demand Service Manager's port. The default Service Manager port is 8999. This parameter is optional.
[CON FILE]	Determines how messages will be logged and displayed. If the command line contains the string CON, messages will go only to the console. If the command line contains the string FILE, messages will only be written to a log file. If neither string is included (the default), the messages will be written to the console and also to a log file. The name of the log file is based on the name of the XML file. If your XML file name is myxmlfile.xml, then your log file name is myxmlfile.log.

Note: The Host On-Demand Service Manager must be running on the Host On-Demand server specified by hostname in order for the Directory Utility to update Host On-Demand or LDAP configuration information

7.7.2 XML file syntax

The descriptions of the elements below describe the format for valid XML elements that can be included in the Directory Utility XML control file. A basic understanding of XML is assumed. Note that comment lines begin with an "<!--" and end with "-->". All elements are case sensitive.

You must use an ASCII editor that generates valid unicode characters, such as the Windows Notepad or WordPad editors. If you receive the error DIR0037 Fatal error: Invalid XML Character while using the XML file with Directory Utility, the ASCII editor did not generate valid unicode characters. Use a different ascii editor that does generate valid unicode characters.

The XML elements and their structure is shown below followed by a description of each element.

```
<dirsript>
  <action>

    <group>
      <groupid>
      <description>
```

```
<parent>
<removeusers>

<user>
  <userid>
  <groupid>
  <description>
  <authentication>
    <pw>
    <nativeid>
  <savepref>

<session>
  <filename>
  <groupid>
  <userid>
  <description>
```

<dirsript>

The root element in the XML file that contains all the other elements and identifies the document as one that can be processed by Directory Utility is <dirsript>.

Attributes: none

Required elements: <action type=xxx>

Optional elements: none

<action type=xxx>

This element identifies the action to be performed on the elements enclosed in the <action> element. You can have multiple action elements within the <dirsript> element. Elements placed outside either the <dirsript> element or this element in the XML file are ignored by Directory Utility.

Valid types are:

- “add”
- “delete”
- “update”

At least one of the following elements is required within the <action> element:

► **<group>**

This element identifies the group that is affected by the action. If the action is “add” and the group already exists, you will receive a message that the group is a duplicate.

► **<user>**

This element identifies the user that is affected by the action. If the action is “add” and the user already exists, you will receive a message that the user is a duplicate.

► **<session>**

This element identifies the session that is affected by the action. The session element is not valid when the action is “update”. If the session already exists, a new session named “1:description” is added, in the same way that the Administration client adds a duplicate session.

<group>

There is only one required element, a unicode text string that identifies the group. If you are using Host On-Demand the <groupid> is converted to uppercase when the group is added. If you are using LDAP, the <groupid> can be mixed-case.

The optional elements are:

► **<description>**

► **<parent>**

► **<removeusers>**

<description>

This element is a unicode text string that describes the group and is only valid when the action type is “add” or “update”.

<parent>

This element identifies the parent of this group. This element is only valid when the action is “add” or “update”, and when using LDAP. If this element is not specified when the action is “add”, the group is added to the top level.

<removeusers>

This element allows you to delete all the users that belong exclusively to this group when you delete the group. This element is only valid when the action is “delete”, and this element is not valid when using LDAP. Valid values are Yes and No. If Yes is specified, then the users in this group will be deleted when the group is deleted. If No is specified and there are users in the group, the users that belong only to this group are moved to the HOD group and the group is deleted.

The Default is No.

If you have many users, it may take some time for the processing of this element to complete.

<user>

This is the enclosing element in defining a user.

Attributes: none

The following elements are required:

► **<userid>**

This is the identifier that defines the user. This element is always required. If you are using Host On-Demand, the **<userid>** is converted to lowercase. If you are using LDAP the **<userid>** can be mixed-case.

► **<groupid>**

This element defines the group to which the user is being added. This element is required when the action type is “add” and ignored when the action type is “delete”. If you are not using LDAP, you can specify multiple **<groupid>** elements. If you are using an LDAP directory, a user can exist in only one group; therefore, if you specify multiple groups, an error message will be generated and the user is not added. Groups specified must exist before you can add users to them. If the action type is “update”, the user is updated to have membership in this group.

The following elements are optional:

► **<description>**

A unicode text string that describes the user.

► **<authentication type=xxx>**

Specifies the type of authentication that is used for the user. Valid types are “native” and “pw”. If this element is omitted, no authentication will be configured for the user.

► **<savepref>**

Specifies if the user is authorized to save preferences (changes that the user might make to a host session configuration). Valid values are Yes or No. If this element is not specified, the default of Yes will be used.

► **<removegroupid>**

You can update a user so that they no longer have membership in a specified group. This element is only valid if the action type is “update”. You must use a valid **<groupid>** that contains this user.

<authentication type=xxx>

This element specifies the authentication used for the user. You can use password authentication, or Native Authentication if the Service Manager is using an LDAP directory. No authentication will be configured for the user if this element is not specified when the action type is “add”, or if “native” is selected and you are not using an LDAP directory.

Valid values are:

- ▶ “pw”
- ▶ “native”

The following element is required:

- ▶ <nativeid>

The value specified here is the ID of the user on the native operating system. This element is required and valid only when using LDAP directory and when the authentication type is “native”.

The following elements are optional:

- ▶ <pw>

The value entered is the password associated with the user. This element is only valid when the authentication type is “pw”.

- ▶ <change pw>

Specifies whether or not the user is authorized to change his password. Valid values are Yes or No. If this element is omitted a default of Yes is set for the user, enabling the user to change their password. This element is only valid if the authentication type is “pw” and is ignored if the authentication type is “native”.

<session>

This element defines a session available to the user.

The required elements to define the session are:

- ▶ <filename>

This element specifies the file containing the session definition. The session definition file may be created by using the Export Session menu option from any defined Host On-Demand session. The default file extension for session files is .hod. If the file does not exist in the directory from which Directory Utility is run, then the <filename> element should contain the full path to the session file, and it is only required when adding a session.

- ▶ <description>

The description is a unicode text string that describes the session and is used as the session name. The <description> is required to update or delete a session. If <description> is omitted, the session name will be used for the description of the file.

At least one of the following element types is required:

Note: You can specify multiple users or multiple groups in the session element, but you cannot specify both users and groups in the same session element.

► <userid>

The user identifier for the user to which this session is being added. User IDs must already exist before the session can be added. You can include multiple <userid> elements to add this session for multiple users.

► <groupid>

This element specifies the identifier for the group to which the session is being added. Groups must already exist before the session can be added. You can include multiple <groupid> elements to add a session for multiple groups.

7.7.3 Example

This example illustrates how to take the objective of the bulk update, translate the requirements into the required XML file format, and run Directory Utility.

Objectives

The objectives of this example are to add the following groups and users:

- Management
 - Authentication: none
 - Members:
 - Joe Cline (jcline)
 - John Bird (jbird)
 - One session at the group level: WTSCPOK
- ProjectLeaders
 - Place under the Management group
 - Authentication: Host On-Demand password
 - Members:

- George Baker (gbaker)
 - David Russell (drussell)
 - Steve Watts (swatts)
- ▶ Residents
 - Authentication: Native Authentication
 - Members:
 - Bob Bogardus (bogardus)
 - Alan Cohen (cooley)
 - Anna Murphy (parks)

Next, the following sessions must be available to the indicated groups. The session definitions were previously created by the administrator using the Export Session option from the graphical administrative interface and stored in the same directory as the utility command file and sample XML file.

- ▶ Management
 - WTSCPOK
- ▶ ProjectLeaders
 - WTSCPOK
 - MVS03a
- ▶ Residents
 - MVS03a
 - HODAIX

Finally a single session, HODLinux, is added for user George Baker.

Sample XML file

Example 7-1, “ITSOSample.xml” on page 375 illustrates the XML file that was used.

Example 7-1 ITSOSample.xml

```
<?xml version="1.0" encoding="UTF-8"?>

<!-- Begin DTD - The DTD should not be modified.-->
<!DOCTYPE dirsript [
<!ELEMENT dirsript (action)+>
<!ELEMENT action (group | user | session)+>
<!ELEMENT group (groupid, description?, parent?, removeusers?)>
<!ELEMENT user (userid, groupid*, description?, authentication?, savepref?,
removegroupid?)>
<!ELEMENT session (filename?, (groupid | userid)+, description?)>
<!ELEMENT groupid (#PCDATA)>
<!ELEMENT userid (#PCDATA)>
```

```

<!ELEMENT description (#PCDATA)>
<!ELEMENT parent (#PCDATA)>
<!ELEMENT removeusers (#PCDATA)>
<!ELEMENT removegroupid (#PCDATA)>
<!ELEMENT authentication ((pw?, changepw?) | (nativeid))>
<!ELEMENT pw (#PCDATA)>
<!ELEMENT changepw (#PCDATA)>
<!ELEMENT nativeid (#PCDATA)>
<!ELEMENT savepref (#PCDATA)>
<!ELEMENT filename (#PCDATA)>
    <!ATTLIST action type (add | delete | update) #REQUIRED>
    <!ATTLIST authentication type (pw | native) #REQUIRED>
]>
<!-- End DTD -->

<dirsript>
    <action type="add">
        <!-- Add three groups, Management, Project Leaders, Residents -->
        <group>
            <groupid>Management</groupid>
            <description>ITSO Managers</description>
        </group>
        <group>
            <groupid>ProjectLeaders</groupid>
            <description>ITSO Project Leaders</description>
            <!-- the parent element should only be specified if using LDAP -->
            <parent>Management</parent>
        </group>
        <group>
            <groupid>Residents</groupid>
            <description>SG24-6182 Residents</description>
        </group>

        <!-- The following sessions were previously exported and reside in the
executing
        directory: WTSCPOK.hod, HODAIX.hod, HODLinux.hodm MVS03a.hod -->

        <!-- Add a session to the Management group -->
        <session>
            <description>WTSCPOK</description>
            <filename>WTSCPOK.hod</filename>
            <groupid>Management</groupid>
        </session>

        <!-- Add a sessions to the Project Leaders group -->
        <session>
            <description>MVS03a</description>
            <filename>MVS03a.hod</filename>
            <groupid>ProjectLeaders</groupid>

```

```

</session>

<!-- Add a sessions to the Residents group -->
<session>
  <description>HODAIX</description>
  <filename>HODAIX.hod</filename>
  <groupid>Residents</groupid>
</session>
<session>
  <description>MVS03a</description>
  <filename>MVS03a.hod</filename>
  <groupid>Residents</groupid>
</session>
<session>
  <description>MHODLinux</description>
  <filename>HODLinux.hod</filename>
  <groupid>Residents</groupid>
</session>

<!-- Add Management users, no passwords -->
<user>
  <userid>jcline</userid>
  <description>Joe Cline</description>
  <!-- note the authentication element is missing, resulting in no
password -->
  <savepref>No</savepref>
  <groupid>Management</groupid>
</user>
<user>
  <userid>jbird</userid>
  <description>John Bird</description>
  <savepref>No</savepref>
  <groupid>Management</groupid>
</user>

<!-- Add Project Leaders with basic passwords -->
<user>
  <userid>gbaker</userid>
  <description>George Baker</description>
  <authentication type="pw">
    <pw>gwbpassword</pw>
    <change pw>yes</change pw>
  </authentication>
  <groupid>ProjectLeaders</groupid>
  <savepref>Yes</savepref>
</user>
<user>
  <userid>drussell</userid>
  <description>David Russell</description>

```

```

    <authentication type="pw">
      <pw>drpassword</pw>
      <changepw>yes</changepw>
    </authentication>
    <groupid>ProjectLeaders</groupid>
    <savepref>Yes</savepref>
  </user>
  <user>
    <userid>swatts</userid>
    <description>Steve Watts</description>
    <authentication type="pw">
      <pw>swpassword</pw>
      <changepw>yes</changepw>
    </authentication>
    <groupid>ProjectLeaders</groupid>
    <savepref>Yes</savepref>
  </user>

  <!-- Add Residents with Native Authentication passwords -->
  <user>
    <userid>bogardus</userid>
    <description>Bob Bogardus</description>
    <authentication type="native">
  <!-- notice changepw is ignored when using native authentication -->
    </authentication>
    <groupid>Residents</groupid>
    <savepref>Yes</savepref>
  </user>
  <user>
    <userid>cooley</userid>
    <description>Alan Cohen</description>
    <authentication type="native">
    </authentication>
    <groupid>Residents</groupid>
    <savepref>Yes</savepref>
  </user>
  <user>
    <userid>parks</userid>
    <description>Anna Murphy</description>
    <authentication type="native">
    </authentication>
    <groupid>Residents</groupid>
    <savepref>Yes</savepref>
  </user>
  <!-- Add a session to George Baker -->
  <session>
    <description>HODLinux</description>
    <filename>HODLinux.hod</filename>
    <userid>gbaker</userid>

```

```

</session>

</action>
</dirsript>

```

Command-line options

Here are some samples of executing Directory Utility using the sample XML file specified in Example 7-1 on page 375.

1. If running Directory Utility on an AIX machine with the Host On-Demand server on the same machine using the default port, you would specify:
`DirUtil ITSOSample.xml admin password 127.0.0.1`
2. If running Directory Utility on a z/OS system with the Host On-Demand server on the same machine using port 8998 for the Configuration Server port, you would specify:
`DirUtil ITSOSample.xml admin password 127.0.0.1 8998`
3. If running Directory Utility on a Windows system running a locally installed client with the Host On-Demand server on another machine, listening on the default port, you would specify:
`DirUtil ITSOSample.xml admin password HODLinux.itso.ra1.ibm.com`

7.8 Java 2 considerations for iSeries

If you have configured Host On-Demand to use a V1.3 JVM, you must perform the following to allow the DirUtil script to operate properly:

- ▶ Run EDTF `'/qibm/proddata/hostondemand/lib/samples'`
- ▶ Type 5 next to DirUtilCommandFiles
- ▶ Type 2 next to DirUtil-AS400.sh
- ▶ Locate the words “Modify the following to specify your java engine”
- ▶ Add `JAVA_ENGINE="java -Djava.version=1.3"`
- ▶ Comment out the other `JAVA_ENGINE` statement by typing a # in the first column of the line.



LDAP directory server

In its default configuration, Host On-Demand stores its configuration information in a non-shared private data store. Enterprise customers often need to manage Host On-Demand user and group configuration information for multiple Host On-Demand servers. If these enterprise customers were to use the default no-shared private data store, it would require them to separately manage each instance of Host On-Demand. The deployment and use of an LDAP directory simplifies the administration of multiple Host On-Demand servers using the Configuration Server-based model. In addition, if your intent is to deploy Native Authentication the deployment of LDAP directory is also required. For details about Native Authentication refer to 11.11, “Native Authentication” on page 468.

This chapter focuses on the following issues relative to the implementation of an LDAP directory server with Host On-Demand:

- ▶ LDAP overview
- ▶ Supported LDAP directory servers
- ▶ Configuring supported LDAP directory servers
- ▶ Host On-Demand LDAP directory operations
- ▶ Availability, Performance, Backup and Recovery

8.1 LDAP overview

This section provides an introductory overview of LDAP and the benefits it provides for administrators of large network enterprises. Readers who want a broader understanding of the LDAP model than that provided here should refer to the following redbooks:

- ▶ *Understanding LDAP*, SG24-4986
- ▶ *LDAP Implementation Cookbook*, SG24-5110
- ▶ *Using LDAP for Directory Integration: A Look at IBM SecureWay Directory, Active Directory, and Domino*, SG24-6163

The basic unit of information stored in the directory is called an entry. An entry represents objects of interest such as organizations, people, servers, etc. Entries are composed of a collection of attributes that contain information about each object. Every attribute has a type and one or more values. The type of the attribute is associated with a syntax, which specifies what values can be stored. For example, an entry can have a telephone attribute. The syntax associated with this type of attribute would specify that the values are telephone numbers represented as printable strings. It is possible for the directory entry for an organization to have multiple values in this attribute. In addition to defining what data can be stored as the value of an attribute, the syntax of an attribute also defines how those values behave during searches and other directory operations.

Schema files define the types of objects that can be stored in the directory; they also list the attributes of each object type and whether these attributes are required or optional. For example, in the person schema, the attribute *surname* is required, but the attribute *description* is optional. Schema checking ensures that all required attributes for an entry are present before an entry is stored.

Host On-Demand uses a specific schema, called ePerson. This schema is shipped with all current IBM LDAP directory servers; however, if you are using the Netscape directory server or an older release of an IBM LDAP directory server you will need to extend the schema of that directory server to include the ePerson schema. Host On-Demand provides this schema. If required, the schema files must be manually installed on an LDAP server, and must be in effect before Host On-Demand can store configuration information in an LDAP server. Refer to 'Schema Installation' for detailed instructions on the installation of the Host On-Demand schema files.

8.2 Host On-Demand and LDAP overview

The default implementation of Host On-Demand uses a private data store model that does not provide sharing across servers; however, you may optionally use an LDAP directory to store and retrieve configuration information. You may start your Host On-Demand deployment by using the LDAP directory server or you may start using the default private data store and migrating your configuration information to the LDAP directory server later. You can later revert to using the private data store if, for example, you cannot connect to the LDAP directory for any reason; however, you cannot migrate information from an LDAP directory back to a private data store. Specific migration issues are detailed in 8.5.3, “LDAP migration implications” on page 390.

8.3 Supported LDAP directory servers

Host On-Demand supports the following LDAP directory servers:

- ▶ IBM LDAP Directory Server V2.1, V3.1.1, V3.2.1 (also known as SecureWay directory).

This directory server is available from the following IBM Internet site:

<http://www.ibm.com/software/network/directory/>

The schema must be extended for V2.1 as documented in 8.4, “Schema installation” on page 384; however, no action is required for all V3 releases, since the proper schema has been integrated into the base schema.

- ▶ Netscape Directory Server V3.1 and V4.0 (both on Windows and AIX)

Information on the Netscape Directory Server may be obtained at:

<http://enterprise.netscape.com/products/identsvcs/directory.html>

The schema for all Netscape directory servers must be extended as documented in 8.4, “Schema installation” on page 384.

- ▶ IBM LDAP Server running on zSeries Version 2 Releases 5, 6, 7, 8, 9, 10, 11, and 12

The schema for this LDAP directory server must be extended for Version 2 Release 5, 6, 7, 8, and 9. Beginning with Release 10, the Host On-Demand required schema was shipped in the default schema for the TDBM configuration option. Refer to 3.8, “LDAP directory server” on page 137 for complete details on installing and using the zSeries LDAP directory server.

8.4 Schema installation

The IBM standard schema is required by Host On-Demand. If you are using an LDAP directory server that does not already support this schema, see 8.3, “Supported LDAP directory servers” on page 383. You must extend the schema. Host On-Demand provides the extensions for these servers in several files that are located in the publish subdirectory of the Host On-Demand installation directory (for example, C:\hostondemand\HOD\ldap). These files contain extensions to the shipped LDAP schema and are stored in standard slapd format. The schema extensions must be in effect before Host On-Demand can contact, and store configuration information in an LDAP server. If your LDAP administrator has already installed these schema extensions for use by another IBM product, you can skip the following steps; otherwise, follow these steps to install the schema on your directory server.

8.4.1 Netscape Directory Server

Follow these instructions to install the schema on the Netscape Directory Server.

1. Copy the following files from the \hostondemand\HOD\ldap directory to the Netscape LDAP config directory on the LDAP server:

Netscape.IBM.at

Netscape.IBM.oc

2. Stop the LDAP server.
3. Edit the <Netscape LDAP config directory>/slapd.conf file and add the following statements:

```
userat "<Netscape LDAP config directory>/Netscape.IBM.at"
useroc "<Netscape LDAP config directory>/Netscape.IBM.oc"
```
4. Restart the LDAP server.

8.4.2 IBM SecureWay LDAP Directory Server

If you are using the IBM LDAP Directory Server on zSeries, you will find your instructions in 3.8.1, “Schema installation” on page 138. If you are using an IBM SecureWay LDAP Directory Server V3 server, you may skip this section, since the required schema is shipped as part of the base schema.

If you are using the IBM SecureWay LDAP Directory Server V2.1, follow the instructions below to properly install the schema:

1. Copy the following files from \hostondemand\HOD\ldap to your LDAP server <installation directory>\etc directory:

V2.1.IBM.at

V2.1.IBM.oc

2. Stop the LDAP server.
3. Edit the <installation directory>\etc\slapd.at.conf file and add the following statement to the end of the file:

```
include /etc/V2.1.IBM.at
```

4. Edit the <installation directory>\etc\slapd.oc.conf file and add the following statement to the end of the file:

```
include /etc/V2.1.IBM.oc
```

5. Restart the LDAP server.
6. Define the suffix for the Host On-Demand server.

Consult with the administrator of your directory server before proceeding, because the administrator must define the directory structure that Host On-Demand will use. The following steps guide you through the process of defining to the directory server the suffix that has been selected.

- a. Log on to the LDAP server.
 - b. Select the **Suffix** section.
 - c. Select **List Suffixes** and determine if the suffix required is present; if not present, initialize the suffix by selecting **Add Suffix**, and add the desired suffix. An example is o=ibm.
 - d. Restart the LDAP server.
7. Initialize the suffix.

Before the suffix can be used and information stored in the directory, the suffix must be initialized. If your LDAP administrator has previously defined the suffix and initialized it, you have completed the schema installation.

You can determine whether the suffix has already been initialized by following these instructions after logging on to the directory server:

- a. Select **Directory/Access Control - Browse Tree**.
- b. If your suffix is listed, you have completed the schema installation; otherwise, follow the remaining instructions to initialize the suffix.
- c. Select the **Database** section.
- d. Select **Add Entries**.
- e. Enter the fully qualified name of the file that contains the seed record for the suffix. This file must reside on the LDAP server and be in LDIF format.

LDIF describes a directory and directory entries in text format and is commonly used to build a directory database or add large numbers of entries at once. The following is an example of the contents of an LDIF file that would be used to initialize a suffix called o=ibm.

```
dn: o=ibm
objectclass: organization
o: ibm
```

- f. Click **Add Entries to Database**.
- g. Restart the LDAP server. The LDAP server is now ready for use by Host On-Demand.

8.5 Host On-Demand directory operations

The default operational mode for Host On-Demand is to use the private data store. Using an LDAP directory server to manage and share your definitions across multiple Host On-Demand servers is an option that must be carefully planned and executed.

8.5.1 Switching to an LDAP directory server

Enabling LDAP directory support is performed by the Administrator via the window shown in Figure 8-1. Details on how to complete this window are discussed in 7.4.1, “Use Directory Service (LDAP)” on page 359. This section discusses the process that takes place when you switch from the private data store to an LDAP server, and some common events that may occur when you do.

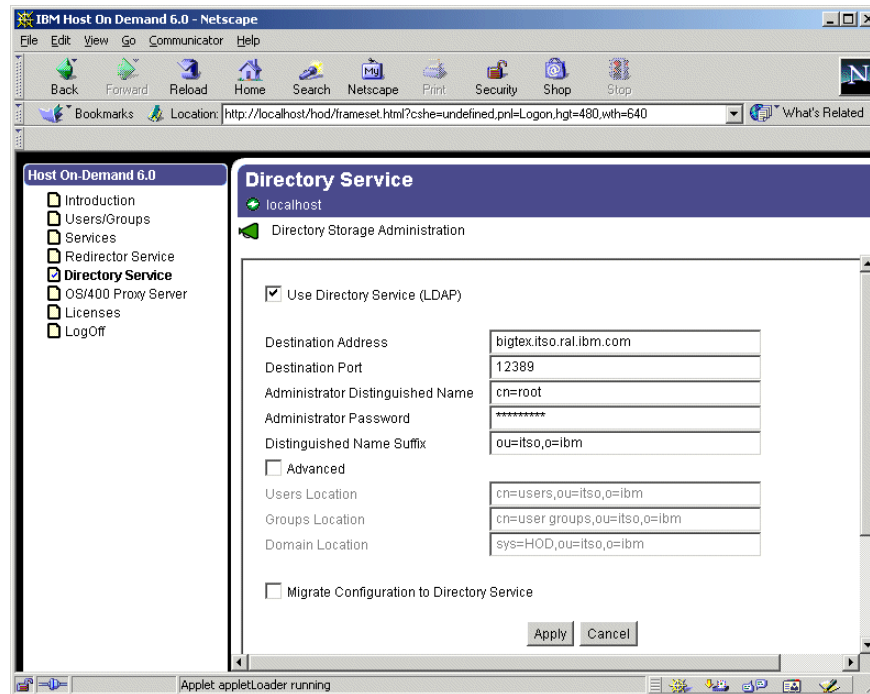


Figure 8-1 Enable LDAP directory support

It is important to understand the process of switching to the LDAP directory so that you will realize what is happening if the window shown in Figure 8-2 appears while the switch to the LDAP server is taking place.



Figure 8-2 LDAP password failure

To understand what caused the error, you must understand what happens during the switch. Figure 8-3 depicts the process flow when the administrator clicks **Apply** on the window shown in Figure 8-1 to activate the LDAP directory.

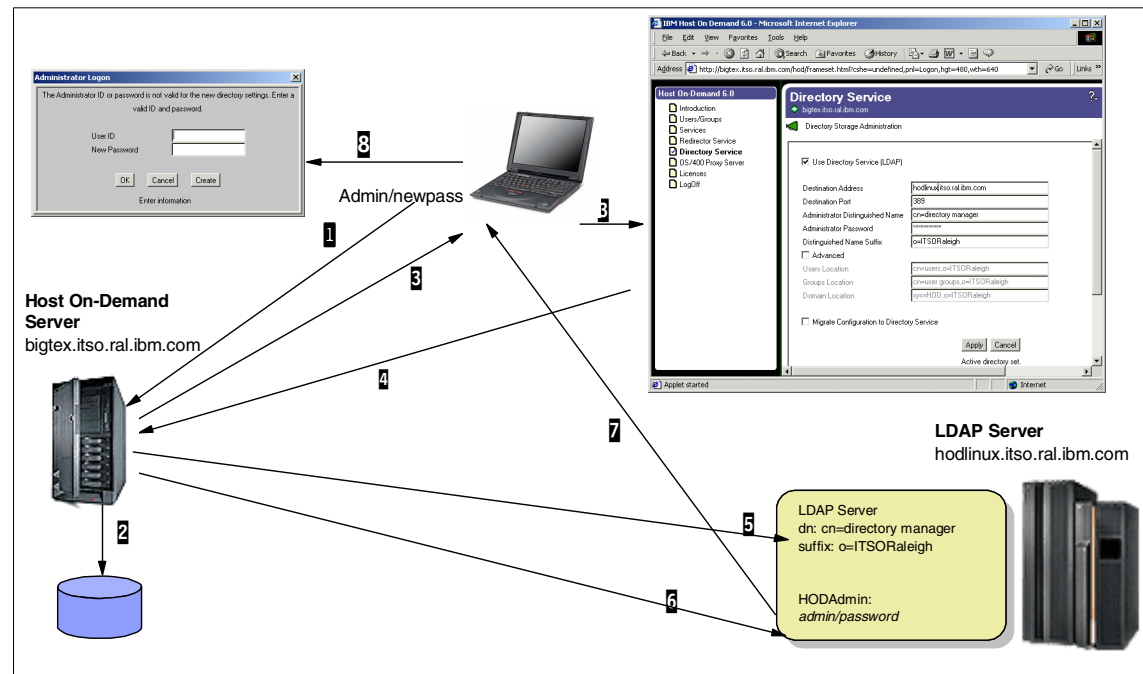


Figure 8-3 Switch to LDAP directory server

1. The administrator logs on, using admin as the user ID and newpass as the password (assume the password had been previously changed).
2. The Host On-Demand Service Manager successfully authenticates the user ID/password (admin/newpass) with that contained in the private data store (admin/newpass).
3. The administrator enters the required information to switch to the LDAP server.
4. When the administrator clicks **Apply**, the Service Manager contacts the LDAP server, passing the LDAP administrator's distinguished name (DN) (cn=directory manager) and password (note that this password is not the password used to log in to HODAdmin.html), and the suffix (o=ITSORaleigh), at which point the Host On-Demand information will be inserted into the directory.
5. The LDAP server authenticates the request as follows:
 - a. Are the DN and password valid as the LDAP administrator?
 - b. Is the suffix defined and initialized?
 If the DN and suffix are validated, the Service Manager can proceed.

6. The Host On-Demand Service Manager attempts to create the default Host On-Demand administrator user ID (`admin`) and password (`password`). If this is the first time this directory has been contacted (or if the `admin` ID does not exist), this update will succeed. If the `admin` ID already exists, the request will fail to create the user ID since it is already present (the existing password need not be `password`); however, the Service Manager ignores the failure and proceeds.
7. This step is similar to step 2. on page 388. The only difference is that this step is performed with the new LDAP settings. The administration applet is now prompted to reauthenticate itself to the LDAP server, and it responds by sending `admin/newpass`. In this example, the authentication fails (the directory service is expecting `admin/password`), which results in the window shown in Figure 8-2 on page 387. You should enter the user ID and password of an administrative user that are valid for the new LDAP settings (in our example, `admin/password`).

Note: Remember that multiple Host On-Demand systems can use the same LDAP server and suffix to allow for workload balancing, so another system may have reset this password.

8. After the appropriate user ID/password combination is entered, the user is then authenticated and the Service Manager finishes creating the required information in the LDAP directory.
9. Finally, if the migration check box was checked, the information in the private data store is created in the LDAP directory. An audit trail of all updates is maintained in `hostondemand\private\hodldap.log`.

8.5.2 Unable to enable LDAP

If you are using the Netscape LDAP directory and are unable to successfully enable LDAP, you may need to temporarily disable the UID uniqueness filter in the LDAP directory. Each user or person object created in an LDAP directory has a User ID (UID) field. Some of the objects created by Host On-Demand during the initialization process may be rejected by the UID uniqueness filter.

To resolve this problem do the following:

1. Open the Directory Server Console for your Netscape Directory
2. Navigate to the Configuration tab
3. Expand plug-ins
4. Select **uid uniqueness**
5. Clear the associated check box

6. Click **Save**
7. Restart the LDAP server

Important: Do *not* leave UID uniqueness disabled permanently. The UID uniqueness plug-in should be disabled only if you are having trouble enabling LDAP for Host On-Demand. After enabling LDAP, you should re-enable UID uniqueness.

8.5.3 LDAP migration implications

Before converting to the LDAP directory server, make sure that you understand all the issues in this section, because any changes made in the LDAP directory cannot be migrated back to the private data store.

Hierarchical structure

LDAP enables you to manage Host On-Demand configuration information by arranging those users into a hierarchical tree of groups. A group can have one or more subgroups as children, and each subgroup inherits all of the sessions defined by the parent group. A user can be an immediate member of any one group and inherits sessions from all the groups in its inheritance tree. This means that you can define sessions in a high-level group for a large number of users and subgroups and then customize them in lower-level groups for smaller numbers of users. It also means that a user cannot belong to more than one group.

The Host On-Demand private data store is not arranged hierarchically; therefore, migrating your configuration information to an LDAP directory changes the relationship between your users and groups. Specifically, all groups and their sessions become children of the specified suffix in the LDAP directory, and all users become members of one of the groups that they were members of before migration. Users that are members of multiple groups will not lose configuration information as a result of migration, because the group settings will be allocated to them as individual users.

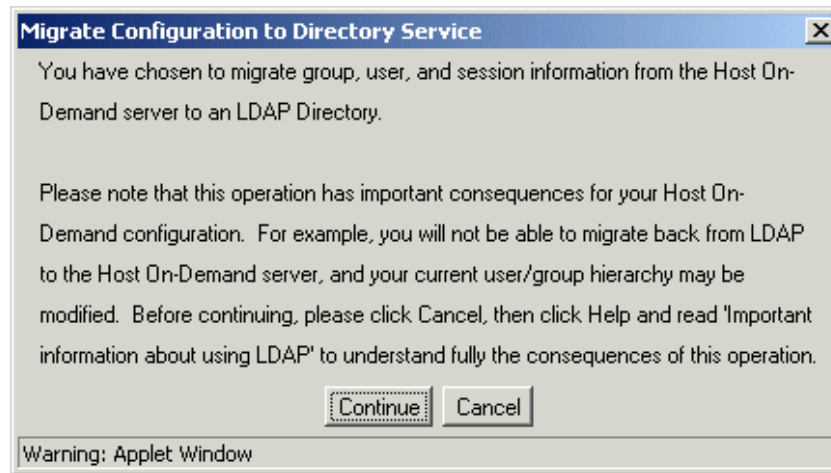


Figure 8-4 Migration warning

Migration process

The migration process assumes that items are taken from the private data store and added to the LDAP directory. Migration is subject to the following rules:

- ▶ If a group or user already exists in the LDAP directory, that entry is skipped.
- ▶ Groups are migrated before users.
- ▶ A user is added to a single group. There is no way of knowing exactly which group a user will be added to; however, it appears to be a function of the order in which the groups are created in the private data store and the order in which the user is added to a group. The migration log, `hodldap.log`, provides details about to which group a user is added.
- ▶ Sessions that a user would have inherited from groups that the user did not migrate to are not lost. These sessions are assigned at the user level.

During the migration, a progress indicator will be displayed, showing the status of the process (Figure 8-5).

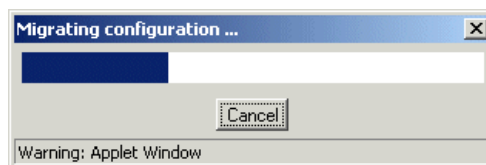


Figure 8-5 Migration progress

Private data store status

After you have migrated your configuration information to the LDAP directory, and as long as you are using the LDAP directory, the users, groups, and sessions that are defined in the private data store will not be accessible to the administrator or a client. The private data store is preserved and is not modified by the migration process, but any changes made in the LDAP directory cannot be migrated back to the private data store.

Reverting to the private data store

You can revert to using the Host On-Demand private data store, but remember that none of the user or group changes made while you were using the LDAP directory will be reflected in the private data store. Follow the instructions below to switch back to the private data store:

1. Log on to Host On-Demand as the administrator.
2. Select **Directory Service** (refer to Figure 8-1 on page 387).
3. Clear the Use Directory Service (LDAP) check box.
4. Click **Apply**.

Note: You do not have to restart the Service Manager when switching back to the private data store. If you later want to switch back to the same directory server, you can do so without restarting the Service Manager. However, if you switch to a different directory, follow the instructions in the next section.

Switching to a Different LDAP directory server

If you are using an LDAP directory as a data store, and it becomes necessary to switch to another LDAP server, you must perform the following steps carefully:

1. Revert to the private data store; refer to “Reverting to the private data store” on page 392 for instructions.
2. Restart the Service Manager.
3. Log on as the administrator and enter the address information for the new LDAP server (refer to 7.4.1, “Use Directory Service (LDAP)” on page 359).
4. Activate the new LDAP directory server.

Multiple Host On-Demand system implementation

It is recommended that you create a separate administrator ID to be used for LDAP on all your Host On-Demand servers. Use this ID to do your administration, and leave the defaults (admin/password) alone. The advantage of this is that regardless of which Host On-Demand server switches or migrates data to the LDAP server, the password for the administrative ID used to do the switch will not be compromised.

8.6 Operational issues

This section covers some operational issues of which the administrator should be made aware.

8.6.1 Startup sequence

When the Service Manager starts, it reads the data store to get its configuration information. When Host On-Demand is using the private data store, this works very smoothly; however, when Host On-Demand is configured to use an LDAP directory, a dependency on the directory server is introduced. The directory server must be operational prior to the startup of the Service Manager so that it can read the data store from the directory server. If the LDAP directory server is not operational, the Service Manager will periodically retry the connection until it is successful. Until the Service Manager connects with the LDAP directory server, it will not accept any end user or administrative client connections. The symptom the user will see is a LOG0001 error message as shown in Figure 8-6.

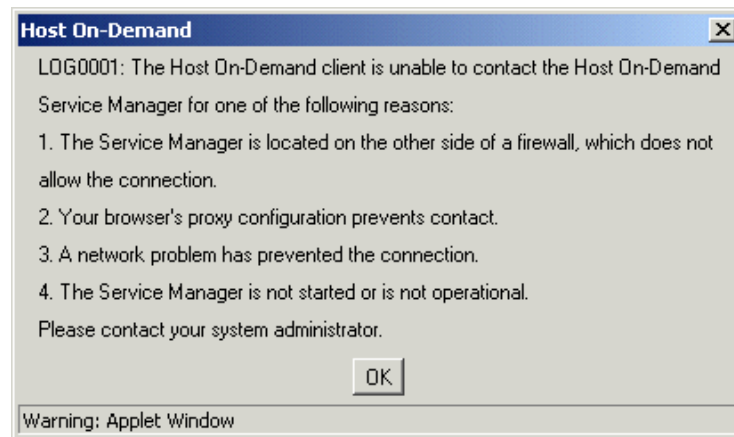


Figure 8-6 LOG0001 error

8.6.2 Reverting to the private data store if a directory server fails

If you must get your Host On-Demand server operational after an LDAP directory server failure, you can revert to the Host On-Demand Private data store. However, doing so means that you will not have access to any additions made to the LDAP directory server since you migrated from the private data store.

Because your Host On-Demand server is configured for LDAP directory server use, you must clear this condition and restart the Service Manager. To clear the condition, delete the `dirInfo.active` file in the `\hostondemand\private` directory, then restart the Service Manager. This clears the configuration of references to the directory server so that when the Service Manager starts, it will be using the private data store. After the LDAP directory server is again operational, you can follow the procedure in 7.4.1, “Use Directory Service (LDAP)” on page 359 to re-enable the LDAP directory support.

8.6.3 Debug Tracing of the Service Manager

There is a special trace facility for debugging problems with the Host On-Demand Service Manager. To enable this, add one of the following flags to the end of the command line that is used to start the Service Manager: `/d`, `/d1`, `/d2`, or `/d3`.

The `/d` flag turns on tracing. The `/dn` flag (where `n` is 1, 2 or 3) sets the debug level. Level 1 produces very little trace information, levels 2 and 3 produce progressively more information.

On Windows NT the Service Manager runs as a system service, so to use this trace facility, we recommend you stop the service and run the Service Manager from a command prompt. Example 8-1 shows a sample command file; modify it for your own installation.

Example 8-1 Sample debug command file

```
set PATH=%PATH%;c:\hostondemand\bin

c:\hostondemand\bin\jre.exe -mx20000000 -nojit -classpath
c:\hostondemand\lib\rt.jar;c:\hostondemand\lib\i18n.jar;c:\hostondemand\lib\ibm
jndi.jar;c:\hostondemand\lib\jndi.jar;c:\hostondemand\lib\jsdk.jar;c:\hostondem
and\lib;c:\hostondemand\lib\ods.jar;c:\hostondemand\lib\sm.zip
com.ibm.eNetwork.HODUtil.services.admin.NCServiceManager c:\hostondemand /d3 >
output.log
```

On OS/390, the `ServiceManager.sh` shell script found in `/hostondemand/lib` directory has a Java command statement commented out for the `/d3` tracing. Uncomment the statement that has the `/d3` trace and the output redirected into `/private/HOD.stdout`. Restart the Service Manager.

8.6.4 LDAP logs

Several logs are maintained by the Service Manager, some of which are present only if you use the LDAP directory server for the data store.

The `\hostondemand\private\server.log` contains LDAP operational messages for the current execution of the Service Manager. This file is overwritten every time the Service Manager starts the LDAP interface.

The `\hostondemand\private\hodldap.log` is introduced with the LDAP directory server. This file contains an audit trail of the migration from the private data store to the LDAP directory data store. It is especially useful when you migrate users that exist in multiple groups to an LDAP directory, because a user can only belong to a single group in the directory server data store. This log tells you into which group a specific user is migrated if it belonged to multiple groups in the private data store.

There are three subdirectories that are used, or created if they do not exist, when the LDAP interface is activated. Below is an explanation of what you will find in these logs:

- ▶ The `\hostondemand\private\TivoliLogs\` subdirectory contains three trace files, `trace1.log`, `trace2.log` and `trace3.log`, that track activity information on updates to the directory server. These three files are used in a round-robin fashion whenever the Service Manager starts (that is, `trace1` is used, then `trace2` the next time, then `trace3`, then `trace1` again).
- ▶ The `\hostondemand\private\serverlogs\` subdirectory contains a round-robin set of files that contain Java error messages relating to LDAP operations.
- ▶ The `\hostondemand\private\OpMgr\` subdirectory contains three logs, `UEvents1.log`, `UEvents2.log` and `UEvents3.log`, used in a round-robin fashion, that contain Java events regarding the LDAP operations.



Configuration Servlet

The traditional technique for retrieving and saving user preferences is for the Host On-Demand client to talk directly to the Host On-Demand Configuration Server via a predefined port, 8999 by default. Although efficient, it has two drawbacks when used in an environment that demands security:

1. It requires that the port be opened through a firewall.
2. The data between the client and the Configuration Server is not encrypted.

To resolve these issues, a servlet was added to tunnel the configuration information between the client and the servlet over an HTTP(S) connection, and then to pass that information on to the Host On-Demand Configuration Server of choice over the defined configuration port. This resolves both of the above-mentioned issues by using the existing HTTP(S) port already open through the firewall, and the encryption of the data by using HTTPS.

The implementation of the Configuration Servlet requires a Web server that supports servlets, such as WebSphere Application Server or Lotus Domino Go Webserver. There are many products that are capable of running the Configuration Servlet, and the configuration procedure for each is different. Check your Web server and servlet engine documentation for servlet configuration details on your operating system.

9.1 Installation

During Host On-Demand graphical installation for Windows NT or 2000 and for AIX, the Configuration Servlet may be automatically configured and installed if the installation program detects a supported Web application server installed. On Windows NT and Windows 2000, the recognized Web application servers include:

- IBM WebSphere Application Server Version 3.5
- IBM WebSphere Application Server Version 4.0
- Lotus Domino Go Webserver
- IBM Domino Go Webserver

On AIX, you can choose to have the Configuration Servlet installed and configured only if WebSphere Application Server Version 3.5 or 4.0 is detected.

For zSeries systems, the Configuration Servlet is installed with the Host On-Demand product; however, the WebSphere Application Server must be manually configured to use the servlet. The configuration of the zSeries Configuration Servlet is discussed in Chapter 3, “z/OS implementation” on page 79. Refer to 3.5, “Configuration Servlet setup” on page 99.

For iSeries the Configuration Servlet can be installed using a script. For details refer to 4.4, “Using the Configuration Servlet” on page 155.

Note: You must manually install the Host On-Demand Configuration Servlet in any environment where the servlet is not automatically installed and configured, or when the servlet is to be added after Host On-Demand is installed.

9.1.1 Manual installation

Manual installation will depend upon the specific Web application server that you are using. If you are installing on a zSeries machine, refer to 3.5, “Configuration Servlet setup” on page 99 for a sample scenario.

The following are the general steps required to configure the servlet on any Web application server.

1. Add `cfgsrvlt.jar` from the Host On-Demand installation's `lib` directory to the servlet engine's classpath, for example `d:\hostondemand\lib\cfgsrvlt.jar`. Refer to your Web server or servlet engine documentation for information about how

to do this. You can get a copy of `cfgsrvlt.jar` from the `\servlet` directory of the Host On-Demand CD, or from the `\hostondemand\lib` directory where you installed Host On-Demand on your server.

2. Add a servlet definition named “hodconfig” with a class name of `com.ibm.eNetwork.HODUtil.services.remote.HODCfgServlet`. The Host On-Demand default configuration uses “hodconfig” as the servlet name; however, any name will work as long as you are consistent. Refer to your Web server or servlet engine documentation for information about how to add a servlet definition.
3. Configure the servlet.

9.2 Configuring WebSphere Application Server 4.0

IBM WebSphere Host On-Demand automatically configures Websphere Application Server 4.0 on a windows platform. Use the Application Assembly Tool that is installed with Websphere Application Server to modify the default servlet configuration

Accessing Config Servlet

Start the Application Assembly Tool and choose to edit an existing application:

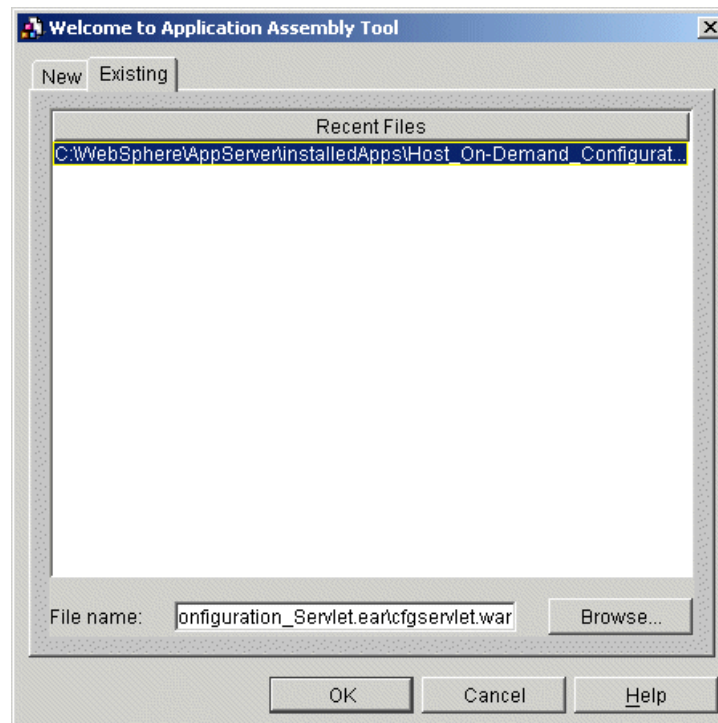


Figure 9-1 Modifying HODConfig Servlet with WAS 4.0

Now it is a simple process to choose the parameters associated with HODConfig

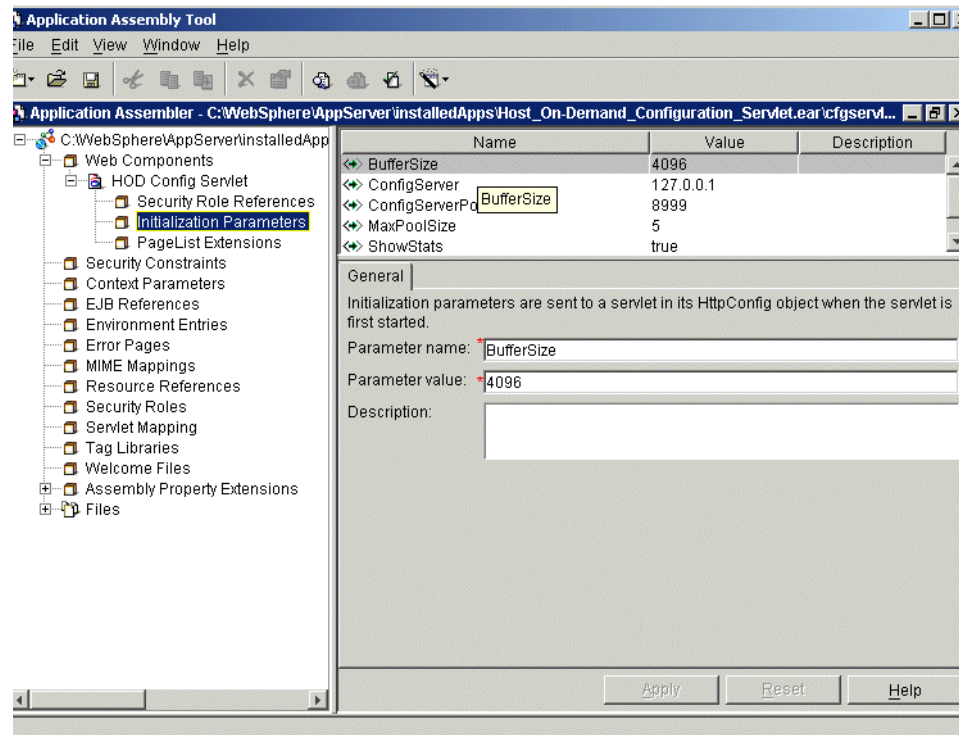


Figure 9-2 HODConfig Parameters with WAS 4.0

9.3 Configuring WebSphere Application Server 3.5

WebSphere Application Server installs a default servlet engine. We provide an example of how to install the Configuration Servlet using the graphical interface on a Windows platform, and how to install the servlet using the XMLConfig batch utility that was introduced with WebSphere Application Server V3.5. The graphical interface scenario will show how to install the Configuration Servlet running under the default servlet engine. For the scenario using the XMLConfig batch, we provide a sample that defines a new application to host the Configuration Servlet in addition to using the default application.

9.3.1 IBM WebSphere graphical configuration

Open the WebSphere Administrator's Console by clicking **Start -> Programs -> IBM WebSphere -> Application Server V3.5 -> Administrator's Console**. Once the Administrator's Console is up, you will see an icon that contains the name of your server. This is your node name. You must expand that tree by clicking the + sign and then expanding the Default Server and the Default Servlet Engine icons.

Set WebSphere alias

A WebSphere Application Server can provide a platform for multiple hosts. Each of these hosts is represented by a virtual host name and a list of one or more DNS aliases by which it is known. When a servlet request is made, the server name and port number component of the URL is compared to a list of all known aliases in an effort to locate the correct virtual host and serve the servlet. If no match is found, an error is returned to the browser. When no port number is specified in the URL, port 80 is assumed. If you will use any port other than port 80, including port 443 for HTTPS, you must add an alias statement with that port number specified.

There are several conditions that may not be obvious that will require you to add an alias:

- ▶ If your URL specifies a port number, then you must define an alias that includes the port number.
- ▶ If you will use HTTPS to connect with your WebSphere Application Server server, you must define an alias with the port number that HTTPS is using, even if you are using the default port of 443.
- ▶ If your Web server is host for multiple IP addresses, each IP address must have an alias and appropriate port number(s).

Let us illustrate with the setup shown in Figure 9-3. Assume the Web server has two network cards and two addresses (meaning two virtual hosts). The internal address is 9.24.105.38 (bigtex.itso.ral.ibm.com), and uses standard HTTP on port 80, while the external address is 205.223.100.15 and uses HTTPS on port 443.

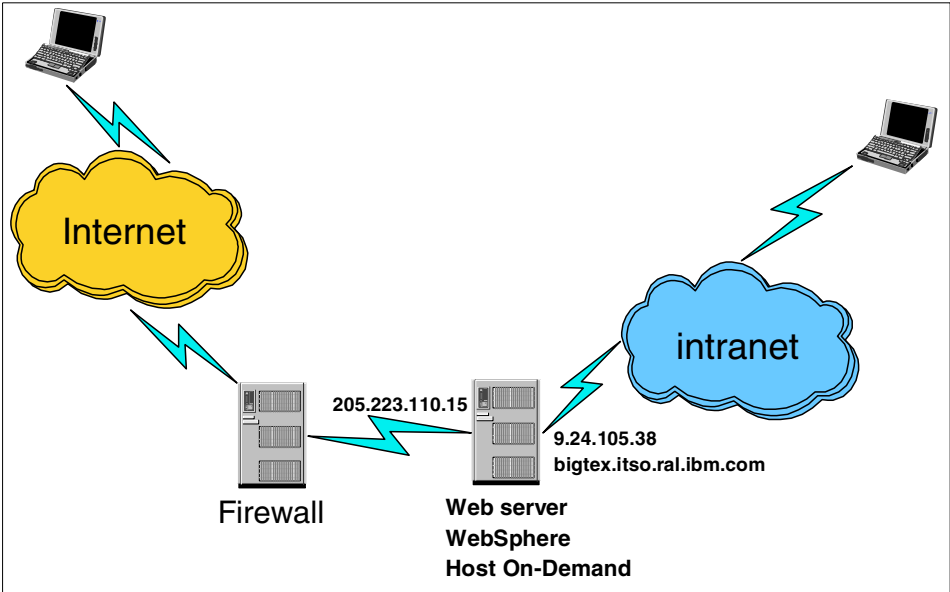


Figure 9-3 WebSphere alias environment

Table 9-1 illustrates the required alias rules.

Table 9-1 WebSphere alias examples

Reference URL	Required alias
http://127.0.0.1/servlet/HODConfig (usable only from the WebSphere machine)	127.0.0.1
http://localhost/servlet/HODConfig (usable from the WebSphere machine)	localhost
http://bigtex.itso.ral.ibm.com/servlet/HODConfig	bigtex.itso.ral.ibm.com
http://bigtex/servlet/HODConfig	bigtex
http://9.24.105.38/servlet/HODConfig	9.24.105.38
https://205.223.100.15/servlet/HODConfig	205.223.100.15:443

If the Web server is properly configured for all the connections and ports prior to the installation of the WebSphere Application Server, the WebSphere Application Server will add all the appropriate aliases. However, if anything changes, you must update the aliases manually.

To set the required aliases, you must first select **default_host** then select the **Advanced** tab as shown in Figure 9-4. Next, scroll down to an empty alias field and from here enter the required alias. Repeat this process until all aliases are entered, then click **Apply**.

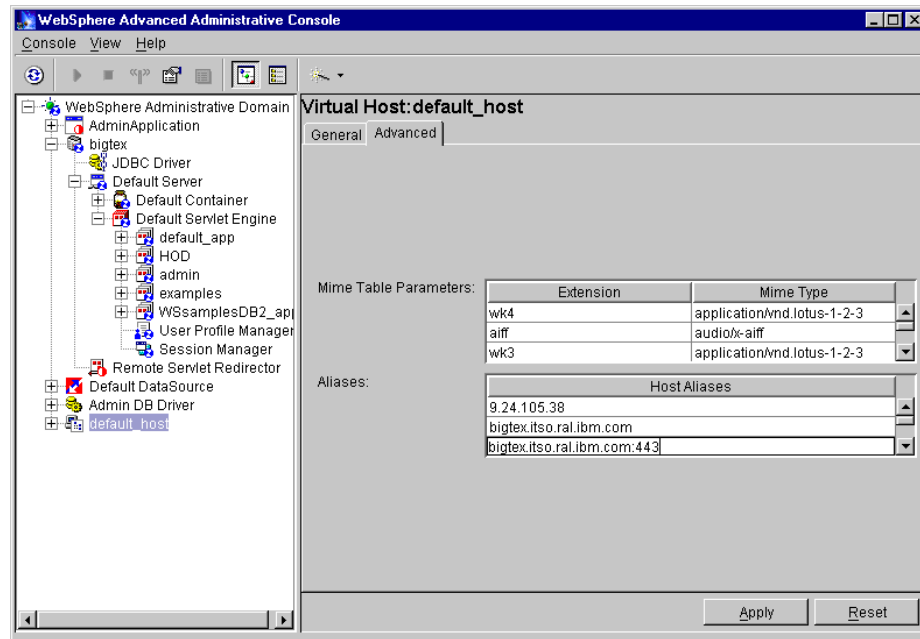


Figure 9-4 WebSphere default_app alias

Adding the classpath

You must now add an entry to the classpath for the application that hosts the Configuration Servlet, default_app in this example. Select the **default_app** entry in the left-hand pane, then select the **Advanced** tab from the resulting right-hand pane (see Figure 9-5).

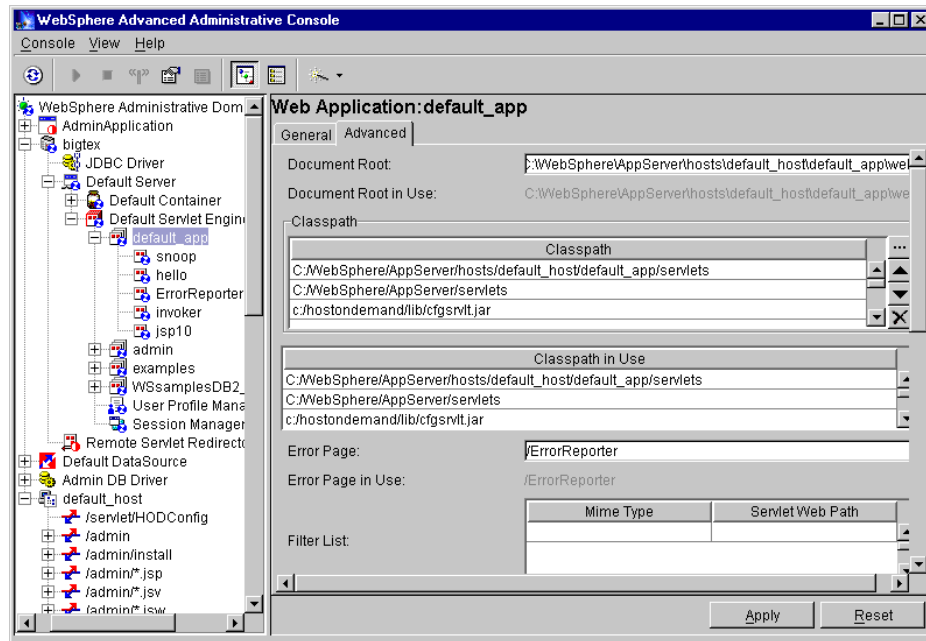


Figure 9-5 Select default application

You will see a frame called Classpath. This is where you will add the location of the cfgservlet.jar file. Select one of the empty entry boxes under Classpath and type the location of the cfgservlet.jar file, for example C:\hostondemand\lib\cfgservlet.jar. You must click the **Apply** button to update the classpath.

Adding the servlet

The next step is to add the Host On-Demand Configuration Servlet, so select the default application using the right mouse button to display the context menu. From that menu select **Create** to display the next context menu where you then select **Servlet** (see Figure 9-6).

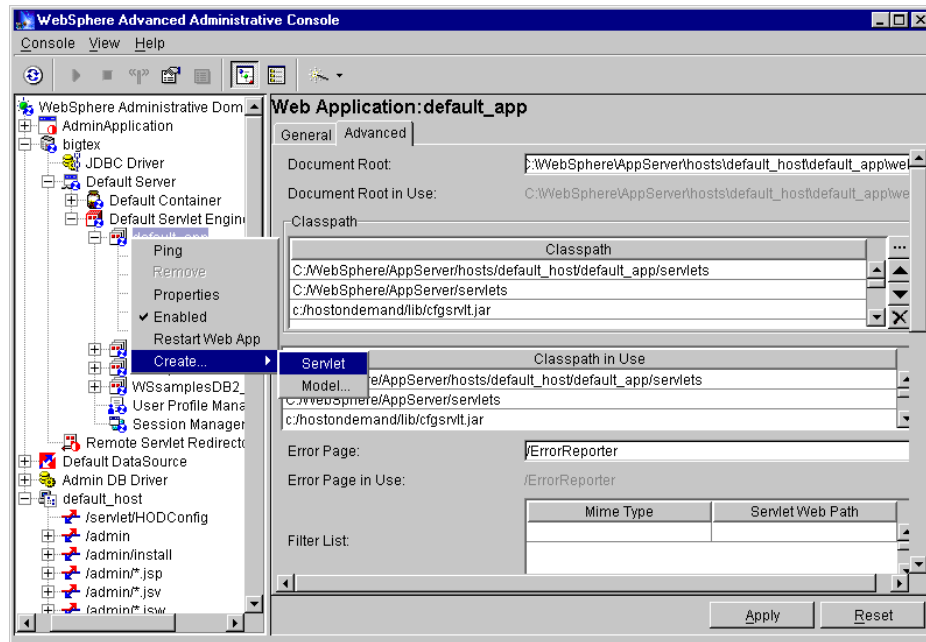
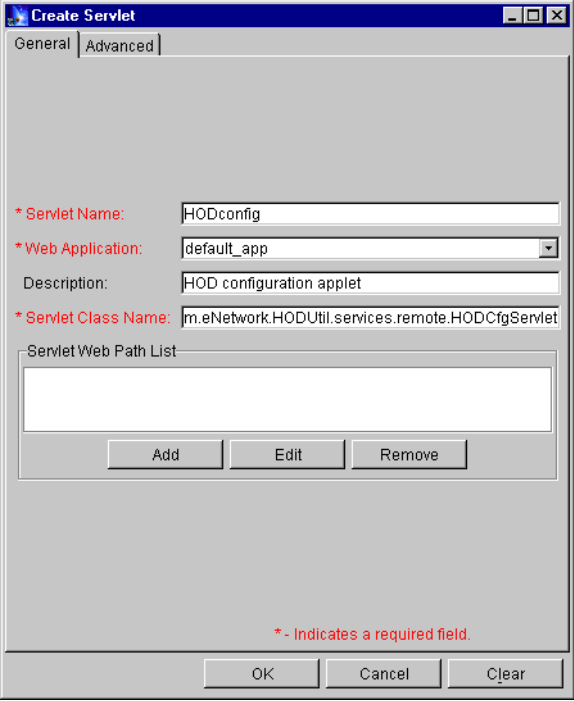


Figure 9-6 Create servlet, step 1

The resulting window is shown in Figure 9-7. On this window there are three required fields and one optional field:

- ▶ **Servlet Name:** the name of the servlet as it will be known to WebSphere (required).
- ▶ **Web Application:** the name of the Web application will be displayed (required).
- ▶ **Description:** a textual description of the servlet, for example Host On-Demand Configuration Servlet (optional, but recommended).
- ▶ **Servlet Class Name:** the full name of the class for this servlet. This is a required field and the value must be `com.ibm.eNetwork.HODUtil.services.remote.HODCfgServlet`
- ▶ **Servlet Web path list:** the string that will be used in the URL to identify the servlet.

The servlet name may be any name you wish to use in your URLs; we used HODConfig. The Description field is optional and is used only as comments. The Servlet Class Name field is critical and must be specified exactly. It is recommended that you cut and paste it directly from the help file. The value is `com.ibm.eNetwork.HODUtil.services.remote.HODCfgServlet`.



The "Create Servlet" dialog box has two tabs: "General" and "Advanced". The "General" tab is active. It contains the following fields:

- * Servlet Name:** HODconfig
- * Web Application:** default_app (dropdown menu)
- Description:** HOD configuration applet
- * Servlet Class Name:** m.eNetwork.HODUtil.services.remote.HODCfgServlet

Below these fields is a "Servlet Web Path List" section with an empty text box and three buttons: "Add", "Edit", and "Remove".

At the bottom of the dialog are three buttons: "OK", "Cancel", and "Clear".

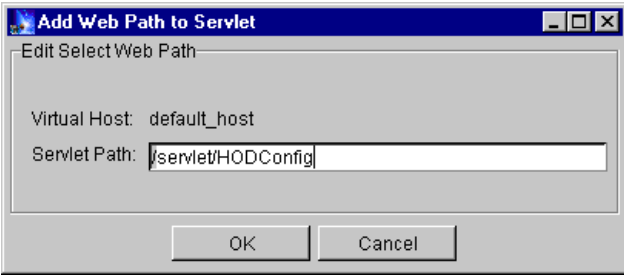
* - Indicates a required field.

Figure 9-7 Create servlet, step 2

Lastly, you must click the **Add** button, which will display the Add Web Path to Servlet window (see Figure 9-8). Here you enter the alias of this servlet, for example /servlet/HODConfig. Note the complete string:

/servlet/HODConfig

This will be the value that must be specified in the URL when accessing the Configuration Servlet. To exit this window, select the **OK** button to return to the previous window (Figure 9-7) where you must then click the **OK** button.



The "Add Web Path to Servlet" dialog box has a tab labeled "Edit Select Web Path". It contains the following fields:

- Virtual Host:** default_host
- Servlet Path:** /servlet/HODConfig

At the bottom of the dialog are two buttons: "OK" and "Cancel".

Figure 9-8 Create servlet, step 3

Upon returning to the main Create Servlet window, you must select the **Advanced** tab. This will display the window shown in Figure 9-9 into which you may specify the parameters as described in Table 9-2 on page 408. These parameters are optional. You need to specify them only if they differ from the defaults shown in Table 9-2. It is recommended that you specify at least the ConfigServer, ConfigServerPort, and the ShowStats parameters, as shown in Figure 9-9.

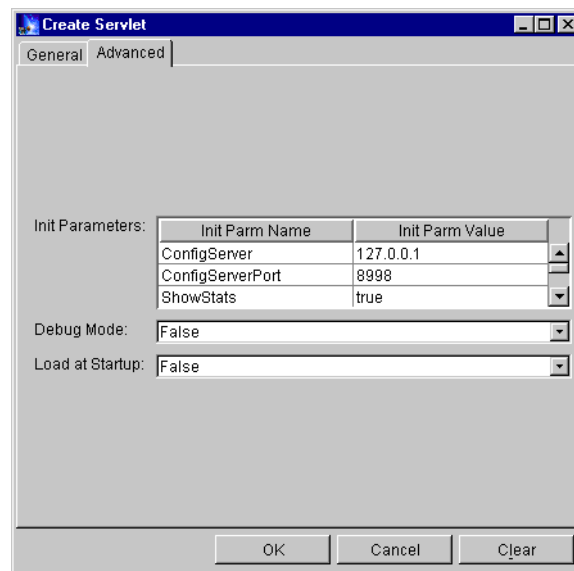


Figure 9-9 Servlet parameters

Specifying the ShowStats as true is recommended so that you can easily verify if the servlet is working properly before deploying the servlet.

Once all windows are completed, select the **Finished** button.

The Host On-Demand applets will recognize the following parameters that are specified in the definition of the Configuration Servlet.

Table 9-2 Configuration Servlet parameters

Parameter	Default Values	Description
ConfigServer	127.0.0.1	Host name or address of the Host On-Demand Configuration Server.

Parameter	Default Values	Description
ConfigServerPort	8999	Port Number of the Host On-Demand Configuration Server. This must match the port that the target Configuration Server is listening on.
Trace	false	When set to true, the Configuration Servlet writes servlet messages to the servlet engine log file, and to the browser when requested, for debugging purposes.
ShowStats	false	When set to true, allows the Configuration Servlet to return configuration information and statistics to browser requests. To invoke this option, specify <i>info</i> as the parameter passed to the applet. See "Testing the servlet" on page 410.
BufferSize	4096	Size of the buffer to use on buffered input or output streams.
PoolSize	5	Size of the buffer or socket pool to maintain. To turn off pooling, set PoolSize to 0.

Start the default server

To enable the servlet, the default server must be stopped and started. To stop the default server, highlight it and either click the **Stop** icon on the toolbar, or click the right mouse button to display the context menu and select **Stop**. You must wait until you receive the window shown in Figure 9-10, indicating that the server has stopped. This process could take some time.

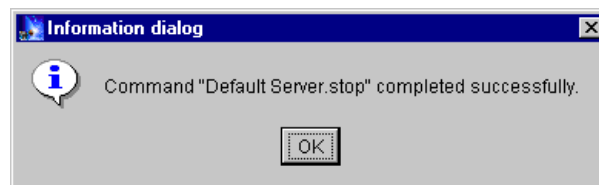


Figure 9-10 Default server stopped

Select **OK** to clear the information window, then start the server by selecting **Start** from the context menu of the default server. Again, you must wait for the information window (see Figure 9-11) for the process to complete.

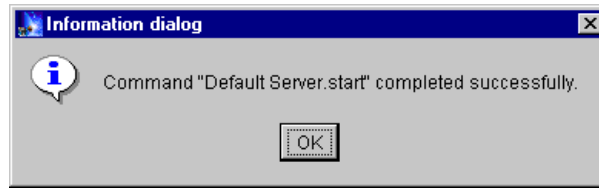


Figure 9-11 Default server started

Testing the servlet

After restarting the default server, it is recommended that you test the servlet by invoking the ShowStats function. This is done by specifying the following URL from a browser:

```
http://server_name/servlet_location/HODConfig/info
```

Using the example just created, the URL would look like the following:

```
http://server_name/servlet/HODConfig/info
```

When successful, your browser will return a window similar to that shown in Figure 9-12.

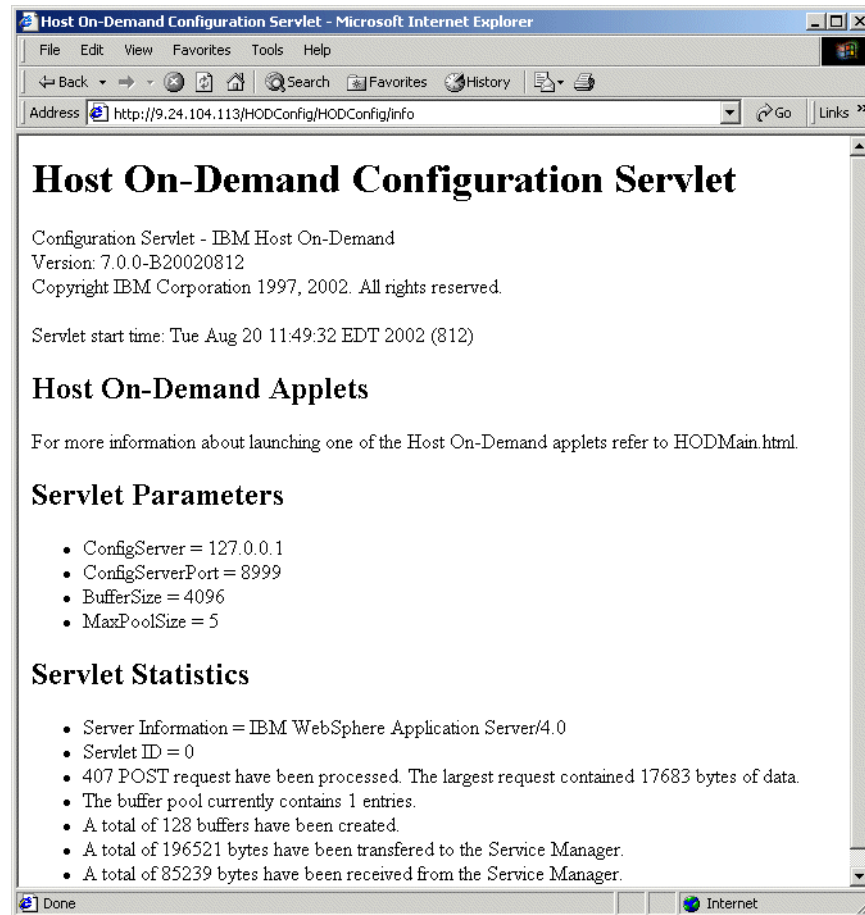


Figure 9-12 Servlet information

9.4 Enabling clients

There are two ways to enable a client to use the Configuration Servlet:

1. Set the ConfigServerURL parameter in the config.properties file. When this parameter is detected in the config.properties file, all clients will use this method of communication with the Host On-Demand Configuration Server.
2. Set the ConfigServerURL parameter in the HTML file used to launch the Host On-Demand client. This technique allows the administrator to specify which clients use the Configuration Servlet, such as external users, and which clients use the Configuration Server directly, such as internal users.

9.5 Referencing the Configuration Servlet

There are two ways to reference the Configuration Servlet, the direct reference and the indirect reference.

9.5.1 Direct reference

The direct reference is a complete URL. It includes the protocol, HTTP or HTTPS. For example, if you specify:

```
https://hodserver.raleigh.ibm.com/servlet/HODConfig/hod
```

You force the applet to use an encrypted HTTP connection to contact the Host On-Demand Configuration Servlet running on `hodserver.raleigh.ibm.com` over the default port 443. If this reference is used, the Configuration Servlet information will flow over an encrypted session even if the URL used to load the Host On-Demand client specified an unencrypted session, for example `http://hodserver.raleigh.ibm.com/hod/HOD.html`.

This technique may also be used to force the login to a machine other than the one used to load the client. Refer to 9.7, "Implementation scenarios" on page 417 for an example of how this may be used.

9.5.2 Indirect reference

An indirect reference specifies only a path name on the server that launched the Host On-Demand client. Using this method results in the `ConfigServerURL` being appended to the host portion of the Host On-Demand applet's URL. For example if the Configuration Servlet reference was:

```
/HOD/HODConfig/hod
```

and the Host On-Demand applet was loaded using the following URL:

```
https://hodserver/hod/HOD.html
```

then the resulting URL used to contact the Configuration Servlet would be:

```
https://hodserver/HOD/HODConfig/hod
```

This method is more flexible, allowing the reference to be used for HTTP and HTTPS connections from a single specification.

9.6 XMLConfig utility

If you are using WebSphere 3.5 there is a batch utility, XMLConfig, that may be used to add the Configuration Servlet. This utility is available on all platforms and is located in the \AppServer\bin directory. To use the utility you must create an XML file that defines the changes that you wish to implement. The general syntax is to invoke the utility:

```
XMLConfig -import filename.xml
```

A complete description of how to use the XMLConfig utility may be found in Chapter 21 of the *WebSphere V3.5 Handbook*, SG24-6161. The remainder of this section provides sample XML files to add a Configuration Servlet to the server.

9.6.1 Add Configuration Servlet to default_app

In this scenario we will configure the Configuration Servlet to run under the default_app. The objective is to define aliases to allow secure connections to the Configuration Servlet, and to add the Configuration Servlet under the default_app. The result will allow you to specify one of the following URLs to access the servlet:

- ▶ /servlet/HODConfig (a relative URL may be used with HTTP or HTTPS)
- ▶ http://bigtex.itso.ibm.com/servlet/HODConfig
- ▶ https://bigtex.itso.ibm.com/servlet/HODConfig

Note that even though the port number is not specified in the URL, it is still required in the definition.

The XML input file used in our example is shown in Example 9-1.

Example 9-1 Sample - add servlet to default application

```
<?xml version="1.0"?>
<!DOCTYPE websphere-sa-config SYSTEM
"$server_root$$dsep$bin$dsep$xmlconfig.dtd" >

<websphere-sa-config>
  <virtual-host name="default_host" action="update">

    <alias-list>
      <alias>localhost</alias>
      <alias>127.0.0.1</alias>
      <alias>bigtex</alias>
      <alias>bigtex.itso.ral.ibm.com</alias>
      <alias>9.24.105.38</alias>
      <alias>bigtex:443</alias>
```

```

        <alias>bigtex.itso.ral.ibm.com:443</alias>
<alias>9.24.105.38:443</alias>
    </alias-list>
</virtual-host>
<node name="bigtex" action="locate">
    <application-server name="Default Server" action="locate">
        <servlet-engine name="Default Servlet Engine" action="locate">
            <web-application name="HOD" action="create">
                <description>Host On-Demand</description>

<document-root>C:\WebSphere\AppServer\hosts\default_host\HOD\web</document-root
>

        <classpath>
            <path
value="C:/WebSphere/AppServer/hosts/default_host/HOD/servlets"/>
            <path value="c:/hostondemand/lib/cfgsrvlt.jar"/>
        </classpath>
        <error-page>/ErrorReporter</error-page>
        <filter-list/>
        <group-attributes/>
        <auto-reload>true</auto-reload>
        <reload-interval>9000</reload-interval>
        <enabled>true</enabled>
        <root-uri>default_host/</root-uri>
        <shared-context>false</shared-context>
        <shared-context-jndi-name>SrdSrvltCtxHome</shared-context-jndi-name>

    <servlet name="HODConfig" action="create">
        <description>Configuration Servlet</description>
        <code>com.ibm.eNetwork.HODUtil.services.remote.HODCfgServlet</code>
        <init-parameters>
            <parameter name="ConfigServerPort" value="8998"/>
            <parameter name="ShowStats" value="true"/>
            <parameter name="Trace" value="true"/>
            <parameter name="ConfigServer" value="127.0.0.1"/>
        </init-parameters>
        <load-at-startup>false</load-at-startup>
        <debug-mode>false</debug-mode>
        <uri-paths>
            <uri value="/HODConfig"/>
            <uri value="/hodconfig"/>
        </uri-paths>
        <enabled>true</enabled>
    </servlet>

</web-application>
</servlet-engine>
</application-server>
</node>

```

```
</websphere-sa-config>
```

Note: It is recommended that you export the existing WebSphere Application Server definition prior to beginning, then copy the alias virtual host definitions from the exported file and paste them into your new file and add any additional aliases you require. In our lab environment we discovered that when the XMLConfig utility was run and virtual host aliases were present, XMLConfig did replace the existing definitions with the one specified in the deck. It did not do an update. By default, if an action is updated and the item does not exist, then it will be created.

9.6.2 Add Configuration Servlet to new application

In this scenario we will define a new application, HOD, to run under the Default Servlet Engine, and to define the Configuration Servlet to run under the new application, HOD. We also define aliases to allow secure connections to the Configuration Servlet as we did in the scenario described in 9.6.1, “Add Configuration Servlet to default_app” on page 413. The result will be the same except that the URL will be one of the following:

- ▶ /HOD/HODConfig
- ▶ http://bigtex.itso.ral.ibm.com/HOD/HODConfig
- ▶ https://bigtex.itso.ral.ibm.com/HOD/HODConfig

The procedure is similar to the scenario described in 9.6.1, “Add Configuration Servlet to default_app” on page 413.

The XML file used for this example is shown in Example 9-2.

Example 9-2 Sample - add servlet to new applicationS

```
<?xml version="1.0"?>
<!DOCTYPE websphere-sa-config SYSTEM
"$server_root$$dsep$bin$dsep$xmlconfig.dtd" >

<websphere-sa-config>
  <virtual-host name="default_host" action="update">

    <alias-list>
      <alias>localhost</alias>
      <alias>127.0.0.1</alias>
      <alias>bigtex</alias>
      <alias>bigtex.itso.ral.ibm.com</alias>
    </alias-list>
  </virtual-host>
</websphere-sa-config>
```

```

        <alias>127.0.0.1:443</alias>
        <alias>bigtex:443</alias>
        <alias>bigtex.itso.ral.ibm.com:443</alias>
        <alias>9.24.105.38:443</alias>
    </alias-list>
</virtual-host>
<node name="bigtex" action="locate">
    <application-server name="Default Server" action="locate">
        <servlet-engine name="Default Servlet Engine" action="locate">
            <web-application name="HOD" action="update">
                <description>Host On-Demand</description>

<document-root>C:\WebSphere\AppServer\hosts\default_host\HOD\web</document-root
>
        <classpath>
            <path
value="C:/WebSphere/AppServer/hosts/default_host/HOD/servlets"/>
            <path value="c:/hostondemand/lib/cfgsrvlt.jar"/>
        </classpath>
        <error-page></error-page>
        <filter-list/>
        <group-attributes/>
        <auto-reload>true</auto-reload>
        <reload-interval>9000</reload-interval>
        <enabled>true</enabled>
        <root-uri>default_host/HOD</root-uri>
        <shared-context>false</shared-context>
        <shared-context-jndi-name>SrdSrvltCtxHome</shared-context-jndi-name>
        <servlet name="HODConfig" action="update">
            <description>HOD Configuration Servlet</description>
            <code>com.ibm.eNetwork.HODUtil.services.remote.HODCfgServlet</code>
            <init-parameters>
                <parameter name="ConfigServerPort" value="8998"/>
                <parameter name="ShowStats" value="true"/>
                <parameter name="ConfigServer" value="127.0.0.1"/>
                <parameter name="Trace" value="true"/>
            </init-parameters>
            <load-at-startup>false</load-at-startup>
            <debug-mode>false</debug-mode>
            <uri-paths>
                <uri value="/HODConfig"/>
                <uri value="/hodconfig"/>
            </uri-paths>
            <enabled>true</enabled>
        </servlet>
    </web-application>
</servlet-engine>
</application-server>
</node>

```

</websphere-sa-config>

9.7 Implementation scenarios

In addition to the obvious use with firewalls, the Configuration Servlet opens other new ways to deploy Host On-Demand and solve some very difficult issues. We will explore only two: load balancing and Native Authentication.

9.7.1 Load balancing

Let us assume that a company wants to deploy Host On-Demand using a redundant, highly available solution, and also wishes to use the registered user model. In prior releases only one option was available for them: to implement an LDAP directory server to house all user IDs and preferences for all servers, thus providing centralized management. By using the Configuration Servlet, an additional option becomes available: deploy the Configuration Servlet and route all login requests to a central system, such as the zSeries, and maintain the information in the Host On-Demand native data store.

In this example the customer would have two or more Web servers configured in a redundant load balancing configuration, or two or more servers in physically separate locations providing alternate access points. In either case, the servers would be configured identically with the following components:

- ▶ A Web server
- ▶ A Host On-Demand server
- ▶ A Configuration Servlet running on a Web server that support servlets, Lotus Domino Go Webserver, WebSphere Application Server, or some other Web application server. The servlet would be configured to route all requests to a centralized third server. See Figure 9-12 on page 411.

The distributed Host On-Demand systems would not be configured to accept client login requests. Instead they would deploy the Configuration Servlet, which would route the login requests to the OS/390-based Host On-Demand system, or some other system that processes user login requests. The advantage of this scenario is that all the distributed Host On-Demand servers could be exact clones of one another, and an LDAP directory server would not be required, while still allowing all login processing to be centralized. The remote Host On-Demand servers would be optimized for Web serving exclusively and any platform, or combination of platforms, could be used.

9.7.2 Native Authentication

Let's assume the same scenario as described above, but now add the requirement that all users must use their RACF user ID and password. The only modification to the previous scenario would be to deploy Native Authentication on the zSeries system, refer to 3.9, "Native Authentication" on page 140. Refer to 11.11, "Native Authentication" on page 468 for details on how to configure that system. The result would be that regardless of the platform(s) chosen to deploy distributed Host On-Demand servers, all users logging in would do so on the same system.

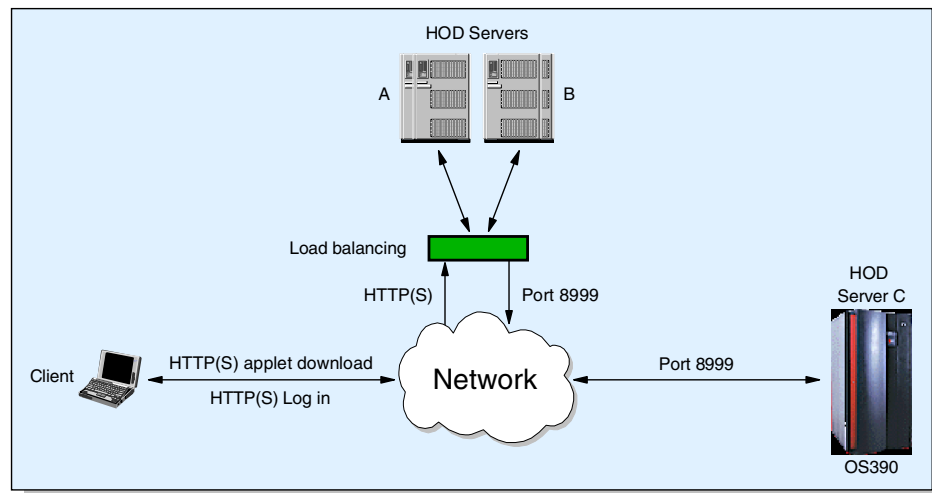


Figure 9-13 Servlet with Native Authentication

9.8 Problem determination

You can access trace, configuration and statistic information from the Configuration Servlet for debugging purposes. To access trace information, you need to set the Trace parameter to true for the Configuration Servlet. To view the trace, load the following URL into your browser:

```
http://server_name/servlet_alias/HODConfig/trace
```

The Configuration Servlet's trace information will be displayed in the browser and written to the servlet engine's log file.



OS/400 Proxy

If you are planning to provide file transfer or Database On-Demand to your iSeries (AS/400) users, you may wish to consider using the OS/400 Proxy feature. This feature is only appropriate for connections to an AS/400. It is called a “proxy” because connections are made from the workstation to the OS/400 Proxy server, which in turn connects to the target iSeries system. The connection will be completed only if the user enters a valid user ID and password on the target iSeries system.

If you will be accessing multiple iSeries systems from the Internet you may use the OS/400 Proxy to reduce the number of Internet addresses for each target system. If used in conjunction with the Redirector feature, only one address needs to be Internet addressable for multiple back-end systems. In addition, only one port needs to be opened on the firewall (typically port 3470) for the File Transfer and Database On-Demand features. The typical ports for file transfer (like ports 20 and 21) can be blocked on the firewall to prohibit direct access.

You may optionally encrypt the connection from the proxy to the back-end host system.

In this chapter, we will discuss:

- ▶ How to configure a simple session
- ▶ Enabling SSL
- ▶ How to use the proxy with Database On-Demand
- ▶ Sample firewall rules

10.1 How to configure a simple session

In our sample configuration (see Figure 10-1), we are using server C as the Host On-Demand server, the Redirector and the OS/400 Proxy server. In practice, each of the services could be split among multiple computers.

One other key concept is that the OS/400 proxy server (C) does not necessarily have to be an iSeries. In the example below, we used a Windows 2000 server.

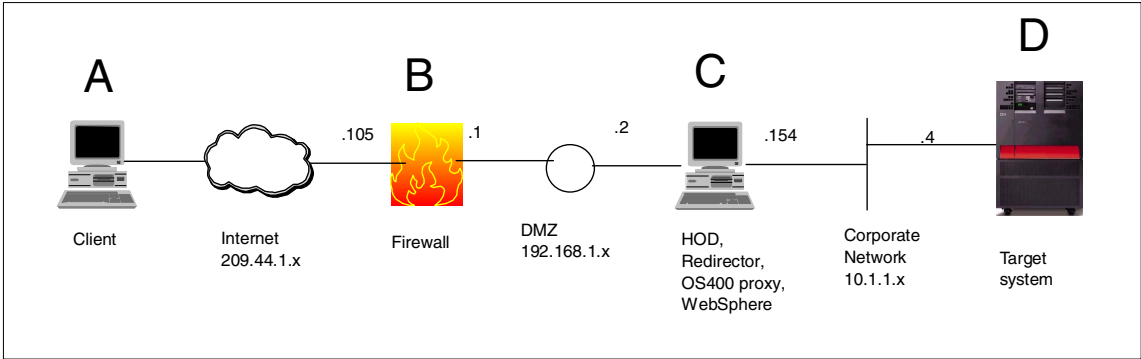


Figure 10-1 Our sample OS/400 Proxy network

The firewall will typically be configured to map a network address using Network Address Translation (NAT).

Table 10-1 Sample NAT rule

System	Internet address	Local address
HOD server	209.44.1.106	192.168.1.2

Figure 10-2 illustrates how you may select the port you wish to use for the proxy server.

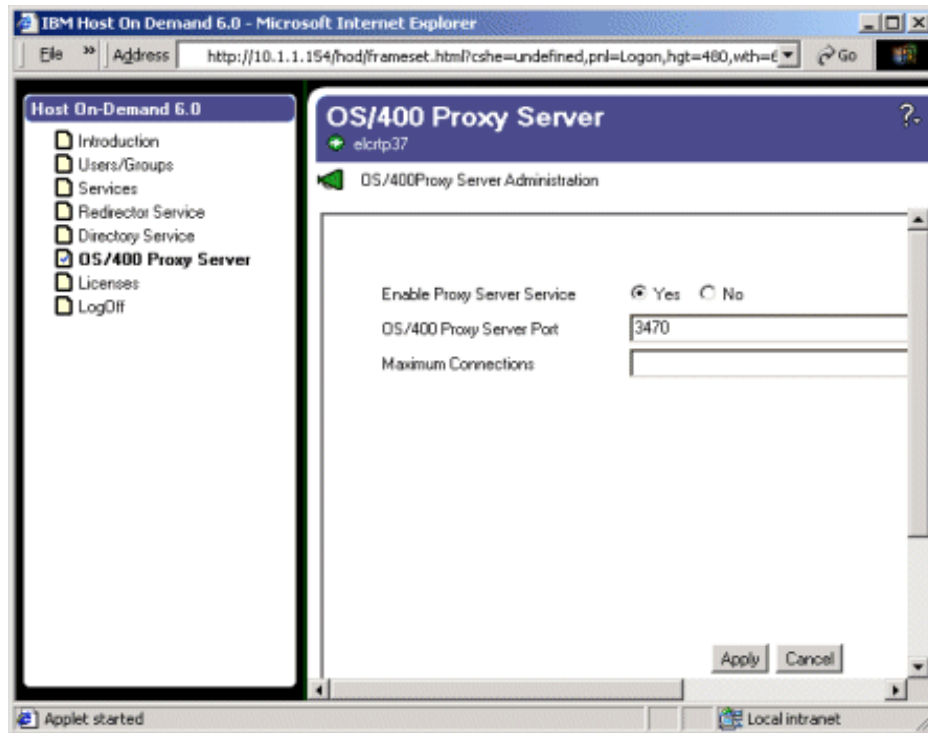


Figure 10-2 OS/400 proxy server window within HODAdmin.html

Note: The OS/400 Proxy does not support Telnet connections and the Redirector does not support non-Telnet connections. Thus, they are typically used together.

Figure 10-3 illustrates a sample redirected session. The session uses the Redirector (C in Figure 10-1) to get to the target iSeries (D). Click **File Transfer Defaults**.

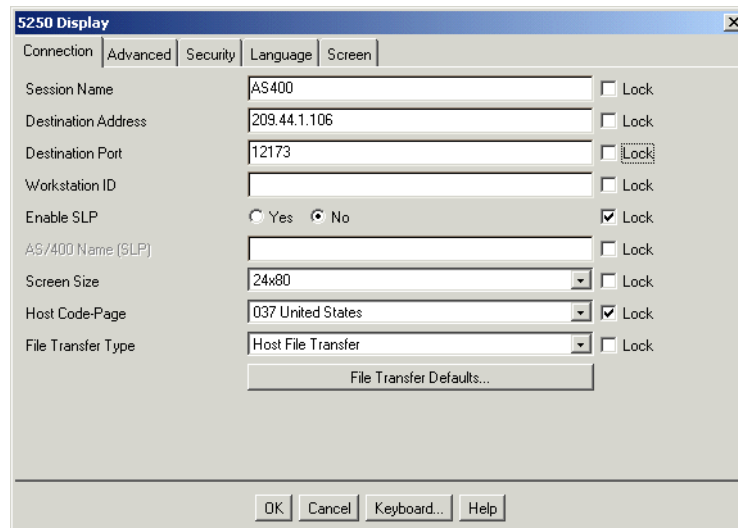


Figure 10-3 Sample session using Redirector and OS/400 Proxy

Important: The File Transfer Destination Address field (see Figure 10-4) is resolved by the OS/400 Proxy server (C in Figure 10-1). If you leave it blank, it will default to the same value as the Telnet session. Since we are using the Redirector, you should specify the address (or URL) of the target iSeries (D). Specify the address (or URL) of the target iSeries (D). The OS/400 Proxy server (C) will need to have access to the corporate DNS or to have the addresses in its local hosts table.

Click **Yes** for Enable Proxy Server.

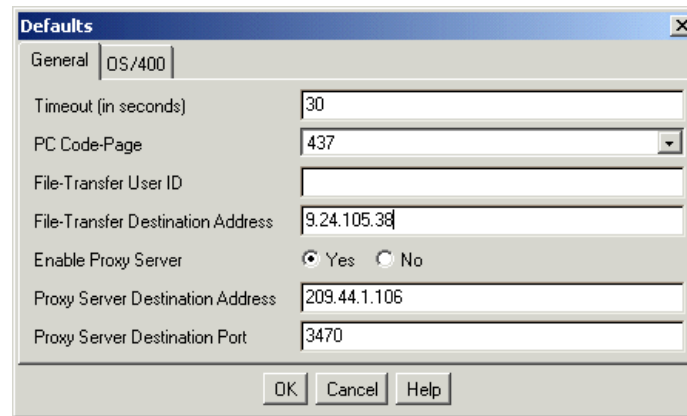


Figure 10-4 File Transfer defaults for the session

This concludes the configuration portion. Let's see what the connection looks like in action.

10.2 Using the OS/400 Proxy

There are two components that can utilize the OS/400 Proxy:

- ▶ The 5250 file transfer
- ▶ Database On-Demand

10.3 Enabling SSL

The following section shows how to make the connection between the proxy and the iSeries system secure.

10.3.1 Prerequisites

The following is required on each target iSeries and iSeries system:

- ▶ OS/400 V4R4, or later
- ▶ OS/400 Host Servers (5722-SS1 or 5769-SS1, option 12)
- ▶ Digital Certificate Manager (5722-SS1 or 5769-SS1, option 34)
- ▶ A Cryptographic Access Provider product. You can choose from the following licensed products: 5769-AC1 (40-bit), 5722-AC2 or 5769-AC2 (56-bit), 5722-AC3 or 5769-AC3 (128-bit). The bit size for these products indicates the

varying sizes of the digital keys that they employ. A higher bit size results in a more secure connection. Some of these products are not available in all areas due to government export regulations.

- Client Encryption. You may choose from one or more of the following licensed products: 5769-CE1 (40-bit), 5722-CE2 or 5769-CE2 (56-bit), 5722-CE3 or 5769-CE3 (128-bit).

Note: To help you to meet the SSL legal responsibilities, you must change the authority of the directory that contains the SSL files to control user access to the files. In order to change the authority, you must follow the steps below:

1. Enter the command: `wrk1nk '/QIBM/ProdData/HTTP/Public/jt400/*'`
2. Select option 9 in the directory (SSL40, SSL56, or SSL128)
3. Ensure *PUBLIC has *EXCLUDE authority.
4. Give users who need access to the SSL files *RX authority to the directory. You can authorize individual users or groups of users.

10.3.2 Configure each target iSeries

The following steps are required on each target iSeries (AS/400) system:

1. From a Web browser enter `http://<server.name>:2001` (where <server.name> is the host name of your iSeries). If you are unable to connect, start the HTTP server with the following OS/400 command:

```
STRTCPSVR SERVER(*HTTP) HTTPSVR(*ADMIN)
```

2. Enter a iSeries user profile and password (when prompted). You must have *ALLOBJ authority to complete the configuration activities below.
3. Click **Digital Certificate Manager**
4. Click **System Certificates**
5. Click **Work with Secure Applications**
6. Click **QIBM_OS400_QZBS_SVR_CENTRAL**, then click **Work with System Certificate**
7. Verify that the *DFTSVR certificate is selected and click **Assign New Certificate**.
8. Repeat steps 6 and 7 for the following applications:
 - QIBM_OS400_QZBS_SVR_DATABASE
 - QIBM_OS400_QZBS_SVR_DTAQ
 - QIBM_OS400_QZBS_SVR_NETPRT

- QIBM_OS400_QZBS_SVR_RMTCMD
- QIBM_OS400_QZBS_SVR_SIGNON
- QIBM_OS400_QZBS_SVR_FILE
- QIBM_OS400_QRW_SVR_DDM_DRDA

10.3.3 Configuring the OS/400 Proxy keyring

The following instructions are required only on the Host On-Demand server:

1. Type the following OS/400 command:

```
QSH
```

2. Type the following command (note cd must be in lower case):

```
cd /qibm/proddata/hostondemand/lib
```

3. Check to see if some directories exist (if they don't exist, they will be created; if they already exist, you will get a message):

```
mkdir com/ibm/as400
```

4. Check for an additional directory:

```
mkdir com/ibm/as400/access
```

5. The following command obtains a certificate from the SSL-enabled sign-on server (**Note:** <server.name> is the host name of your iSeries). Port 9476 is the commonly used port for the "Sign-on" Host Server. This command is in actuality a single line.

```
java -classpath ./QIBM/ProdData/hostondemand/lib/sm.zip
com.ibm.hodsslght.tools.keyrng com.ibm.as400.access.KeyRing connect
<server.name>:9476
```

Restriction: You must enter toolbox as the password (*in lowercase*).

6. Multiple pages of information may be displayed; press the Page-up and Page-down keys to see additional details about the certificates, including the fingerprint. You will typically have two selections to choose from:

- 0 = Use the Server Certificate
- 1 = Use the Certificate Authority (CA)

Always select **0** to trust the server certificate, then press Enter.

7. Repeat steps 1-6 for each target iSeries server.
8. Perform the final step:

```
cp com/ibm/as400/access/KeyRing.class ../hod/com/ibm/as400/access
```

10.4 Firewall rules for OS/400 Proxy

Table 10-2 identifies the firewall ports that must be opened if the OS/400 Proxy is used. The key point is that OS/400 Proxy does not negate the need for HTTP and Telnet services.

Table 10-2 Firewall rules for Host On-Demand with OS/400 Proxy

Host On-Demand Function	Firewall Port(s) used	Secure Firewall Port(s) used
3270 and 5250 Display and Printer Emulation	23 (Telnet) 80 (HTTP)	992 (Telnet) 443 (HTTPS)
File transfer, Database On-Demand	3470 (Proxy Tunnel)	3471 (Proxy Tunnel)
Host On-Demand Administration	80 (HTTP) 8999 (config server) ¹	443 (HTTPS)
License Use Count License Use Management (LUM)	80 (HTTP) 8999 (config server) ²	
¹ If used in conjunction with the WebSphere Configuration Servlet, port 8999 is not required. Refer to Chapter 9, "Configuration Servlet" on page 397. ² To disable License Use Management, refer to 7.6.2, "Disabling License-Use Count" on page 365.		



Security

Host On-Demand is primarily a downloaded application that obtains the session configuration information from the Web server. This configuration information consists of an IP address and port to access the host system. If you are using a registered user model, you must also have a user ID and optionally a password that will be used to obtain the configuration information from the server. If you are using an anonymous user model, this information is provided as part of the HTML download process. Finally, host systems also require a user ID and password to log on.

Unless your Web server is configured for SSL (HTTPS), the login and the transfer of the HTML data is not encrypted, and could be read by a third party. If your users are accessing Host On-Demand and host data from within your intranet, this default security setup might be enough.

If you have users on the Internet accessing Host On-Demand and data on your intranet, you may want additional security. You can configure your Web server to use HTTPS so that the data sent to your browser is encrypted. See your Web server documentation for more information about configuring for HTTPS.

Once the client is loaded in a browser, it communicates directly with the host. The configuration information the Configuration Server sends to the client regarding the sessions, such as IP address, port number, and user preferences, is not encrypted, unless you have implemented the Configuration Servlet and utilize HTTPS.

If the Telnet server supports SSL, the clients can be configured to use SSL also. See your Telnet server's documentation for more information about configuring SSL on the Telnet server, and see the Host On-Demand online help for more information about configuring a client to connect to a secure Telnet server.

Using Secure Sockets Layer (SSL) with Host On-Demand extends secure host data access across intranets, extranets, and the Internet. Mobile workers can access a secure Web site, receive authentication and establish communication with a secure enterprise host. With client and server certificate support, Host On-Demand can present a digital certificate (X.509, Version 3) to the Telnet server, such as IBM Communications Server for Windows NT Version 6, Communications Server for AIX Version 6, OS/400 V4R4 and higher or IBM Communications Server for OS/390 Version 2.6, or later, for authentication.

Host On-Demand can also integrate the SSL client authentication with IBM Vault Registry, providing you with the benefit of using industry-standard public key infrastructure (PKI) methods.

If your Telnet server does not support SSL, and you are running Host On-Demand on Windows NT, Windows 2000 or AIX, you can configure the Host On-Demand Redirector to provide SSL support. The Redirector acts as a transparent proxy between the client and the Telnet server by using port remapping. It can encrypt data between the client and itself, between itself and the host, or both. Refer to the online documentation for instructions on how to configure the Redirector or 7.3.1, "Configuring the Redirector" on page 351.

11.1 Signed applet support

The original Java security model prevented a Java applet from:

- ▶ Communicating with servers other than the one from which it had originated
- ▶ Accessing system resources such as hard disks, printers and the clipboard

These constraints are often referred to as the sand box. Their purpose was to:

- ▶ Prevent an applet from causing harm on the Internet, which it might be able to do if it were allowed to connect to any destination
- ▶ Prevent an applet from doing harm to the machine to which it was downloaded

This was found to be too restrictive in practice, and Java Development Kit (JDK) Version 1.1 introduced the notion of a signed or trusted applet. Such an applet has an embedded X.509 certificate, which identifies the creator of the applet. A user can instruct the browser to allow certain signed applets to operate outside of the sand box.

To sign an applet, the developer must first obtain a certificate from a Certificate Authority (CA). He can then sign his applet with a special signing tool, which embeds the certificate in the file that contains the applet code. There will usually be two of these: a JAR file for use by Netscape and a CAB file for use by Internet Explorer.

Browsers are preconfigured with public-key certificates from well-known CAs such as VeriSign. When a browser encounters a signed applet from a new source, it checks the embedded certificate to see if it has been signed by one of its preconfigured CAs. If it has, the browser tells the user who the developer is and asks if he trusts the applet (and whether the decision is to be remembered). It also asks if all applets from that developer are to be trusted. If the user agrees, the applet continues to load.

This is a much-simplified description of signed applet security. The following Web sites contain further details:

http://www.suitable.com/Doc_CodeSigning.shtml

<http://developer.netscape.com/docs/manuals/signedobj/trust/index.htm>

11.2 Host On-Demand SSL support

Host On-Demand can ensure the privacy of communications through the use of the Secure Sockets Layer (SSL) Protocol when connecting to SSL-capable Telnet servers. Host On-Demand implements SSL Version 3 to provide message privacy and integrity. This section describes how Host On-Demand has implemented SSL.

The key part of SSL negotiation is the client's ability to trust the certificate presented by the server, and the server's ability to trust the certificate presented by the client.

For Host On-Demand clients the public certificates of the trusted CAs are stored in one of three places:

1. WellKnownTrustedCAs.class file
2. CustomizedCAs.class file
3. Microsoft cryptographic database

Host On-Demand now has two places to look for the client certificate if required:

1. A password-protected PKCS12 file accessed via the local file system, or in a URL.

This is the same level of support as was provided by Host On-Demand Version 4 and the initial release of Host On-Demand Version 5. The URL protocols you can use depend on the capabilities of your browser. Most browsers support HTTP, HTTPS, FTP, and FTPS.

2. A client certificate accessible through the Microsoft cryptographic API (CAPI).

The Microsoft cryptographic API is the security interface used by Internet Explorer to access its certificates, client or server, and is available only on Windows platforms. Microsoft introduced this API with Internet Explorer Version 5.

11.2.1 Java class files

There are two key Java class files that are used by the Host On-Demand emulation clients when negotiating SSL sessions with Telnet servers: WellKnownTrustedCAs.class and CustomizedCAs.class. The WellKnownTrustedCAs.class file contains the public certificates of all the CAs that Host On-Demand trusts. The CustomizedCAs.class file contains the certificates of unknown CAs and self-signed certificates.

The WellKnownTrustedCAs.class file is supplied by Host On-Demand and is not to be modified by the customer. If a self-signed certificate or a certificate from a unknown authority (CA) is to be used, the CustomizedCAs.class must be created or updated by the customer.

Both the WellKnownTrustedCAs.class file and the CustomizedCAs.class files are stored in the publish directory. All Host On-Demand download clients, including cached clients, obtain or refresh these files from the server when the applet is loaded.

Locally installed clients have the WellKnownTrustedCAs.class file installed on the workstation during product installation. A CustomizedCAs.class file is not installed by default, so if a locally installed client requires a certificate from a unknown CA or self-signed certificate, it must be created. The recommended method is to send the certificate to the client and have the user create the CustomizedCAs.class file at the client. Refer to 12.2.4, “Making server certificates available to clients” on page 490. To create the CustomizedCAs.class file on the Host On-Demand server on OS/390, refer to “Creating the CustomizedCAs.class file on the server” on page 119.

11.2.2 Microsoft cryptographic service provider database

Starting with Version 5.03, Host On-Demand provided an enhancement that allows the administrator to enable Host On-Demand to use the cryptographic API interface to store client certificates and public key certificates for CAs into the Microsoft cryptographic service provider database, hereafter referred to as the cryptographic database. This function has been tested and is supported on the following platforms:

- ▶ Windows 98
- ▶ Windows NT 4.0
- ▶ Windows 2000
- ▶ Windows Millennium Edition

The use of this interface provides simplification and usability improvements. All user prompting and card access for client authentication can be performed by the CAPI software. When selected, the Host On-Demand client receives a list of available client certificates and security providers, then presents the list to the user for selection to send to the server. As long as the Microsoft cryptographic database is installed, the option is available on both Netscape and Internet Explorer browsers, and is the preferred interface for Microsoft Internet Explorer.

Through the use of the Microsoft cryptographic service provider database, not only do you have access to the client certificates, but to the CA certificates trusted by the browser as well. Therefore, if the Telnet server is using a certificate by a CA unknown to Host On-Demand, but known to the cryptographic database, then you can use the certificate located in the cryptographic database, thus eliminating the need to add the signer certificate to the CustomizedCAs.class file.

Many of the smart card readers are CAPI-compliant. By leaving hardware-level smart card processing to the CAPI and vendor interfaces, IBM is able to support new security devices without changing the Host On-Demand code. For instance, if a new thumbprint reader device becomes available, Host On-Demand will be able to access it through the use of the CAPI or the vendor interfaces without realizing it is not a smart card.

Viewing certificates

Certificates that are registered in the cryptographic database can be displayed in the following way:

1. Start the Internet Explorer 5.x browser.
2. Select **Tools -> Internet Options**.
3. Select the **Content** tab in the Internet Options window, as shown in Figure 11-1.

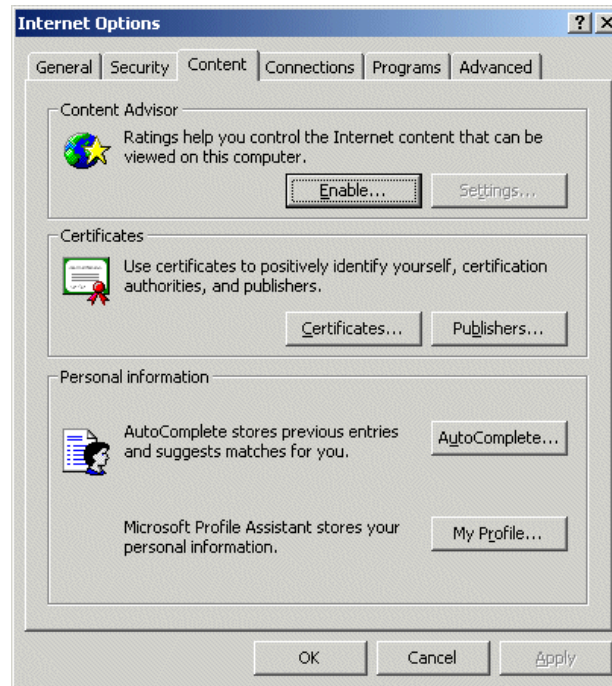


Figure 11-1 Internet Explorer Content tab

4. In the Content tab, click **Certificates**.
5. In the Certificates window shown in Figure 11-2, select the **Personal** tab. Displayed will be the certificates that will appear in the drop-down list on the Host On-Demand session configuration window and the Server Requesting Certificate window. If the certificate is not in this list, it will be obtained from either the WellKnownTrustedCAs.class file or the CustomizedCAs.class file.

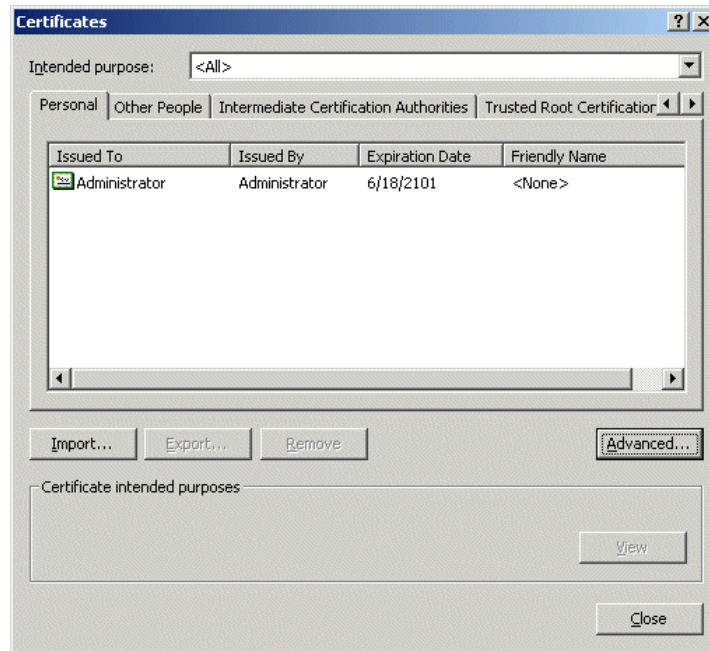


Figure 11-2 Personal certificate window

Adding a personal certificate

There are some security issues you need to be aware of when you add your personal certificate to the cryptographic database. The following instructions will assist you in the process. Please note that this procedure was developed using Microsoft Internet Explorer Version 5.5.

1. Start the Internet Explorer Version 5 browser.
2. Select **Tools -> Internet Options**.
3. Select the **Content** tab in the Internet Options window.
4. In the Content window, click **Certificates**.
5. Make sure you have selected the **Personal** tab, as shown in Figure 11-2, then click **Import** to start the process of adding your personal certificate to the database.
6. Click **Next** on the wizard startup window.
7. Enter the location of your personal certificate. You may click **Browse** to navigate to and select the file, or you may just type the location into the input field (see Figure 11-3). When completed, click **Next**.

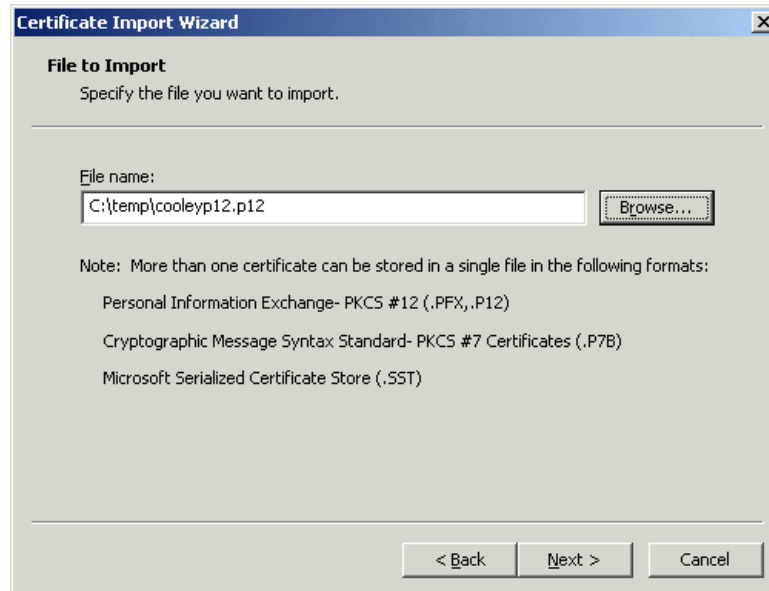


Figure 11-3 Import certificate

8. You must enter your password for your personal certificate in the entry field as shown in Figure 11-4. If you select the first check box, then the browser will take an active role in prompting you for permission to use the certificate.



Figure 11-4 Prompt for certificate password

9. Click **Next** to continue to the next window shown in Figure 11-5. You should specify that the certificate is to be stored in the Personal store by selecting the second radio button as shown in Figure 11-5, then click **Browse**, and select **Personal** from the resulting list as shown in Figure 11-6.

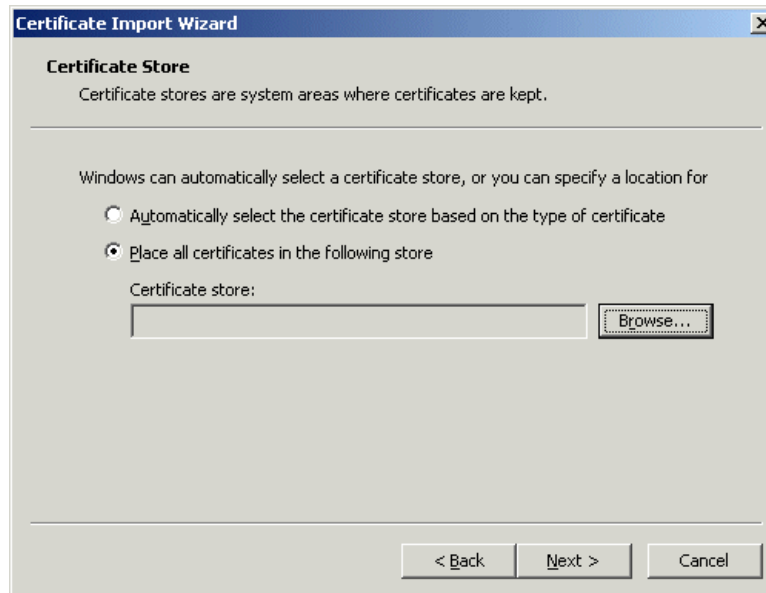


Figure 11-5 Certificate store

10. Click **OK**, then **Next** to proceed to the last window of the wizard where you will click **Finish**.



Figure 11-6 Select Personal certificate store

11. If you selected the second check box in Figure 11-4 on page 436, then you are finished.
12. If you selected the first check box, then you will be presented with the window shown in Figure 11-7.



Figure 11-7 Importing a private exchange key with security wizard

13. Click **Set Security Level** in the window shown in Figure 11-7 to set the desired security level for this certificate. You will be presented with three choices as shown in Figure 11-8.



Figure 11-8 Choose security level

- If you select the default of **Medium**, you will be notified with a pop-up window that an application (Microsoft Internet Explorer) is requesting access to the protected file. When you click **OK**, this will allow Host On-Demand to present the certificate to the Telnet server. Clicking **Cancel**

will not allow Host On-Demand access to the certificate and Host On-Demand will present an error window. When you clear the error window, Host On-Demand will then prompt you for the certificate password.

- If you choose **High** security then you will be prompted, as shown in Figure 11-9, to provide a common name for your certificate and the password to be used when accessing it. Remember, this password is not the same as the certificate password. Click **Finish**, then **OK**.

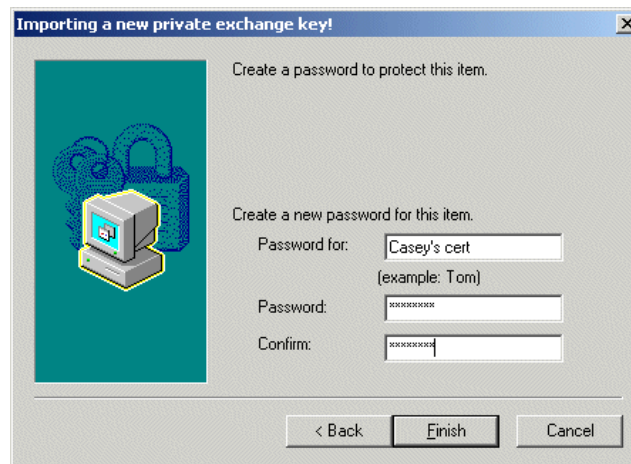


Figure 11-9 Create database password

- If High security is selected, the user will be prompted at runtime to provide a password (see Figure 11-10) in order to release the certificate. Notice that there is a check box to remember the password.

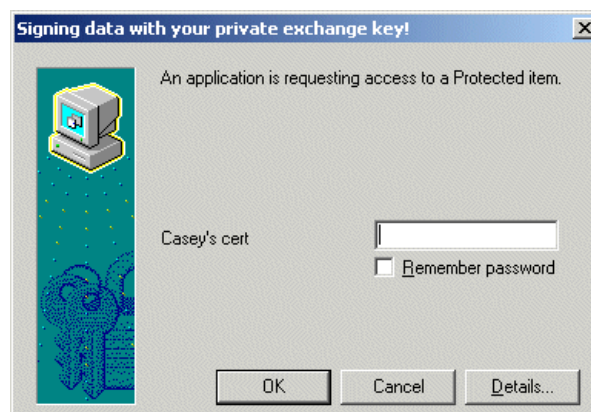


Figure 11-10 Cryptographic password prompt

- If the check box is selected, then the system will remember the password and the next time the certificate is accessed, the prompt window shown in Figure 11-11 will appear, and all the user needs to do is click **OK**.



Figure 11-11 Cryptographic remembered password

Recommendation

Implementation of the cryptographic database depends upon a user model that requires each user of the system to log in to the Windows system with a unique ID. If multiple users use the same Windows user ID, then they will share the same copy of the cryptographic database and thus each user will have access to all client certificates.

11.3 Host On-Demand SSL implementations

SSL is supported by all Host On-Demand emulator clients, 3270, 5250 and VT, whether they are locally installed, cached, or downloaded clients. There are three ways to use SSL with the emulator clients:

1. Basic SSL
2. Server authentication
3. Client authentication

11.3.1 Basic SSL

By default, when SSL is enabled for the Host On-Demand client, a basic SSL session is established. The server will present its certificate to the client during the negotiation process; refer to “Establishing secure communications with SSL” on page 1020. With basic SSL enablement, all that is required is that the client recognize that the certificate is signed by an authority that it trusts.

If the client is running on a Windows platform and the session properties have the MSIE browser keyring file enabled, then the Microsoft cryptographic database is checked first to determine if the signer is trusted. If the signer certificate is not found in the cryptographic database, the cryptographic database is not enabled, or the client is not running on a Windows platform, then the WellKnownTrustedCAs.class file followed by the CustomizedCAs.class files will be checked. If the signer is not found in any of these repositories, the session is rejected. If the signer is found, the session is established.

11.3.2 Server authentication

Encrypting the data exchange between the client and the server does not guarantee the client is communicating with the correct server. To help avoid this danger, you can enable server authentication. Server authentication is a process where the client validates that it is communicating with the correct server before the session may be established. When implementing server authentication, the client must trust the server's certificate before the session will be initiated.

Server authentication is not enabled by default and must be selected on the Security tab in the session properties definition as shown in Figure 11-12.

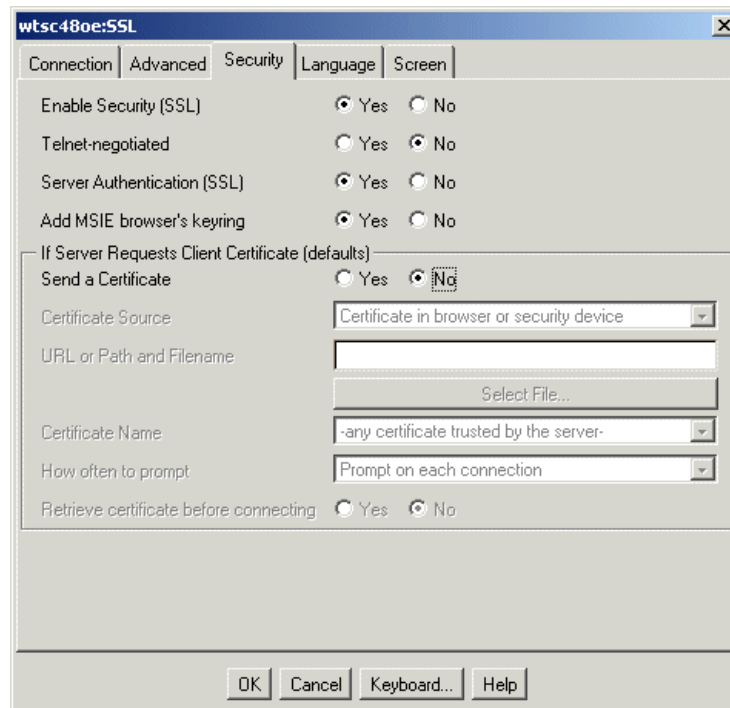


Figure 11-12 Enabling server authentication

When server authentication is selected, a secure session is negotiated as described in “Establishing secure communications with SSL” on page 1020. However, the Host On-Demand client immediately looks at the Common Name field of the server’s certificate to determine if the host name of the server presenting the certificate is stored in the Common Name field of the certificate.

Using one or more Java virtual machine (JVM) calls, the client obtains all IP numeric addresses associated with the Common Name in the server’s certificate. Next, JVM calls are made requesting all IP numeric addresses associated with the server as specified in the destination field of the session properties definition. When the results of both searches are complete, the client compares the two lists of addresses looking for at least one IP address that appears in both lists. If any IP address appears in both lists, the connection continues and data can be sent; however, if no IP address appears in both lists, then the connection is terminated, and an error generated to the session status line. For server authentication to work, a DNS must be available that can resolve these addresses, or the server address must be defined in the TCP/IP hosts file.

For server authentication to be valid and to give a positive result, two conditions must be met if you are not using the cryptographic database:

1. The client must be locally installed.

A client downloaded using HTTP cannot be trusted for server authentication because the WellKnownTrustedCAs.class file and the CustomizedCAs.class file are downloaded from the server.

2. The Common Name in the server's certificate must match its Internet name.

The crucial step in the process is when the client checks its list of trusted CAs and self-signed certificates. For a locally installed client, the list is kept on the local hard disk. This is considered adequately secure. However, for a download client, on which the client is a browser that downloads all its code from the server using HTTP(S), the only place the browser can look for the list of trusted CAs or self-signed certificates is on the server from which it has just downloaded the certificate. If that server is an intruder, or if an intruder can intercept and alter data passed from the server to the client, security is breached.

11.3.3 Client authentication

The server may also want to restrict access only to clients that the server trusts. The process of client authentication has the Telnet server requesting a certificate from the client to verify that the client is who it claims to be, and that it is allowed access to the server. Not all servers support client authentication, including the Host On-Demand Redirector. To configure client authentication, you must obtain certificates for clients, send the certificates to the clients, and configure the clients to use client authentication.

Client authentication is similar to server authentication except that with client authentication the Telnet server requests a certificate from the client to verify that the client is who it claims to be. The certificate must be an X.509 certificate and signed by a Certificate Authority (CA) trusted by the server. You can only use client authentication when a server requests a certificate from a client. Not all servers support client authentication, including the Host On-Demand Redirector.

In order to use client authentication you must:

- ▶ Obtain a client certificate.
- ▶ Transfer the certificate available to the client by either sending it directly, or making it available via a shared LAN drive or a secure HTTPS connection.
- ▶ Always send the password for the certificate via a separate out-of-band secure method so as not to compromise the certificate.

The certificate can be kept in the client's browser, a dedicated security device such as a smart card, or in a local or network-accessed file in PKCS12 or PFX format, which is protected by a password.

When a certificate expires, follow the renewal procedures specified by the CA for that certificate.

11.3.4 FTP client

The FTP client does not support secure FTP file transfer.

11.3.5 TN3270 client

The 3270 display and printer clients support SSL. The Host On-Demand 3270 display session supports all three types of SSL sessions:

- ▶ Basic
- ▶ Server authentication
- ▶ Client authentication

The 3270 printer session supports the following two types of file transfer:

- ▶ Host File Transfer (IND\$FILE)

The IND\$FILE mode uses the 3270 data stream to transfer the data; therefore, if the emulator session is encrypted so is the file transfer data.

- ▶ FTP

The FTP option is the same method as deployed with the FTP client described in 11.3.4, “FTP client” on page 444; therefore, SSL is not supported when using this file transfer method.

11.3.6 TN5250 client

The TN5250 emulator client supports SSL sessions. The Host On-Demand 5250 emulator client supports all three types of SSL sessions: basic, server authentication, and client authentication. The 5250 emulator supports two types of file transfer:

- ▶ host file transfer
- ▶ FTP

Host file transfer with TN5250 does not use the 5250 data stream to do the file transfer as does the TN3270 host file transfer (IND\$FILE). It uses a file transfer method derived from the Host Servers Licensed Product (57xxSS1/Option 12). When configuring host file transfer, the window shown in Figure 11-13 is displayed.

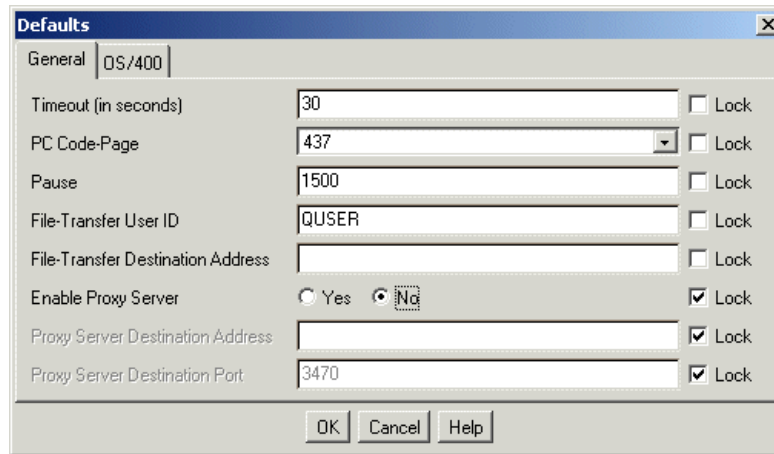


Figure 11-13 Configure 5250 host file transfer

If you select **No** for Enable Proxy Server, then the file transfer operation will occur to the destination file transfer address using the same security as the 5250 client. This means if the 5250 session is not encrypted, then file transfer data will not be encrypted, but if the 5250 session is encrypted, the file transfer data will also be encrypted using server authentication.

If you select **Yes** for Enable Proxy Server, then the file transfer operation from the client to the OS/400 Proxy server will not be encrypted. Refer to Chapter 10, “OS/400 Proxy” on page 419 for details on the operation and configuration of the OS/400 proxy server.

The FTP option is the same method as deployed with the FTP client described in 11.3.4, “FTP client” on page 444; therefore, SSL is not supported when using this file transfer method.

11.3.7 VT client

The VT client supports SSL. Unless the system you will be connecting to supports SSL on the VT session, you must use a redirector that does, such as the Host On-Demand Redirector or the Communications Server for AIX Telnet Redirector.

11.3.8 AS/400 Database On-Demand client

The Database On-Demand client supports the use of SSL. Figure 11-14 illustrates the ;Secure=TRUE parameter that may be added to the Database URL when initiating a database query. If you wish to use the OS/400 proxy server you need to add

;Proxy Server=rallyas4c.itso.ral.ibm.com

to the Database URL, where rallyas4c.itso.ral.ibm.com is the OS/400 Proxy server destination address.

Note: If you are using the OS/400 proxy for port reduction, the session between the client and the proxy will not be encrypted even if the secure parameter is specified.

Figure 11-14 Database On-Demand configuration

Note: If you are using the Netscape browser and you see this message when logging on:

Please disable the JIT compiler and restart the browser.

you must stop your browser, rename the Netscape jit*.dll file and restart your browser. This file is located in the
 \program files\netscape\communicator\program\java\bin\ directory.

11.4 Defining a secure Telnet session

There are many options available when enabling security for a Telnet session. In order to understand the interrelationships and operational implications, each of the settings for each option as shown in the Session Security window (Figure 11-15) is examined.

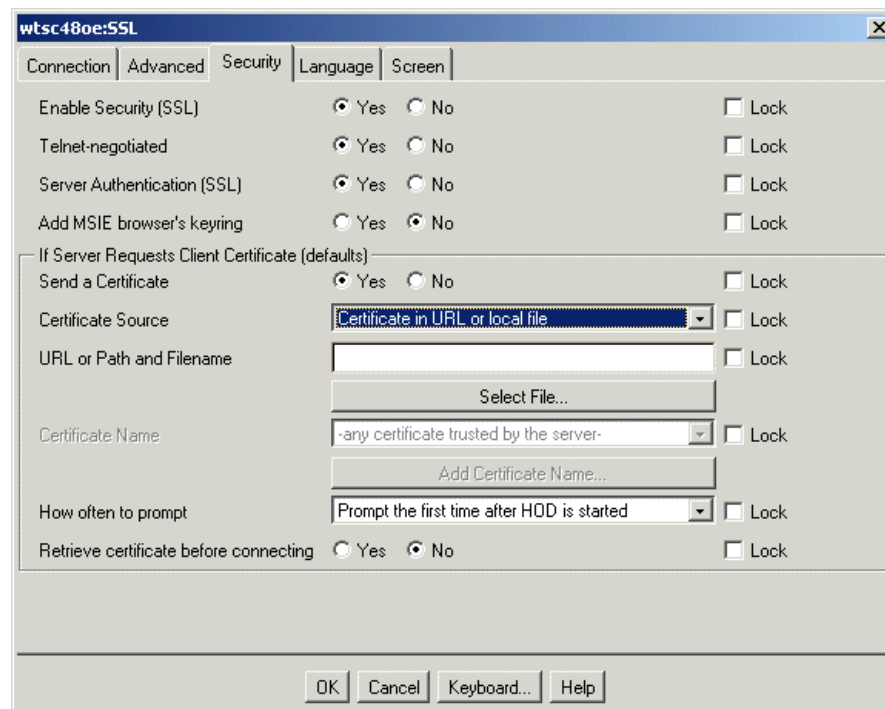


Figure 11-15 Session configuration for security

11.4.1 Enable Security (SSL)

The first and most important option is whether or not to enable security (SSL). If you click **No**, the remaining options in this configuration window will be unavailable. Thus, the client will not attempt to do any SSL and all transmissions will be in the clear. If you clicked **Yes**, then the remainder of the options become available, and at a minimum basic SSL (see 11.3.1, “Basic SSL” on page 441) will be attempted.

11.4.2 Telnet-negotiated session

This option is not available unless you first enable SSL. Selecting Telnet-negotiated determines if the SSL negotiation between the client and the server is done on the Telnet connection or on an SSL connection prior to the Telnet negotiations. The other SSL options are valid regardless of whether the Telnet-negotiated radio button is Yes or No.

If you click **Yes**, then the Telnet protocol defined in IETF Internet-draft TLS-based Telnet Security will be used to negotiate the SSL security after the Telnet connection is established. This support is only applicable with a Telnet server that supports TLS-based Telnet Security. Communications Server for OS/390 V2R10 is the only IBM Telnet server at this time that supports this function.

If you click **No**, the traditional SSL negotiations will be done on an SSL connection with the server, and subsequently the Telnet negotiations with the server will be done. Since this is not yet an RFC, few Telnet servers have this support, so the default is **No**.

11.4.3 Server authentication

The default here is **No**, but should you click **Yes**, then the client will perform the server authentication process as documented in 11.3.2, “Server authentication” on page 441.

11.4.4 Add MSIE browser’s keyring

Clicking **Yes** for this parameter allows the client, when running on a Windows platform, to search the cryptographic database when validating the server’s certificate. If the CA’s public certificate is not found in the cryptographic provider database, then the applet will look in the WellKnownTrustedCAs.class file followed by the CustomizedCAs.class file if necessary to validate the certificate.

This setting is valid only on a Windows platform. The cryptographic database is available to the Host On-Demand client regardless of the browser being used by the client. This setting has no effect on the client authentication process.

11.4.5 Client authentication

There are many options available if client authentication is to be deployed. First and foremost the session must be configured to respond to the Telnet server with a client certificate. This is done by clicking **Yes** to the Send a Certificate option as shown in Figure 11-15 on page 447. Once you click **Yes**, then the remaining options in this section of the window are enabled.

Certificate source

There are two places that the Host On-Demand client will look for the X.509 certificate:

1. The client's local file system, which includes any configured LAN, NFS, AFS, etc. drives, or from a standard URL
2. A security device, such as a smart card, or from the cryptographic database

If you select **Certificate in URL or local file system**, then you may enter the location where the certificate is found into the URL or Path and Filename field. You may use **Select File** to browse your file system to find the file.

Certificate name

This option will become active when you indicate the certificate is in the browser or security device. Host On-Demand will read the cryptographic database from the machine on which this function is being performed. If you select the default entry, **-any certificate trusted by the server-**, Host On-Demand will search the list returned at runtime and select the first certificate recognized by the server. The operator also has the option to view the certificates found in the cryptographic database at configuration time and select one of them as well. This option is fine if the operator is operating on the settings for his own session, but if the operator is an administrator there is no way to know beforehand what certificates will be available on any given client that will use this setting. For administrator operations, refer to "Selecting the client certificate" on page 449.

When a server requests a certificate, the client will check the status of the Send Certificate option set in the session properties file. If it is set to no, a certificate will not be sent and the session may be denied. If the option is set to yes, then the certificate will be located as per the settings in the session configuration file (see Figure 11-15 on page 447) and the certificate sent to the server. The user may be prompted for the password of the certificate before it is sent. Finally, the server makes a connection if the client's certificate can be trusted.

Selecting the client certificate

When defining the sessions and selecting that the certificate shall be found in the browser or security device (see Figure 11-16), the administrator has the ability to

set a mask that will be used to identify the proper certificate from the cryptographic database.

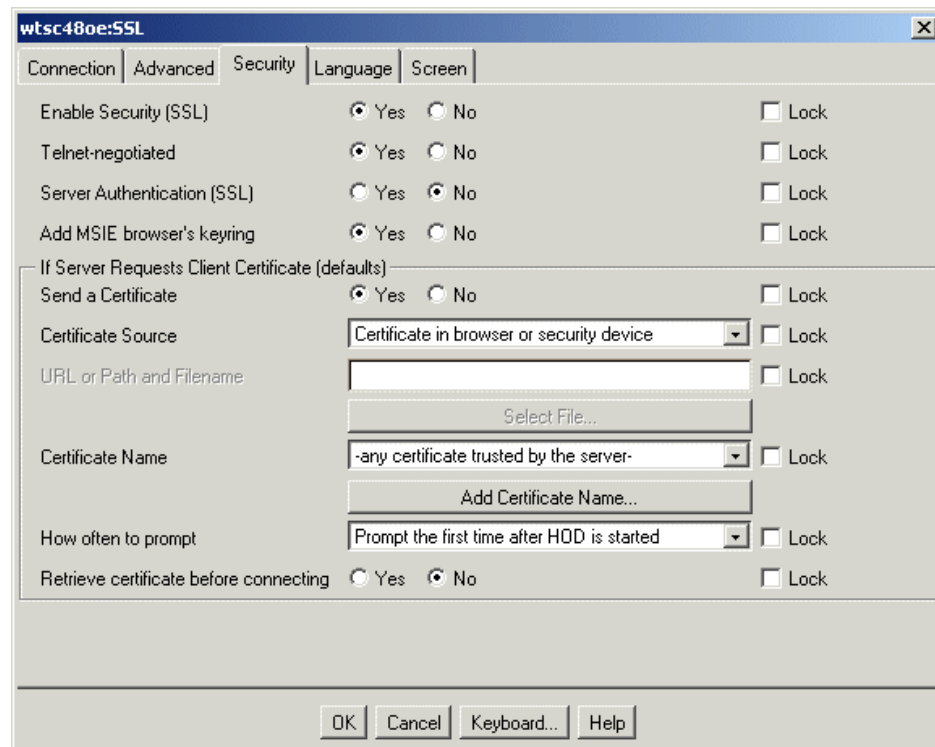


Figure 11-16 Using certificate from browser

The administrator sets the mask by clicking **Add Certificate Name**. This results in a window (Figure 11-17) that allows the administrator to specify a mask that will be used in selecting which certificate from the client's cryptographic database will be selected.

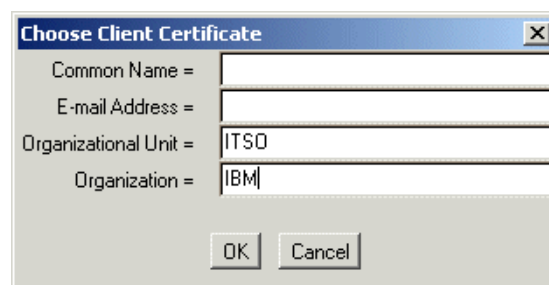


Figure 11-17 Set up client certificate mask

The mask is not case sensitive and wild cards are not allowed. When the certificate is requested, the cryptographic database is searched and the first valid certificate to fit all the components specified is sent. If no certificates are found, the client displays the window shown in Figure 11-18, prompting the end user for further action.

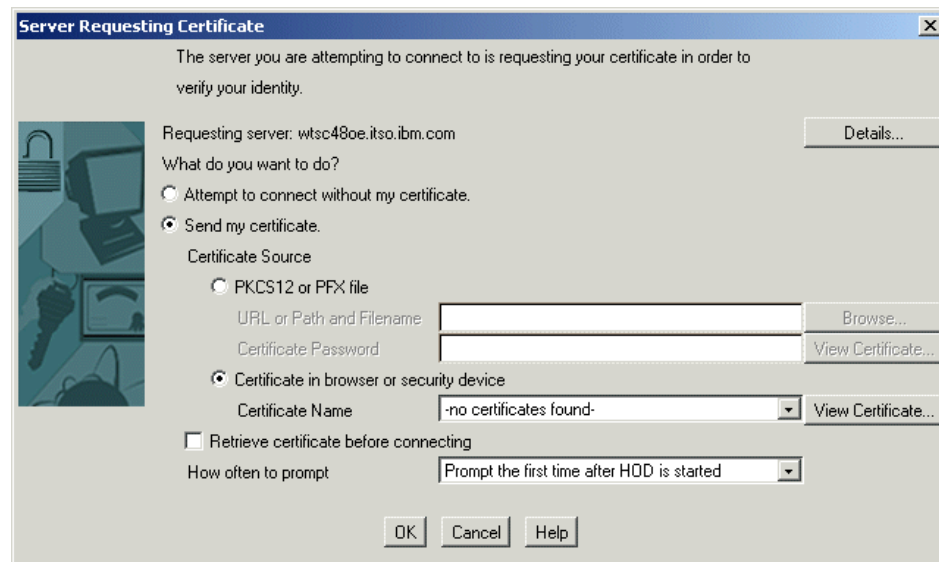


Figure 11-18 No certificates found

The option to specify a mask is available only to the administrator, because if you are the user you should see the list of all of your certificates, but if you are the administrator, there is no way you can see all the certificates on the client's computer.

Note: Using the mask is one technique that may be used to restrict access to valid certificate holders based upon an organizational requirement.

How often to prompt

This drop-down box allows you to control the timing of Host On-Demand prompts for client certificates. The four options are:

- ▶ Prompt on each connection - client is prompted each time a connection is made to the server.
- ▶ Prompt the first time after Host On-Demand is started - client is prompted only once each time the Host On-Demand server is started.
- ▶ Prompt only once, storing preferences on client - if your client stores preferences locally (specified when the client HTML file was created via the

Deployment Wizard), the client is prompted for the password the next time the connection is made, but never after that, unless the connection attempt fails. This option is only available on the client's configuration window.

- Do not prompt - disables the prompt from Host On-Demand, but not from the browser or security device.

If you specified a browser or security device for the certificate source, then the do-not-prompt option will be available to the client. If the certificate is stored in the cryptographic database (browser), no Host On-Demand password prompt is required and you may select the no-prompt option. Host On-Demand will not prompt you for your certificate password; however, depending upon the options you selected when you stored your certificate, the cryptographic database may prompt or notify you. Refer to "Adding a personal certificate" on page 434 for more details.

Retrieve certificate before connecting

If you click **Yes** to retrieve a certificate before connecting (Figure 11-16 on page 450), the client will access its certificate before connecting the server, whether the server requests a certificate or not. If you click **No**, the client will access the certificate only after the server has requested it; depending on other settings, this may force the client to abnormally terminate the connection to the server, prompt the user, and then re-connect. It is recommended that you choose Yes if you will be authenticating with a Communications Server for OS/390 system; otherwise, unnecessary error messages may be generated.

11.5 The Host On-Demand Redirector

The Redirector is a Telnet proxy that is written primarily in Java. The Redirector is able to accept connections from clients and pass them on, through a different port, to the next stage in the link to the host. The Redirector has the following main functions:

- Hide the real host system address and port number from the client, a common requirement when providing Internet-attached clients access to secure host systems.
- Provide SSL support for all emulator clients when the Redirector is running on Windows NT or 2000, or AIX.

The Redirector when running on either a Windows NT or AIX server is capable of supporting SSL sessions in one of the following ways:

- Client-side: SSL is enabled between the Redirector and the client.
- Host-side: SSL is enabled between the Redirector and the host Telnet server.

- Both: the Redirector will support SSL sessions on both the client and the host side simultaneously, managing each SSL session separately.

On all platforms the pass-through mode is supported. The pass-through mode allows the client and the server on the other side of the Redirector to communicate in the clear, or to negotiate a secure session directly, including the use of Telnet-negotiated session, basic SSL, and client authentication.

11.5.1 Redirector certificates

When performing SSL the Host On-Demand Redirector relies on two files for certificate management. These files are found on the Host On-Demand server in the \hostondemand\bin directory, and are:

- HODServerKeyDb.kdb

This is the server's keyring database file, created during SSL configuration (described in 12.3.5, "Create a self-signed certificate" on page 500). It contains:

- Root certificates for well-known CAs (these are inserted when the file is created)
- A self-signed certificate (when one exists)
- Certificates that you have imported from authorities you trust
- Public keys of all the above certificates
- Private keys of the self-signed certificate, and of any of your own certificates that have been validated by a CA

- HODServerKeyDb.sth

This is the password-stash file for the keyring database. It is used to store the password in an encrypted form that can be used by the Redirector to open the keyring database file.

These files are not created at installation. They are created by the key-management utility when you install the server's certificate or any unknown CA certificate you may be using.

When using the Redirector and configuring any connection in anything other than pass-through mode, you must install a public key (site) certificate on your server. There are three choices:

1. Use a certificate from one of the well-known CAs whose root certificate is already in the WellKnownTrustedCAs.class.
2. Use a certificate from a CA whose root certificate is not in the file (a unknown CA).
3. Use a self-signed certificate.

Obtain a certificate from a CA

To use a certificate from one of the well-known CAs or some other CA, you must request the certificate from the CA, receive the certificate, then store it into the keyring database, HODServerKeyDb.kdb file. Nothing else needs to be done to allow the Host On-Demand client to recognize the signer of certificates from the following CAs:

- ▶ RSA Data Security, Inc.
- ▶ VeriSign, Inc.
- ▶ Thawte Consulting

However, if the certificate is from any CA other than one of these well-known CAs that Host On-Demand recognizes, then the public certificate of that signer must be made available to the client in the CustomizedCAs.class file. Refer to 12.3.6, “Make a certificate available for the clients” on page 503 and “Using Microsoft cryptographic database” on page 491 or if you are using the Microsoft Internet Explorer 5.0 or above browser, you may authenticate the server’s certificate from the list of CAs that Microsoft recognizes. For OS/390 server, refer to “Make certificates available to clients” on page 119.

Self-signed certificate

If your security requirements do not warrant the purchase of a commercial certificate, if you need a temporary certificate while you are waiting for your permanent certificate, or if you need one just for testing, you can create your own (self-signed) certificate by using the key management utility to create a self-signed certificate. Refer to 12.3.5, “Create a self-signed certificate” on page 500, or for OS/390 “Create a self-signed certificate” on page 117.

11.5.2 Configuring the Host On-Demand Redirector

Configuring the Host On-Demand Redirectory is covered in 7.3.1, “Configuring the Redirector” on page 351.

11.5.3 Certificate management

For a complete discussion on certificate management, refer to Chapter 12, “Certificate management” on page 487.

11.6 The OS/400 Proxy server

The OS/400 toolbox delivered the OS/400 proxy server to Host On-Demand. The OS/400 Proxy server is a service that runs on the Host On-Demand server and provides the ability of a Database On-Demand client and OS/400 file transfer to operate through a single port rather than the standard multiple-port implementation. Refer to Chapter 10, “OS/400 Proxy” on page 419 for details.

11.7 Configuration Servlet

Using a Configuration Servlet allows clients to exchange user authentication and session configuration data over an HTTP(S) connection instead of using the Configuration Server directly. This eliminates the need to open the Configuration Server port on the firewall and provides the potential to encrypt all configuration information as it moves from the Web server to the client. For more information regarding configuring the Configuration Servlet, refer to Chapter 9, “Configuration Servlet” on page 397.

11.8 Express Logon Feature

The Express Logon Feature (ELF) allows a user running a 3270 client session to log on to a host system without entering a user ID and password. ELF is invoked via a macro that uses digital certificates in place of user IDs and passwords to log the user on to RACF-enabled applications. Using ELF reduces the number of user IDs and passwords that users need to remember.

To use Express Logon Feature, the following are required:

1. The host session must be configured for SSL with client authentication.
2. The connection must be to one of the supported Telnet servers.
3. Each user must have his own unique digital certificate because ELF and RACF will associate each digital certificate to the user's RACF user ID and password.
4. You must record a macro that the user will use to log on to the host application. The macro record function steps you through the process for creating an ELF macro.
5. Distribute that macro to the clients.

Some configuration needs to be done on the Telnet servers and on the iSeries system that you are accessing. The information in this book will assist you in configuring the Host On-Demand component. For a complete tutorial and examples of all supported platforms, refer to:

<ftp://ftp.software.ibm.com/software/network/library/whitepapers/elf.pdf>

For additional configuration information, you may also refer to the documentation for the server platform you have implemented:

- ▶ Communications Server for OS/2 Warp - *What's New*
- ▶ Communications Server for Windows NT - Readme file
- ▶ Communications Server for AIX - Readme file
- ▶ Communications Server for OS/390:
 - *z/OS V1R1.0 CS: IP Migration*, SC31-8773

Refer to 3.7, “Express Logon Feature (ELF)” on page 130 for details on how to set up the zSeries for ELF.

11.8.1 Host On-Demand session setup

Before you can start recording a macro using the Express Logon Feature, you have to define a session that is able to provide ELF support. When recording the macro, the session definitions are not checked, that is, you can record a macro for ELF support that might not work correctly when played.

The session must be configured for SSL and client authentication. A client certificate must have been installed on the client or must be accessible from a server. The destination IP address must specify a server that has been set up to support the Express Logon Feature.

Note: If the connection to the TN3270 server is through the Host On-Demand Redirector or the Communications Server for AIX Telnet Redirector, the security option for the Redirector must be set to pass-through.

Refer to 7.1.7, “Configuring sessions” on page 285 for details on how to configure the Host On-Demand client sessions.

11.8.2 Record the macro

Recording the macro is started the normal way by clicking **Record** on the session window's toolbar or by selecting **Actions -> Record Macro**. The session itself may have been started from a client by logging in as a user and then opening the intended session window. You may also record an ELF macro as an administrator customizing an HTML page that, when referenced, automatically opens the session window, starts the macro, logs the user on to his host application, and navigates the user to the application's start window.

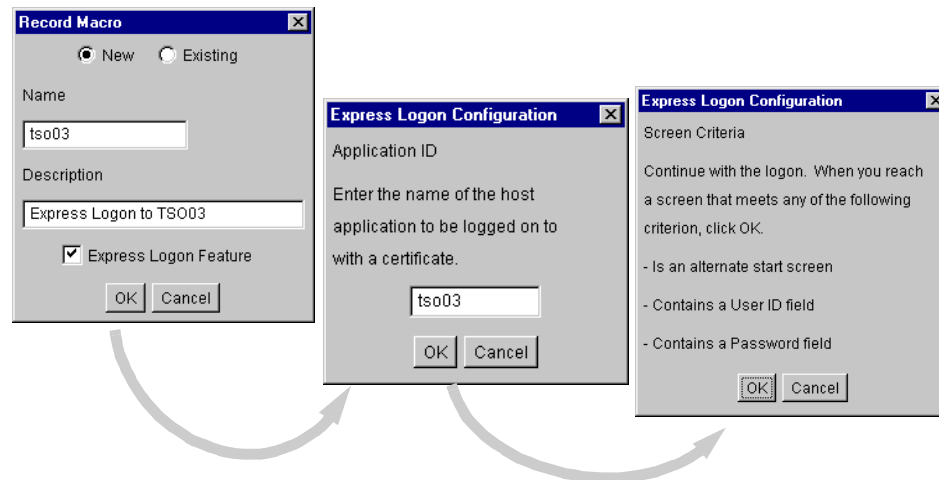


Figure 11-19 Recording the ELF macro - getting started

Figure 11-19 shows the sequence of the first three windows that appear when you start recording an ELF macro. On the first window you have to specify the name of the new macro (of course, you may also append to or overwrite an existing macro). Select the **Express Logon Feature** check box to indicate that you want to use ELF. Clicking **OK** causes the second window in Figure 11-19 to appear, prompting you to enter the application ID of the application you are logging on to with this macro. This is the name of the application that was used when it was defined to RACF on the OS/390 host.

After having entered the application ID and clicking **OK**, the third window in Figure 11-19 appears, prompting you to actually start recording your actions on the session window.

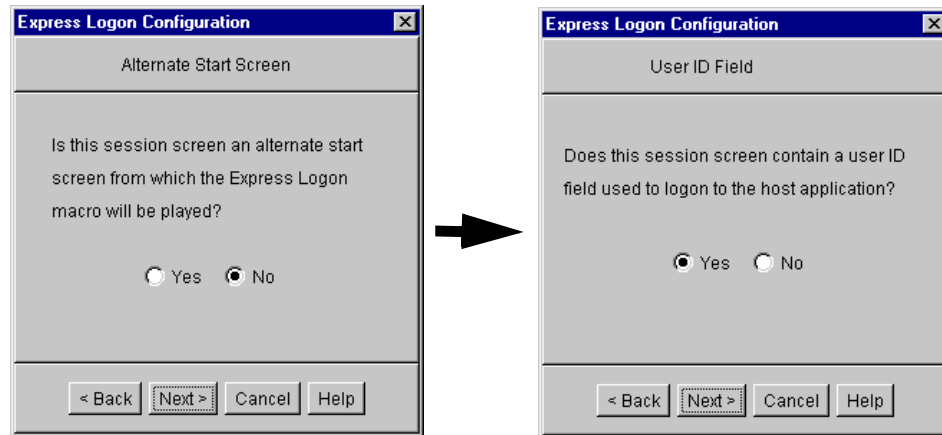


Figure 11-20 Recording the ELF macro - getting to the user ID field

Once you have reached the window prompting you for the user ID, click **OK** on the third window in Figure 11-19. The next window (Figure 11-20) then will ask if this is an alternate start window.

You can define alternate start windows, which can be more than one, in the first or a follow-on editing pass through the macro. This will allow the user to start the macro (or have it started automatically) when the host session is initialized. After having logged off from the application, a different logon window might be presented to the user (for example, the application's logon window and not VTAM's USSMSG10). This then will allow the user to use the same macro for one application, independent of where he starts.

The next window, when not defining an alternate start window, asks if there is a user ID field on the current host window. Clicking **Yes**, then **Next** from the following window leads you to a window that lets you define the position of the user ID field on the host window. See Figure 11-21.

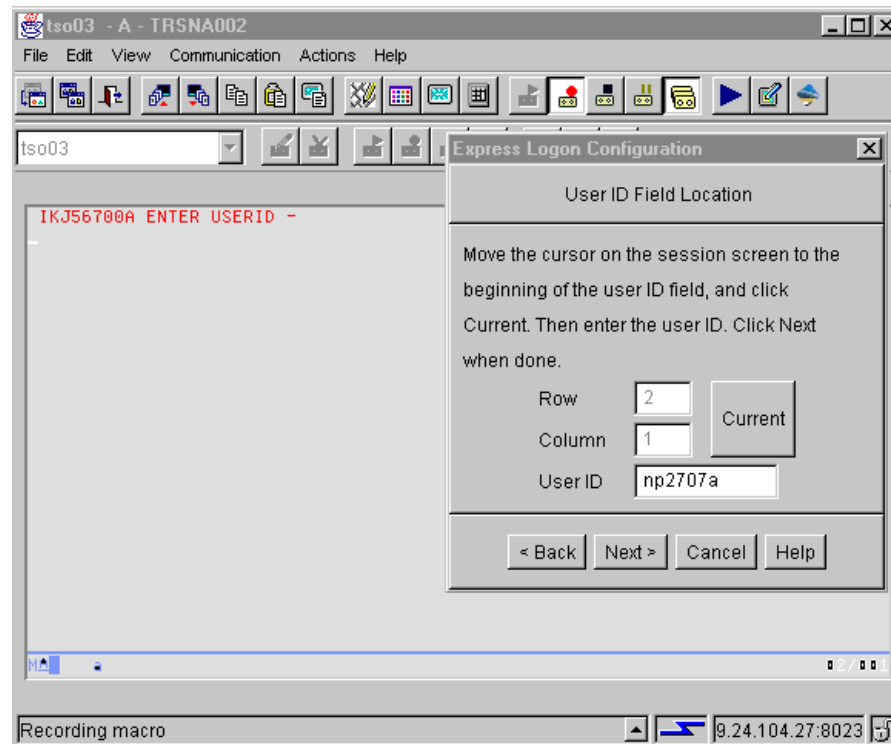


Figure 11-21 Recording the ELF macro - user ID field

The simplest way of getting the correct row and column is by positioning the cursor on the user ID input field (normally it will already be correctly positioned) and clicking **Current**. This will update the input fields in the window with the current cursor position. In the user ID field, fill in a valid user ID. This user ID then will only be used to log on to the host application while recording the macro; it will not be recorded in the macro. Instead, a placeholder variable, `)USR.ID(`, will be placed in the macro and actually filled into the host window's user ID field when the macro is played. The TN3270 server then will replace this variable with the user's correct user ID.

The next window presented will ask if there is also a password field on the host window that prompts for the user ID. If you answer Yes, the password field is on the same window as the user ID field, or after having navigated to the window prompting for the password, you have to define the position of the password field on a window similar to the one used for the password field as shown in

Figure 11-21. Also, the password you are entering here is not recorded in the macro. It is only used to actually log on when recording the macro. The macro will again contain the placeholder variable,)PSS.WD(, that will be replaced with the PassTicket by the TN3270 server when playing the macro.

When you click **Finish**, the left window shown in Figure 11-22 is displayed giving instructions on how to continue. Only when you really want the user to press the Enter key, or whichever PF key is used for the logon, do you follow the instructions to stop the macro immediately. Otherwise, click **OK** to remove the window and continue recording your macro until you have reached the application's start window where you want to leave the user.

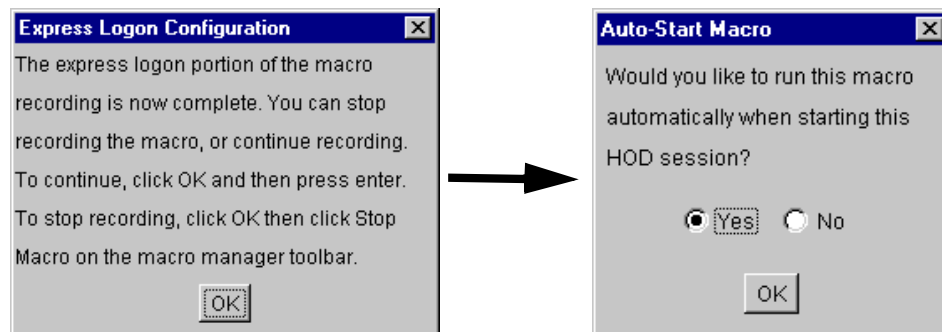


Figure 11-22 Recording the ELF macro - finishing steps

When you stop recording the macro, a final window (shown on the right in Figure 11-22) will appear asking you whether you want this macro to be automatically started when the session window is initialized. If you click **Yes**, the corresponding session definitions will be updated.

Important: The initial release of ELF used the variables \$USR.ID\$ and \$PSS.WD\$, but because of national language translation issues these variables were changed to)USR.ID(and)PSS.WD(. This change for Host On-Demand is introduced in Version 5.0.4. The following releases of the mid-tier communications servers support the new variables:

- ▶ Communications Server for OS/2 V6.1
- ▶ Communications Server for Windows NT and Windows 2000 V 6.1.1
- ▶ Communications Server for AIX V 6.0.0.1

11.8.3 ELF Design

The Express Logon Feature is supported on two-tier and three-tier network designs. The two-tier design utilizes the z/OS TN3270 Telnet server. The three-tier design utilizes a middle-tier TN3270 server and a Digital Certificate Access Server (DCAS).

In order for an application to be accessed using the Express Logon Feature, a PassTicket data class profile (PTKTDATA) must be defined on each target RACF-enabled system (that is, the host where DCAS is running, and any host where RACF and a target application is located).

Both network designs require a TN3270 client workstation that supports Secure Sockets Layer (SSL) connections with client authentication and an X.509 certificate. Using RACF services in z/OS, the client certificate must be associated with a valid user ID. The only client-side product that supports the Express Logon Feature is IBM WebSphere Host On-Demand V5 and V6.

Two-tier network design

In the two-tier design, the user starts an SSL connection with level 2 client authentication, which passes the client certificate to the MVS host TN3270 server. The MVS host TN3270 server uses RACF Secured Signon services to obtain a user ID and PassTicket.

The two-tier design is supported in z/OS V1R2 and OS/390 V2R10 + PQ47742 (UQ55691).

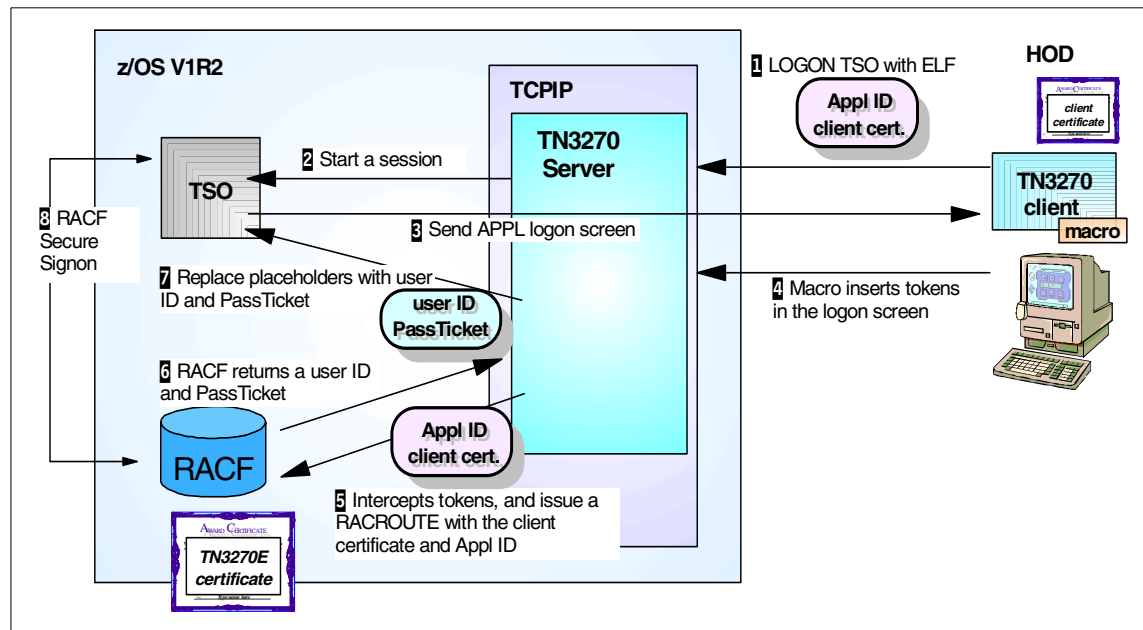


Figure 11-23 ELF two-tier network design

Here are the steps when a client wants to access a TSO session on z/OS:

1. The user has a Host On-Demand icon that starts an emulator session configured to use SSL client authentication. The session has a macro associated with it. A client certificate has to be available from the terminal and presented to the TN3270 server for the SSL handshake. During the SSL handshake, the client certificate is passed to the TN3270 server and validated. During Telnet function negotiation, the ELF capability is negotiated using RFC 1572.
2. The application ID is sent from the client to the TN3270 server and the server starts the session with TSO.
3. The logon screens come to the emulator.
4. The macro plays and inserts placeholder strings in the user ID and password fields.
5. The TN3270 Server intercepts the placeholder strings and sends the certificate and the target application ID to RACF.
6. RACF converts the Host On-Demand client's certificate to a TSO user ID and PassTicket and sends them back to the TN3270 server.

7. The TN3270 server inserts the user ID and PassTicket into the 3270 data stream at the macro-inserted placeholder locations and sends it to the application.
8. The application presents the user ID and PassTicket to RACF or other compatible host access control facility, which approves them and the logon completes as usual.

Three-tier network design

In the three-tier design, the user starts the TN3270 connection to the middle-tier server.

There must be a Digital Certificate Access Requester (DCAR) and a Digital Certificate Access Server (DCAS).

The DCAS's client is the middle-tier TN3270 server or DCAR, which attempts to log on to an SNA application for the workstation client. The DCAS receives a digital certificate from the DCAR and returns a user ID and PassTicket. SSL communication is used between the DCAS and the DCAR. The server recognizes that the client wants the Express Logon Feature and invokes the DCAR, which opens an SSL connection with client authentication and passes the workstation's certificate and application name to the DCAS on the host. The DCAS uses RACF Secured Signon services to obtain a user ID and PassTicket, which the DCAS returns to the DCAR. The DCAR passes this information back to the TN3270 server.

The middle-tier IBM TN3270 servers supporting ELF are:

- ▶ Communications Server for OS/2 Warp V6.1
- ▶ Communications Server for Windows NT and Windows 2000 V6.1.1 PTF
- ▶ Communications Server for AIX 6.0.0.1 PTF

The term DCAR is used to describe the part of the TN3270 middle-tier server that supports the Express Logon Feature and communicates as a client with the DCAS. It is not separate from the TN3270 middle-tier server.

A Digital Certificate Access Server (DCAS) resides on the host. DCAS uses RACF services to obtain a user ID as shown in Figure 11-24.

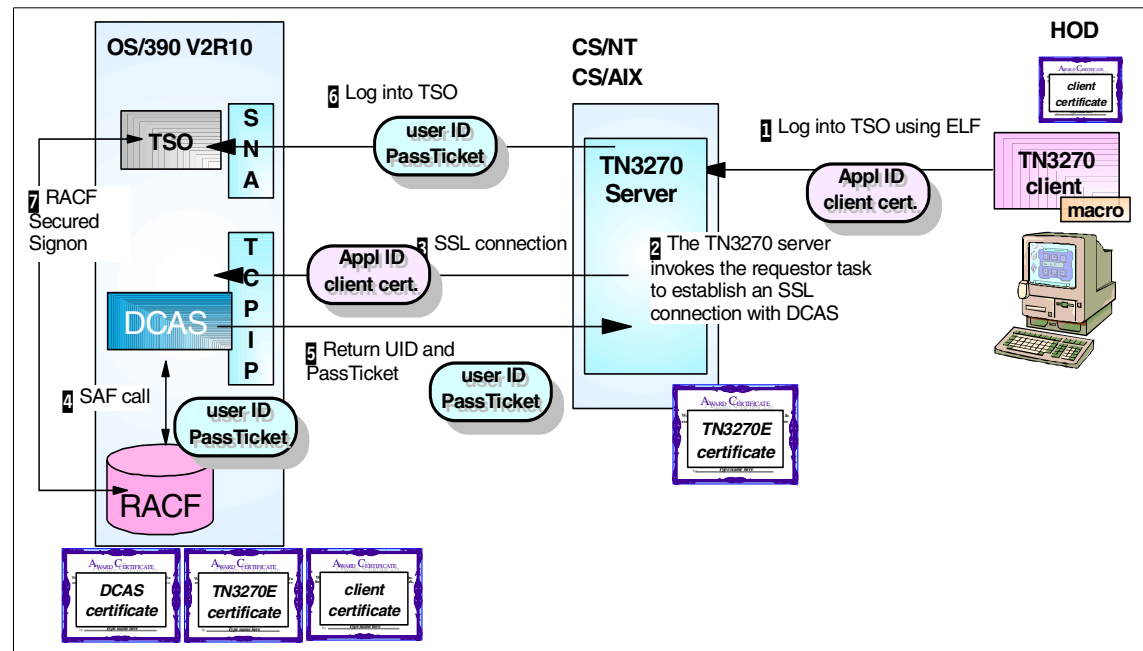


Figure 11-24 ELF three-tier network design

The host also provides RACF Secured Signon services, which the DCAS or the MVS host Telnet server uses to generate a PassTicket. A PassTicket is a RACF token similar to a password except that it is valid only for ten minutes.

The three-tier components are:

- ▶ A client workstation that supports Secure Sockets Layer (SSL) connections with client authentication and an X.509 certificate.
- ▶ A middle-tier TN3270 server, so called because it does not reside on the host, but rather between the client and the host. This server communicates with a DCAS using an SSL connection with client authentication. It sends the user's certificate from the workstation and an application ID to the DCAS and expects to receive a user ID and PassTicket (a one-time password) in response. This is the user ID and password that will be used to log on to the SNA application.
- ▶ The Digital Certificate Access Server (DCAS) resides on the host. The DCAS uses RACF services to obtain a user ID that is associated with the certificate sent by the client. RACF also provides secured sign-on services, which the DCAS uses to generate a PassTicket. A PassTicket is a RACF token similar to a password except that it is valid only for 10 minutes.

The SNA connection between the TN3270 server and SNA application can be SNA LU2, DLUR, HPR/IP (EE), or AnyNet connection.

The Secure Sockets Layer (SSL) communication with client authentication is required in the configuration of the Express Logon Feature on the Host On-Demand client, TN3270 server, and OS/390 DCAS server.

The following describes the example shown in Figure 11-24 where the client wants to access a TSO session on z/OS:

1. The user has a Host On-Demand icon that starts an emulator session configured to use SSL client authentication. The session has a macro associated with it. A client certificate has to be available from the terminal and presented to the TN3270 server for the SSL handshake.

During the SSL handshake, the client certificate is passed to the TN3270 server and validated. During Telnet function negotiation, the ELF capability is negotiated using RFC 1572.

The application ID is sent from the client to the TN3270 server. The logon screens come to the emulator as usual, but the macro plays and inserts placeholder strings in the user ID and password fields. The TN3270 Server intercepts the placeholder strings.

2. Once the application ID and client certificate are received by the TN3270 server, it invokes the DCAR function to establish the SSL communication to the DCAS server. The TN3270 server's certificate has to be sent to DCAS and authenticated.
3. The TN3270 server sends the certificate and the target application ID to DCAS on the z/OS host over a secure, trusted connection.
4. The DCAS server makes SAF calls to convert the Host On-Demand client's certificate to a TSO user ID and PassTicket.
5. The DCAS server sends the user ID and PassTicket back to the TN3270 server.
6. The TN3270 server inserts the user ID and PassTicket into the 3270 data stream at the macro-inserted placeholder locations and sends it to the application.
7. The application presents the user ID and PassTicket to RACF or other compatible host access control facility, which approves them and the logon completes as usual.

11.9 LDAP directory considerations

There are two basic security concerns when using an LDAP security server:

- ▶ Securing the communications between Host On-Demand and the LDAP directory server
- ▶ Encryption of user passwords within the LDAP directory server

Host On-Demand does not support SSL encrypted communications with the LDAP directory server. All communications will be in the clear; therefore, if you use the LDAP directory server, you may wish to configure the communications link between your Host On-Demand server and the LDAP directory server to occur over a link that resides in the secure network.

Most LDAP directory servers support the storage of user passwords in either clear text, or encrypted. The two most common encryption algorithms implemented are Secure Hash Algorithm (SHA) and UNIX crypt. Some LDAP directory servers now also support MD5. Host On-Demand currently supports passwords stored in clear text, SHA, and UNIX crypt.

11.10 Using Host On-Demand with a firewall

If you are configuring Host On-Demand to go through a firewall, it is recommended that the firewall administrator open only those ports required for the clients to function. At a minimum, you will need to open the following ports:

1. HTTPS port

This port will be used for downloading the applet and for obtaining configuration information via the Configuration Server.

2. Telnet ports

There may be one or more ports open, depending upon the Telnet server requirements. These ports should allow SSL-encrypted session traffic.

3. OS/400 proxy server port

If you are using Database On-Demand or TN5250 file transfer, you should utilize one or more OS/400 Proxy server ports to pass all traffic over a single port per proxy server.

11.10.1 TCP/IP ports used by Host On-Demand

Table 11-1 identifies all the ports that Host On-Demand will use in its default configuration. Remember, that many of the ports are configurable, such as the Configuration Server port and the OS/400 Proxy server port. Use Table 11-1 to determine how to configure your firewall.

Table 11-1 Ports used by Host On-Demand

Host On-Demand Function	Unsecure Port(s) used	Secure Port(s) used
3270 and 5250 Display Emulation	23 (Telnet) 80 (HTTP) 8999 (config server) ³	992 (Telnet) 443(HTTPS)
3270 and 5250 Printer Emulation	23 (Telnet) 80 (HTTP) 8999 (config server) ³	992 (Telnet) 443(HTTPS)
3270 File Transfer (IND\$FILE)	23 (Telnet) 80 (HTTP)	992 (Telnet) 443(HTTPS)
5250 File Transfer - SAVF file	80 (HTTP) 8999 (config server) ³ 21 (FTP) ⁴ >1024 (FTP) ⁴ 446 (drda) ⁴ 449 (as-svrmap) ⁴ 8470 (as-central) ^{1 2 4} 8473 (as-file) ^{1 4} 8475 (as-rmtcmd) ^{1 4} 8476 (as-signon) ^{1 4}	
5250 File Transfer - database	80 (HTTP) 8999 (config server) ³ 446 (drda) ⁴ 449 (as-svrmap) ⁴ 8470 (as-central) ^{1 2 4} 8473 (as-file) ^{1 4} 8475 (as-rmtcmd) ^{1 4} 8476 (as-signon) ^{1 4}	
5250 File Transfer - stream file	80 (HTTP) 8999 (config server) ^{1 2 4} 449 (as-svrmap) ⁴ 8470 (as-central) ^{1 2 4} 8473 (as-file) ^{1 4} 8476 (as-signon) ^{1 4}	443 (HTTPS) 9470 (as-central) ^{1 2 4} 9473 (as-file) ^{1 4} 9476 (as-signon) ^{1 4}

Host On-Demand Function	Unsecure Port(s) used	Secure Port(s) used
Host On-Demand Administration	80 (HTTP) 8999 (config server) ³	443 (HTTPS)
Database On-Demand	80 (HTTP) 8999 (config server) ³ 449 (as-svrmap) ⁴ 8470 (as-central) ^{1 2 4} 8471 (as-database) ^{1 4} 8476 (as-signon) ^{1 4}	
License Use Count License Use Management (LUM)	8999 (config server) ³ 80 (HTTP)	
<p>Notes: Port numbers listed are the default values.</p> <p>¹ You can change the port numbers with the OS/400 WRKSRVTBLE command. The port numbers listed are the default values.</p> <p>² The port for as-central is used only if a code-page conversion table needs to be created dynamically (EBCDIC to/from unicode). This is dependent on the JVM and the locale of the client.</p> <p>³ You can change the config server port.</p> <p>⁴ These ports do not need to be opened on the firewall if you are using OS/400 proxy server support. You will need to open the default proxy server port 3470. You can change this port.</p>		

11.11 Native Authentication

User ID and password management has become an ever-increasing issue for users as the number of systems and applications that require authentication continues to grow. In order to save a user's preferences at the Host On-Demand server, a user ID is required to uniquely identify the user. This ID is used as the index under which the user's preferences are stored in the repository. Host On-Demand does not require passwords to be implemented with the user ID; however, most customers implement a password for an additional level of identification. Because of the platform-independent nature of Host On-Demand, this user ID and password management as implemented by Host On-Demand is independent of any other user ID and password management system.

Host On-Demand Version V6 still requires the administrator to define and manage user IDs when a Configuration Server model is implemented, but with the introduction of the Native Authentication component we allow the administrator to associate the Host On-Demand user ID with a user ID and password known to the native operating system. The native platform

authentication service allows users to log on to Host On-Demand using the same password as they would to log on to the operating system where Host On-Demand is active (Windows NT Server, AIX or z/OS). When a user logs on to Host On-Demand, their password is validated against the system password, rather than a separate Host On-Demand password, thus providing the customer with the following benefits:

- ▶ Reducing the number of passwords that the end user must remember. In many cases this means that the user will have only one password to remember.
- ▶ Better security, and a reduction in the administrative workload for the Host On-Demand administrator by delegating password management to an administrative system that can implement a password management policy that typically includes:
 - Enforcement of password rules
 - Enforcement of password expiration times
 - Ability to revoke access by invalidating a password

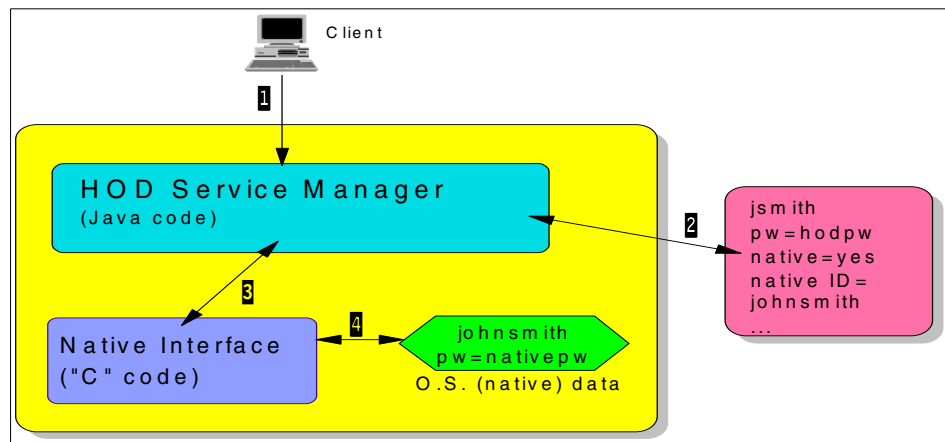


Figure 11-25 Native Authentication login flow

When a user logs on (1), the user ID and password are sent to the Host On-Demand Configuration Server. The Configuration Server sends a request for logon information about the user to the LDAP server (2). The LDAP server returns a message indicating if the user is configured for Native Authentication. If the user is not configured for Native Authentication the password stored in the LDAP directory server is returned to the Configuration Server. If the user is configured for Native Authentication the native ID stored in the LDAP directory is returned along with an indicator that Native Authentication should be invoked. The Configuration Server checks the returning information from the LDAP

directory server. If the user is configured to use Native Authentication, the Configuration Server sends the user ID and the password to be authenticated to a Host On-Demand module written in C and compiled for the specific operating system (3). That module will invoke the appropriate native operating system security call to validate the user ID and password combination (4). If the user is not configured for Native Authentication, the Configuration Server compares the password that was entered by the user with the password returned by the LDAP server.

If the user ID and password are successfully validated by the operating system, processing continues. All other returns will result in an invalid password error message as shown in Figure 11-26. Other than a legitimately invalid password, one of the most common reasons for this return message will be an expired password. There is no mechanism within Host On-Demand to intercept an expired password and prompt for a new one. The user will be required to correct this condition via some other interface and then log on again to Host On-Demand.



Figure 11-26 Native Authentication failure

11.11.1 Native platform authentication requirements

Native platform authentication service must be installed on a Windows NT, Windows 2000, AIX, or zSeries Host On-Demand server. On Windows NT or 2000, native platform authentication requires Windows NT Server or Windows Advanced NT Server (LANMAN) with a non-null domain.

On the Host On-Demand server, LDAP directory services must be enabled and configured for Native Authentication individually for each user that is to use Native Authentication; refer to 7.1.4, “Using Native Authentication” on page 281. The LDAP directory server may reside anywhere in the network and may run on any platform.

Follow the steps below to use native platform authentication with Host On-Demand:

1. Enable users for Native Authentication.
2. Start the native platform authentication service.
3. Configure current users for Native Authentication.

11.11.2 Installation and activation

The files to support native platform authentication are installed with the Host On-Demand server during the installation process. With Windows NT and Windows 2000, some additional installation steps are required as defined below.

Windows NT and Windows 2000

The operating system must be Windows NT Server, Windows 2000 Server or LANMAN.

On Windows NT and Windows 2000, Native Authentication runs as a service: IBM ODS Platform Authentication Service.

Update the registry

On Windows NT and Windows 2000, the following additional steps are required to update the registry:

1. Define a new environment variable, `hod_dir`, and set its value to the drive letter where Host On-Demand is installed. The `hod_dir` environment variable is used by the registry settings to locate Host On-Demand. To update the variable, click **Start -> Settings -> System**, select the Environment tab, and add a system variable `hod_dir=x:`, where `x` is the drive where Host On-Demand is installed. It must be a system variable, not a user variable, so that the services can use it.
2. Using Windows NT Explorer, locate the `odsrpd.reg` file in the Host On-Demand bin directory, and double-click the file to add the registry settings defined in the file.

Important: There will be two ODSRAPD files, ODSRAPD.EXE and ODSRAPD.REG. Make sure you are selecting the `odsrpd.reg` file verifying that the type attribute is Registration Entries not Application.

3. This step is only required for Windows NT; it is not required for Windows 2000. Using `regedit`, find the registry value for:

```
HKEY_LOCAL_MACHINE\SOFTWARE\IBM\On-Demand Server for Windows  
NT\2.0\installpath
```

Edit the `installpath` value so that `%hod_dir%` is replaced with the drive letter where Host On-Demand is installed. For example, if Host On-Demand is installed on the D drive, change:

```
%hod_dir%\hostondemand\private
```

to

```
d:\hostondemand\private
```

4. Reboot the server.

Once you reboot, you can go to the Services window by clicking **Start -> Settings -> Services**, and you should see IBM ODS Platform Authentication Service showing as started.

Set user rights policies

The final step is to set the proper user rights in the Policies section of the User Manager. To set the correct user rights in the Windows NT system, follow these instructions:

1. Open the User Manager on Windows NT. This is normally found by clicking **Start -> Programs -> Administrative Tools (Common) -> User Manager**.

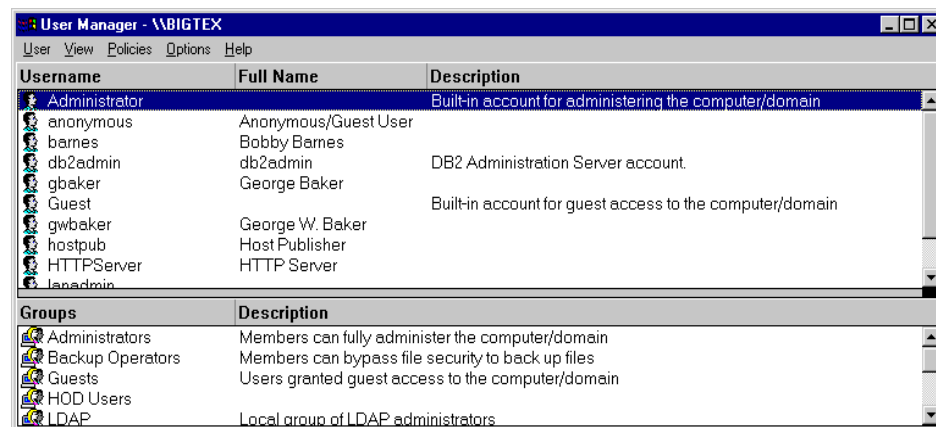


Figure 11-27 Windows NT User Manager

2. Click **Policies > User Rights** from the menu bar of the User Manager.
3. Check **Show Advanced User Rights**.
4. In the Right field, select **Log on as a batch job**. See Figure 11-28.

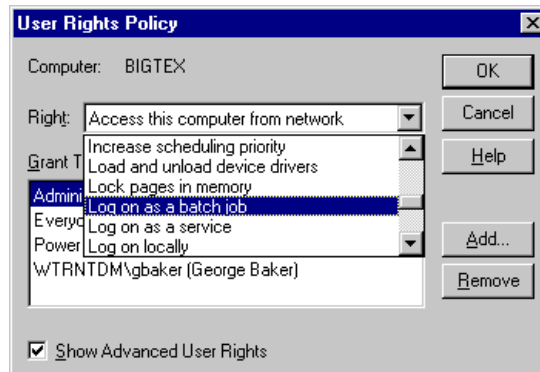


Figure 11-28 Advanced user rights

5. Click **Add**.
6. From the Names field, select users who will be using native platform authentication and click **Add**. To add members of a group, select the group and click **Members**. As you add users, the users' names are displayed in the Add Names field. We recommend that you either allow the group of all users, Everyone, or create a group, such as Host On-Demand, and include all Host On-Demand users in this group.

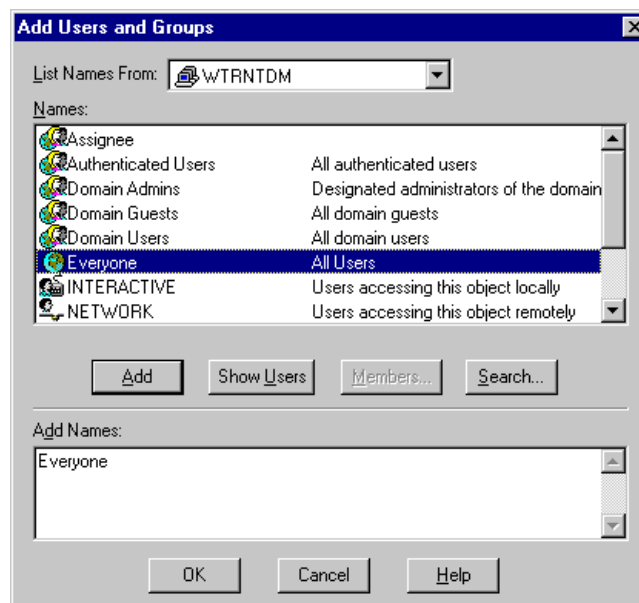


Figure 11-29 Adding authorized users

7. When you are finished adding users, click **OK** to close the Add Users and Groups window and save your changes.
8. Click **OK** to close the User Rights Policy window.

You can now exit the User Manager. All users that were granted the right to log on as a batch job can be authenticated using the native platform authentication service.

The native platform authentication service is started from the Windows NT Services menu. By default, this service is set to start automatically.

zSeries

For information on zSeries support for Native Authentication, refer to 3.9, "Native Authentication" on page 140.

AIX

To start the program, a user with root authority must execute the shell script `odsrapd.sh`, which is located in the `/usr/opt/hostondemand/bin` directory.

The syntax for the start is as follows:

```
odsrapd.sh [parameters]
```

Where the parameters are as follows:

- l** Enables logging. You can also specify `-L`, `/l`, or `/L`. Note: Native Authentication code logs its messages to the syslog, which may need to be configured to log the desired level of messages.
- txx** Sets the socket timeout to some other value instead of the default 20 seconds. You can also specify `-T`, `/t`, or `/T`. `xx` as the new timeout value.
- cxx** Specifies the number of requests the server will allow. You can also specify `-C`, `/c`, or `/C`. `xx` as the new number of requests the server will handle.

The Native Authentication code uses the syslog to log its messages. If you do not already have one defined, you may need to configure one in order to log the desired level of messages. One of the key prerequisites is that the log must exist prior to `odsrapd` trying to access it.

The syslog can be configured to report any level of message desired, but initially it is best to get all levels (debug). Once everything is working well, you can restrict the message reporting to just errors (crit). Modifying syslog information will require root authority.

Adding the following line to the end of the `/etc/syslog.conf` file will log all messages to `/etc/hod/rapd.out`:

```
user.debug/etc/hod/rapd.out
```

The syslog daemon will not log the messages if the file does not exist; therefore, you must create the `rapd.out`, if it does not already exist. Finally, stop and restart `syslogd` so that it reads its new configuration file.

Before you run the shell script for the first time, you will need to change the permissions on both the `odsrpd.sh` shell script and the `odsrpd` executable. You should verify that the Native Authentication code started correctly by checking the syslog for a starting message from `odsrpd`. This message will be in the file `rapd.out` if you used the sample provided here.

11.11.3 Debug information

If you have problems getting Native Authentication working, you will need the following information to assist in debugging the problem:

- ▶ System type
- ▶ User's Host On-Demand and native user IDs
- ▶ User's error message
- ▶ Host On-Demand Service Manager trace with level 3 debug messages
- ▶ Native Authentication logged messages (syslog messages on AIX and zSeries), or the event viewer application log for Windows NT and Windows 2000

11.12 Integrated Windows domain logon

Access to information is becoming increasingly complicated in terms of security. Every user is expected to pass through several layers of security measures before they actually access the information. This could start with a power-on password and proceed through several layers of network and application security. This leads to complex security layers and administration policies. IBM Host On-Demand Integrated Windows domain logon feature is here to reduce the complexity by at least one layer without any compromise to the security.

The Integrated Windows domain logon feature is a new feature available to Host On-Demand V6.0 Windows users only, and is only valid for Configuration Server-based model users, where users are required to log on to the Host On-Demand server to obtain their session preferences. Enabling this function causes the Host On-Demand client to query the Windows operating system and

retrieve the domain and user ID used during Windows logon. The returned user ID then becomes the Host On-Demand user name. The domain returned will be compared to the list of allowed domains for the users and if valid will invoke the login process with the Configuration Server automatically without prompting the user.

The benefits of using this feature are:

- ▶ A reduction in the number of user IDs and passwords the user has to remember.
- ▶ Security and password management is done by the Windows workstation and the domain controller.
- ▶ Reduces the administrative workload on the Host On-Demand administrator, since the administrator need not create and maintain individual user IDs.
- ▶ Users will bypass the Host On-Demand logon window.

The Host On-Demand server may be on any platform, since all Integrated Windows domain logon service functions are performed on the Windows client. For Integrated Windows domain logon to work, the user must invoke Host On-Demand using an HTML file that has been specifically created with the Deployment Wizard by the administrator. See 11.12.1, “Activating Integrated Windows domain logon” on page 476.

11.12.1 Activating Integrated Windows domain logon

The following steps must be performed by the administrator in order to enable Integrated Windows domain logon:

1. Start the Host On-Demand Deployment Wizard (for details, see 14.2, “Starting the Deployment Wizard” on page 530).
2. In the first window, select **Create an HTML file** and click **Next** to display the window shown in Figure 11-30.

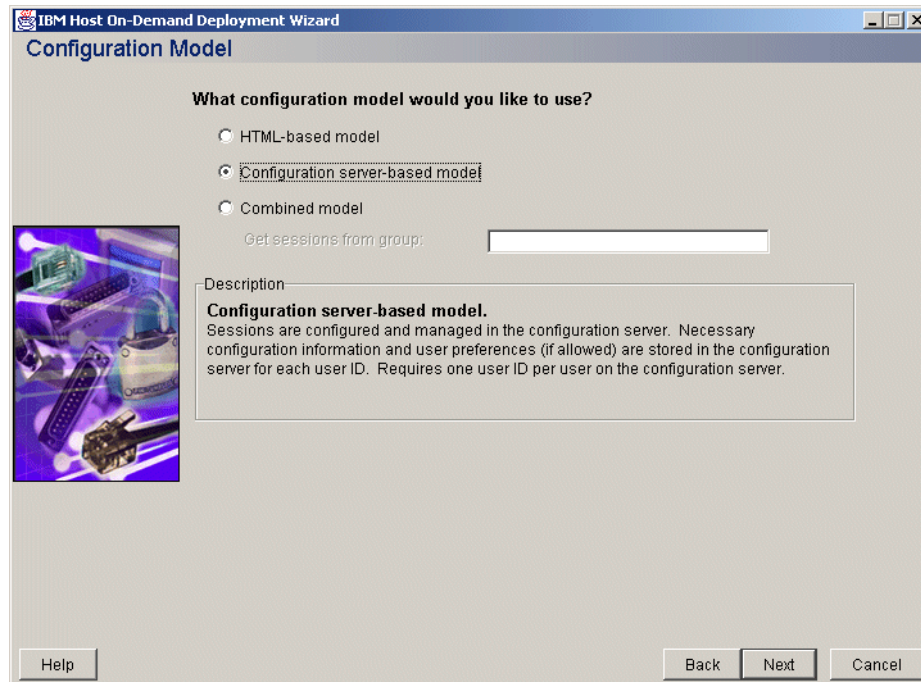


Figure 11-30 Configuration server based model

3. Select **Configuration Server-based model**, then click **Next** to display the panel shown in Figure 11-31.

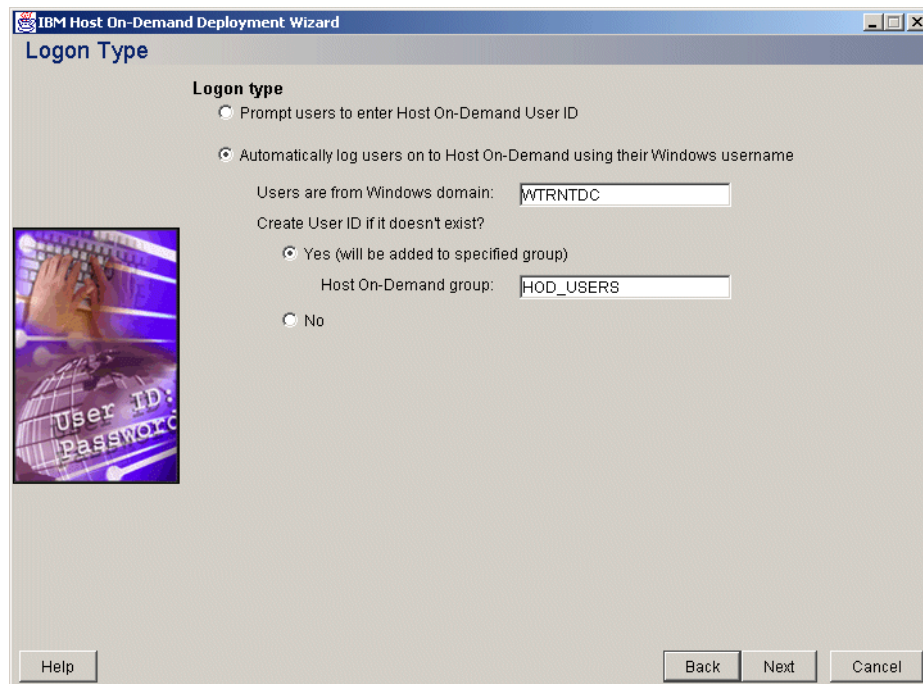


Figure 11-31 Single sign-on option selected

4. Select **Automatically Log users on to the Host On-Demand using Windows username.**
5. This will enable the options previously greyed out. In the first field enter the domain or domains, separated by a comma, into which the user must log on. If the user is not logged into one of these domains, the user will be presented with a message that they are not authorized as shown in Figure 11-32.

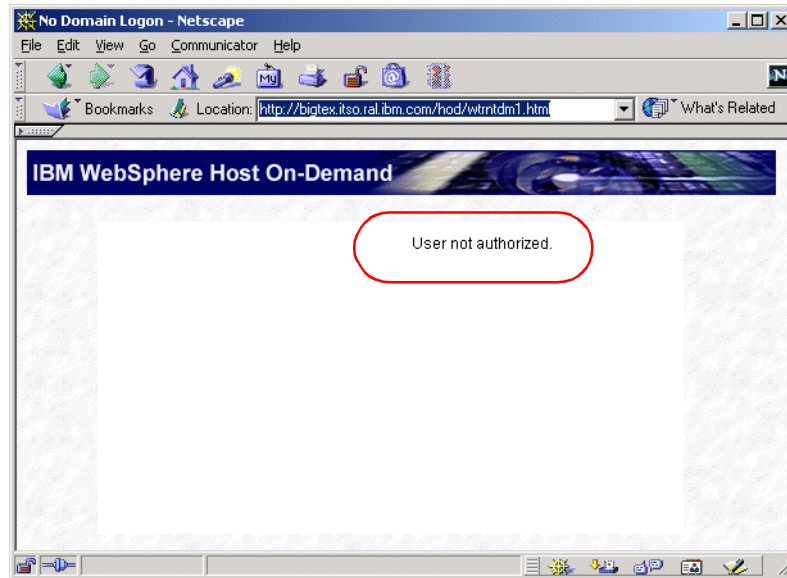


Figure 11-32 Integrated sign-on - not authorized

6. Next, you must indicate if the Configuration Server is to create the user ID for the incoming user ID. Does it or does it not pre-exist?
 - a. Selecting **Yes** results in the specified user ID being created in the group specified in the following field.
 - b. Selecting **No** results in a message that the user does not exist and the logon is denied since the user was not predefined (see Figure 11-33).

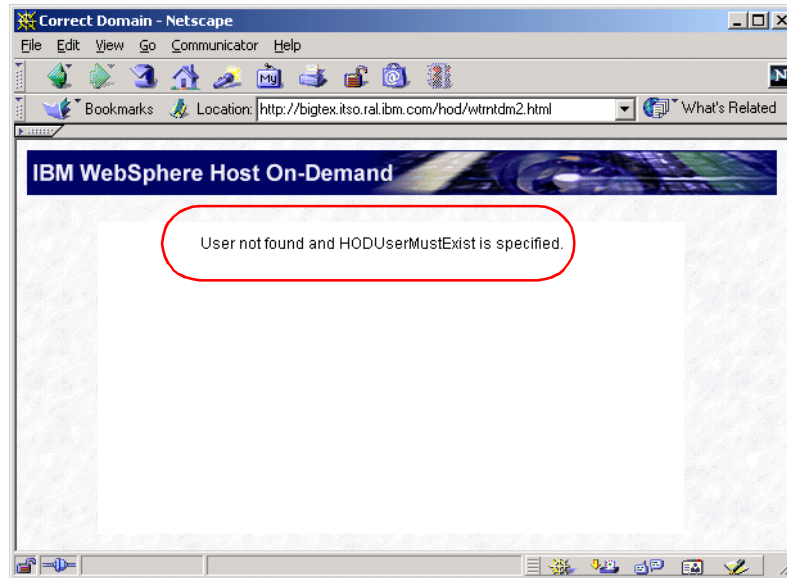


Figure 11-33 User not found

7. Specify the Host On-Demand group in which new users will be created. If this field is left blank, the user will be added to the HOD default group. If specified, the group must already exist.

Note: Host On-Demand will insert Created by System into the description field of the user record.

8. Click **Next** to proceed through the remainder of the Deployment Wizard as documented in Chapter 14, "Deployment Wizard" on page 529 to complete the configuration, store and activate the HTML file.
9. Inform the users of the existence of this HTML file.

11.12.2 Process flow

Figure 11-34 illustrates how this process works.

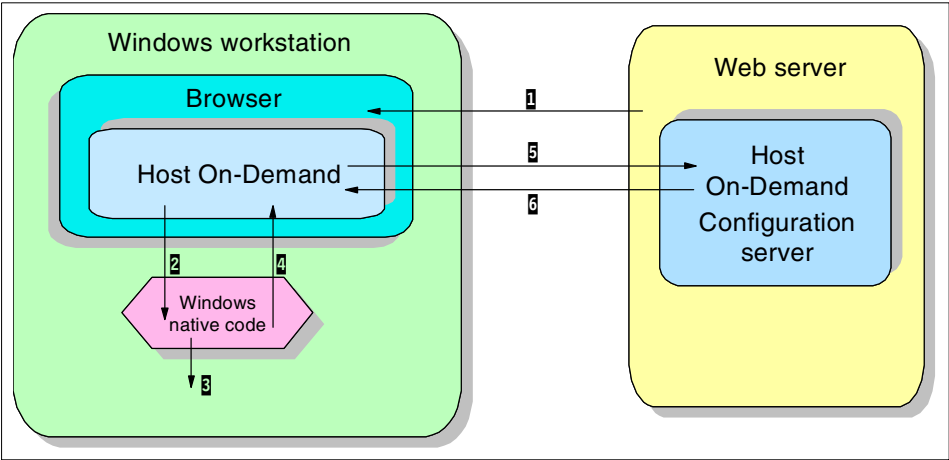


Figure 11-34 Integrated Windows domain logon flow

1. The user downloads Host On-Demand client into the browser using an HTML file created by the administrator with the Deployment Wizard.
2. When the Host On-Demand client detects that Integrated Windows domain logon has been enabled, a call is made to the native Windows code.
3. The native Windows code queries the operating system to obtain the user ID and domain specified during user logon. If the user performed a local logon, the domain to which the system is defined will be returned.
4. The native code returns the user ID and domain name to the Host On-Demand client.
5. The domain name is compared to the list of domain names specified by the administrator. If the domain name does not match the request, it is rejected and the user notified as shown in Figure 11-32 on page 479.
6. If there is a domain name match, then the Configuration Server is contacted to obtain the user preferences. The process will create the user not already present.

11.12.3 Configuration parameters

The following are the parameters generated and stored in the params.txt file by the Deployment Wizard when Integrated Windows domain Logon is specified.

Table 11-2 Integrated Windows domain logon parameters

Parameter	Value
UseWindowsDomain	True if this feature is enabled, else false

Parameter	Value
WindowsDomain	A comma separated list of authorized domains
HODUserMustExist	True, or false if the user may be automatically added if it does not exist
WDHODGroup	Group name to be used if HODUserMustExist=false

11.12.4 Trouble shooting

Be aware of the following issues:

- ▶ Invalid information specified such as the Host On-Demand group in which the user IDs are to be automatically created. This results in the error shown in Figure 11-35.

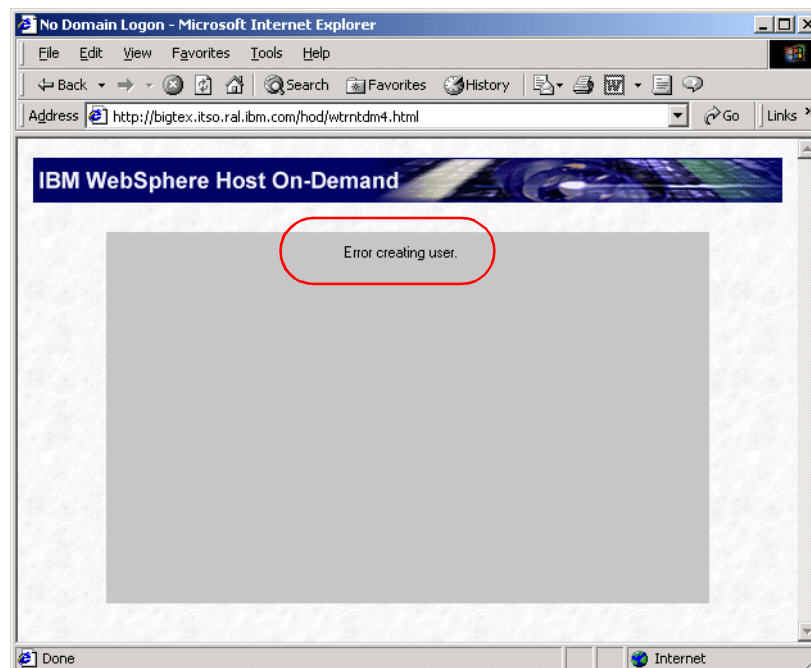


Figure 11-35 Invalid group

- ▶ If the user loads any Host On-Demand HTML file that is configured for the Configuration Server model that was not enabled for the Integrated Windows domain logon feature, he will be prompted for a user ID and password.

Important: The user will be prompted in this environment because the system assigned a random password when his user ID was automatically created. To log on conventionally, the user must have this password reset by the administrator.

11.13 Telnet-negotiated security

The purpose of Telnet-negotiated security is to allow a Telnet session to begin as a non-secure session, but then negotiate a secure session over a Telnet connection using Transport Layer Security (TLS) handshake protocol.

The TLS protocol is defined in IETF Standards Track RFC 2246 “The TLS Protocol 1.0” and is found at:

<http://www.ietf.org/rfc/rfc2246.txt>

It allows for negotiation down to SSL. Host On-Demand does not yet support TLS, so it will always negotiate down to SSL V3.

11.13.1 Session configuration

In order to implement Telnet-negotiated security, you must first select **Yes** to the Enable SSL radio button in order to activate the Telnet-negotiated radio button, then select **Yes** to the Telnet-negotiated radio button (see Figure 11-36).

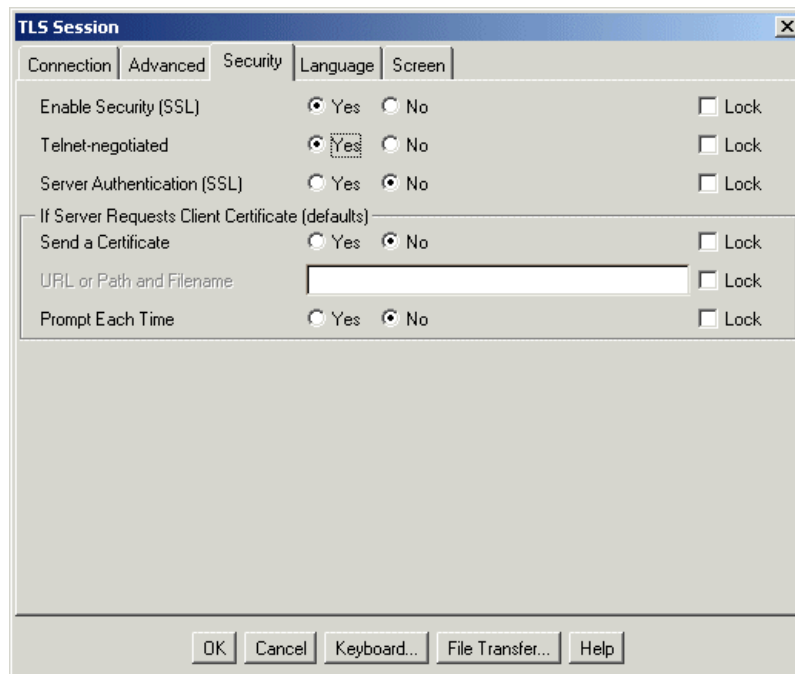


Figure 11-36 Enable TLS-negotiated security

The other SSL options are valid regardless of whether the Telnet-negotiated radio button is Yes or No. Selecting Telnet-negotiated determines if the SSL negotiation between the client and the server is done on the Telnet connection or on an SSL connection prior to the Telnet negotiations. Selecting **No** forces a secure session before initiating the Telnet session.

If **Yes** is selected for both Enable Security (SSL) and Telnet-negotiated, then the Telnet protocol will be used to negotiate the SSL security after the Telnet connection is established. This support is applicable only with a Telnet server that supports Telnet-negotiated security. CS for OS/390 V2R10 is the only IBM Telnet Server at this time that supports this function.

If **Yes** is selected for Enable Security (SSL) and **No** is selected for Telnet-negotiated, the traditional SSL negotiations will be done on an SSL connection with the server, and subsequently the Telnet negotiations with the server will be done. The default is No because few Telnet servers have this support.

If Enable security (SSL) is set to Yes and Telnet-negotiated is set to Yes, then Enable security (SSL) is set back to No, and Telnet-negotiated is no longer selectable. The session has no security, even though Telnet-negotiated is still set to Yes because Telnet-negotiated requires Enable security (SSL) to be first set to Yes. To set Telnet-negotiated back to No, you must first set Enable security (SSL) to Yes so that Telnet-negotiated is again selectable.

If Enable Security (SSL) is set to No and the server requests a Telnet-negotiated secure session from the client, the Host On-Demand client will not start a session and an error message will be issued.

The CS for OS/390 documentation refers to this feature as “negotiable SSL”.

11.13.2 Session negotiation

A typical Telnet-negotiated Telnet SSL flow is shown in Figure 11-37.

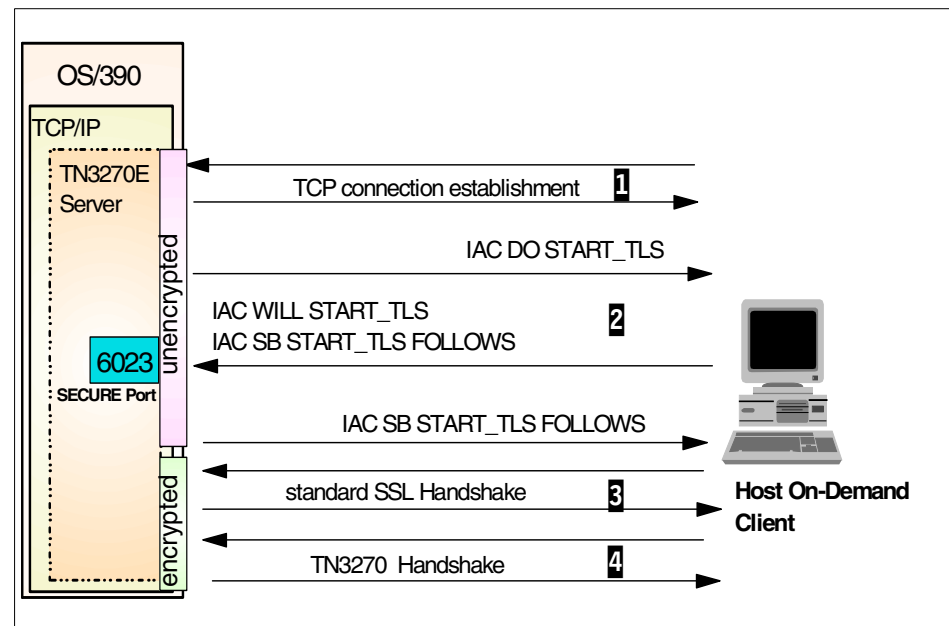


Figure 11-37 TLS-based Telnet SSL flow

1. IP connection establishment.
2. The Telnet server sends the IAC DO START_TLS command to the client to verify if it wants to perform the SSL negotiation.
3. If a positive response is received, then Telnet begins a normal SSL handshake.

4. If no positive response is received, the connection will be dropped.

The IAC D0 START_TLS Telnet command, sent from the server, activates TLS at the beginning of a Telnet connection. The client can respond to this command by sending the IAC WILL START_TLS command, if the negotiation of a TLS connection is required. With the IAC DONT START_TLS command, the client can refuse the TLS connection negotiation. Sending the IAC SB START_TLS FOLLOWS IAC SE command initiates a TLS negotiation. When this subcommand has been sent and received, the TLS negotiation will begin.

If Enable Security (SSL) is Yes and Telnet-negotiated is Yes, then the Telnet connection will be started normally without SSL. However, the 3270 session will not start until the SSL negotiation completes successfully. If the server responds with the message WONT STARTTLS, then the session will not start, and an error message will be issued stating Security was requested, but the server does not support security.

If Enable Security (SSL) is No and the server requests a TLS session, Host On-Demand will not start a TLS session and an error message will be displayed on the status bar stating The server requested security, but Security is not enabled.



Certificate management

Certificate management is central to the operation of SSL. This chapter identifies the critical Host On-Demand files that you must be aware of when managing certificates. We then discuss the impacts of using certificates from different sources, such as well-known CAs, unknown CAs, and self-signed certificates, and how to deploy server certificates to the clients.

Finally we discuss the utilities available for creating and managing certificates:

- ▶ The certificate management utility
- ▶ The certificate wizard
- ▶ The keyring utility

12.1 Files managed by Certificate Management

The Host On-Demand Redirector uses the files for key and certificate management on the Host On-Demand server. These files are kept in `\hostondemand\bin`, and are:

- ▶ `HODServerKeyDb.kdb`, the server's keyring database file. It contains:
 - Root certificates for well-known CAs (these are inserted when the file is created).
 - Self-signed certificates (if they exist).
 - Certificates that you have imported from authorities you trust.
 - Public keys of all the above certificates.
 - Private keys of the self-signed certificate, and of any of your own certificates that have been validated by a CA.
- ▶ `HODServerKeyDb.sth`, the password stash file for the keyring database, is used to store the password in an encrypted form that can be used by the Redirector and/or the Express Server to open the keyring database file.

These files are not created at installation. You must create them, by means of the certificate management utility.

Download and locally installed clients use the Java classes `WellKnownTrustedCAs.class` and `CustomizedCAs.class` to perform authentication. With download clients (including cached clients), these are downloaded from the server as required. Locally installed clients use their locally held copies. The `CustomizedCAs.class` is created or updated by the Certificate Management Utility during SSL configuration; it contains the server and CA-root certificates. It must be updated whenever a self-signed certificate is introduced or a certificate from an authority other than one of those well-known CAs is to be used. To be accessible to download clients, both classes are stored in the publish directory.

12.2 Using certificates

To enable SSL for your Redirector, you must install a public key (site) certificate on your server. There are three choices:

1. Use a certificate from one of the well-known CAs whose root certificate is already in the `WellKnownTrustedCAs.class`:
 - RSA Data Security, Inc.
 - VeriSign, Inc.

- Thawte Consulting
- 2. Use a certificate from a CA whose root certificate is not in the file (an unknown CA).

There are many well-known Certificate Authorities that are not in the WellKnownTrustedCAs.class file that you use as your CA, because they are not listed in the WellKnownTrustedCAs.class file, where they will be referred to as unknown CAs.
- 3. Use a self-signed certificate.

The tool that allows you to manage all these certificates is explained in 12.3, “Certificate management utility” on page 492.

You can send certificate requests to the following companies, which already have their public keys recognized by Host On-Demand:

<http://www.verisign.com>

<http://www.thawte.com>

12.2.1 Using a site certificate from a well-known CA

To use a certificate from one of the well-known CAs, you must obtain it from the CA, and add it to HODServerKeyDb.kdb on the server. All clients will recognize it and authenticate with it. The steps are:

1. Create the certificate request.
2. Submit the certificate request to the CA.
3. Obtain and store the certificate in the server’s keyring database.

Public key certificates of well-known CAs

The public key certificates (or root certificates) of a number of well-known CAs are automatically inserted into the keyring databases when they are created. Their names are displayed by the certificate management utility, as shown in Figure 12-9 on page 501.

12.2.2 Using a certificate from an unknown CA

If you wish to use a certificate from an unknown CA, the procedure is more complicated. The steps are:

1. Create the public-key/private-key pair and the certificate request.
2. Submit the certificate request to the CA.
3. Obtain the certificate from the CA.
4. Obtain the CA’s root certificate and store it in the keyring database.

5. Store the site certificate in the keyring database.
6. For downloaded clients, use the key management utility to add the formatted CA root certificate to the CustomizedCAs.class on the server. Refer to 12.3.6, “Make a certificate available for the clients” on page 503.
7. For locally installed clients, extract the certificate from the server’s keyring database file, take or send the certificate to the client via a secure mechanism, and add it to the client’s keyring database file.

12.2.3 Using a self-signed certificate

SSL can be set up quickly and easily to use a self-signed certificate. This is useful for testing purposes or while waiting for a certificate from an unknown CA. It should not be used in a production environment.

The steps are:

1. Using the certificate management utility, create a self-signed certificate. A public-key and private-key pair and a certificate are automatically created. Then store these into the keyring database.
2. For other Host On-Demand clients, extract the self-signed certificate from the database to the CustomizedCAs.class.
3. For locally installed clients, extract the certificate from the server’s keyring database file, take it to the client and add it to the client’s keyring database file.

12.2.4 Making server certificates available to clients

All clients must be able to authenticate the signer of the server certificate. If the signer exists in the WellKnownTrustedCAs.class file, nothing more needs to be done, since all Host On-Demand clients on all platforms have access to this file either locally installed or downloaded from the server.

If you use a certificate from a CA that is not contained in the Host On-Demand WellKnownTrustedCAs.class file, or you use a self-signed certificate, you must provide the client with the signer’s public certificate.

The traditional method is to add the signer certificate to the CustomizedCAs.class file. Download clients receive this file from the server when they load the applet, while locally installed clients must have this file either created or installed on each client separately.

Beginning with Host On-Demand Version 5.03, there is another method for Windows platform users: the Microsoft cryptographic database, which contains many more signer certificates than does the WellKnownTrustedCAs.class file.

Downloaded and cached clients

Downloaded and cached clients must be able to access the certificate from the Host On-Demand server. If the server is using a certificate from a well-known CA, nothing more needs to be done because the certificate is already in the WellKnownTrustedCAs.class file in the “publish” directory, and is therefore accessible to clients. However, if the certificate is self-signed or from an unknown CA, it must be put into the CustomizedCAs.class file, which must be present in the “publish” directory.

The CustomizedCAs.class file is always downloaded to all download clients and available for use even if they are configured to use the cryptographic database. This insures that all clients will have the certificate when required regardless of platform.

Locally installed clients

There are two ways to enable a locally installed client to recognize certificates signed by unknown CAs:

1. Build the CustomizedCAs.class file at the central site and transfer it to the client via a secure method that meets your security policy. The user need only store it in the \hostondemand\lib subdirectory of his system to make it available.
2. Update the locally installed client on the workstation. The administrator must transfer the binary DER file of the certificate to each user that has a locally installed client. The user of that machine must then run the Certificate Management Utility from the local Host On-Demand installation. For security purposes, it is recommended that the binary DER file and the password be sent via separate out-of-band methods.

Using Microsoft cryptographic database

The Microsoft cryptographic database contains a much larger list of recognized CAs than does the Host On-Demand WellKnownTrustedCAs.class file. You may use this cryptographic database to authenticate server certificates. This database may be used in addition to the WellKnownTrustedCAs.class and CustomizedCAs.class files.

If you are running on a Windows system and you are using certificates from CAs that are not in the WellKnownTrustedCAs.class file that exists in the Microsoft cryptographic database, then you may wish to use this method for authenticating the server to avoid administrative overhead. If your CA is not listed in the cryptographic database or you are using a self-signed certificate, you may add it to that database. Follow the instructions that are provided by the browser.

12.3 Certificate management utility

At the time this book was published, the IBM certificate management utility, a Java application, is available on the following platforms:

- ▶ Windows NT and Windows 2000 Server
- ▶ AIX

The zSeries administrators may use gskkyman or RACF to manage their certificates. Refer to 3.6, “Using SSL with Communications Server for z/OS” on page 105 for specific information for the zSeries platform.

Servers on platforms other than Windows, AIX and zSeries have no Host On-Demand utility for adding unknown CAs to the HODServerKeyDb.kdb file. There is a Java utility that may be used to insert the server's public key certificate into the CustomizedCAs.class file, keyrng. Refer to 12.5, “Keyring utility” on page 509 for details on this utility.

The following sections demonstrate the usage of the certificate management utility for:

- ▶ Requesting a certificate from an unknown CA and making it available to the clients.
- ▶ Creating a self-signed certificate and making it available to the clients.

12.3.1 Starting the certificate management utility

OS/390 or z/OS users should refer to 3.6.3, “SSL Configuration using gskkyman” on page 109.

- ▶ To start the certificate management utility on Windows click **Start -> Programs -> IBM Host On-Demand -> Administration -> Certificate Management**.
- ▶ Before running the certificate management utility on AIX, you must be in the /hostondemand/bin directory, and the JAVA_HOME environment variable must be set to the full path of your Java installation. For example, if the default installation options were chosen and your Java system is installed in /usr/JDK1.1.8, you would run the certificate management utility by doing the following:

```
cd /usr/opt/hostondemand/bin
export JAVA_HOME=/usr/JDK1.1.8
CertificateManagement
```

The graphical interface on Windows and AIX is the same.

The first window you will see is shown in Figure 12-1.

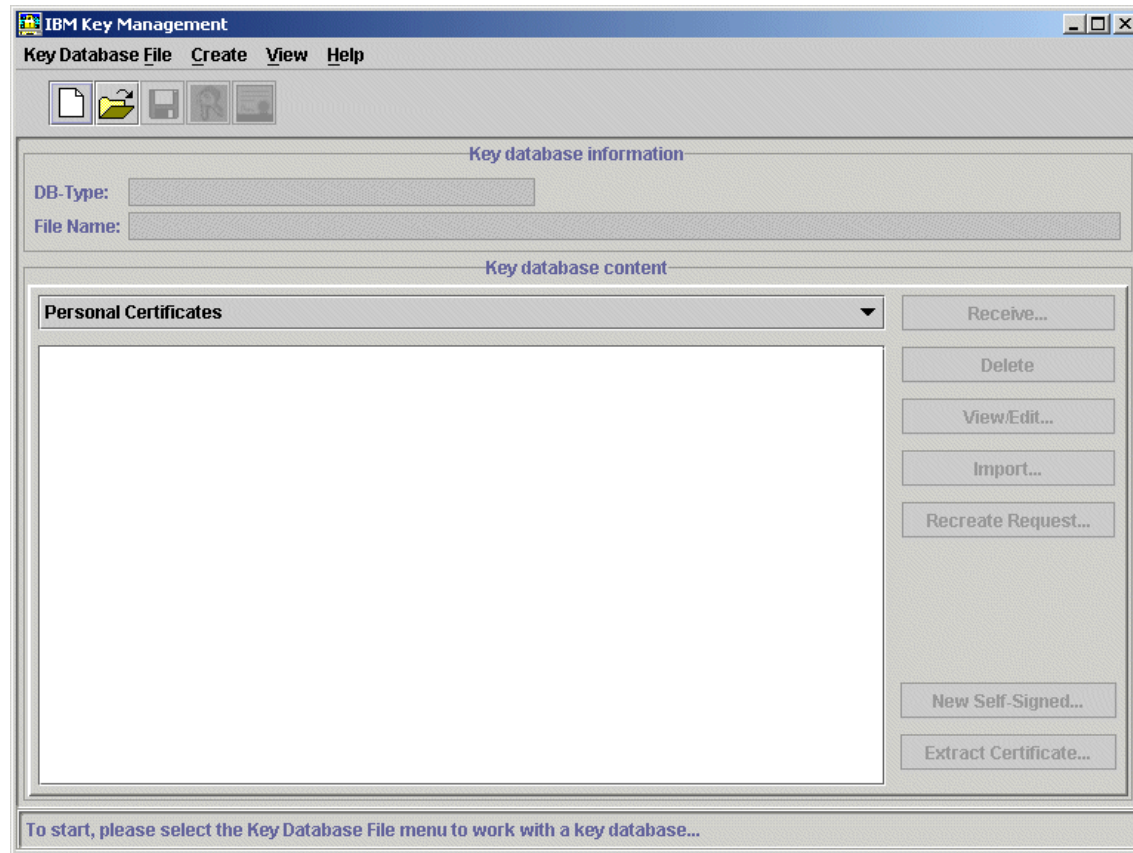


Figure 12-1 Certificate management utility

12.3.2 Create a request for an unknown CA

1. Click **KeyDatabaseFile** -> **New**.
2. Accept the CMS keyring database file, using HODServerKeyDb.kdb for the file name and \hostondemand\bin for the location. Click **OK**.
3. You may be asked if you want to replace an existing file. Click **Yes**.
4. The Password window appears. Enter your password twice and, if you wish, set an expiration time.
5. Select **Stash the Password to a file**. This will cause the password to be held, encrypted, in \hostondemand\bin\HODServerKeyDb.sth. The Host On-Demand server needs to be able to access it at runtime. Click **OK**.
6. Your Certificate Management window will look like Figure 12-2.

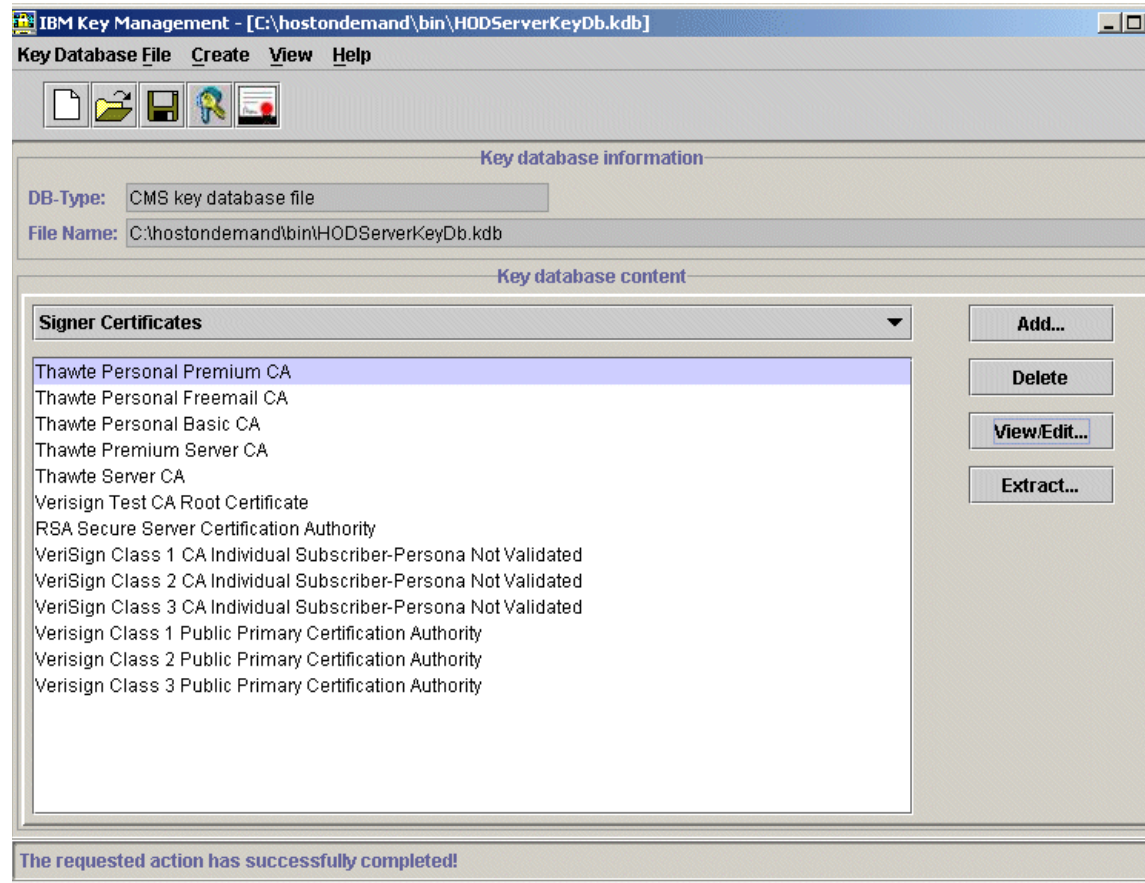


Figure 12-2 Create new certificate request - start

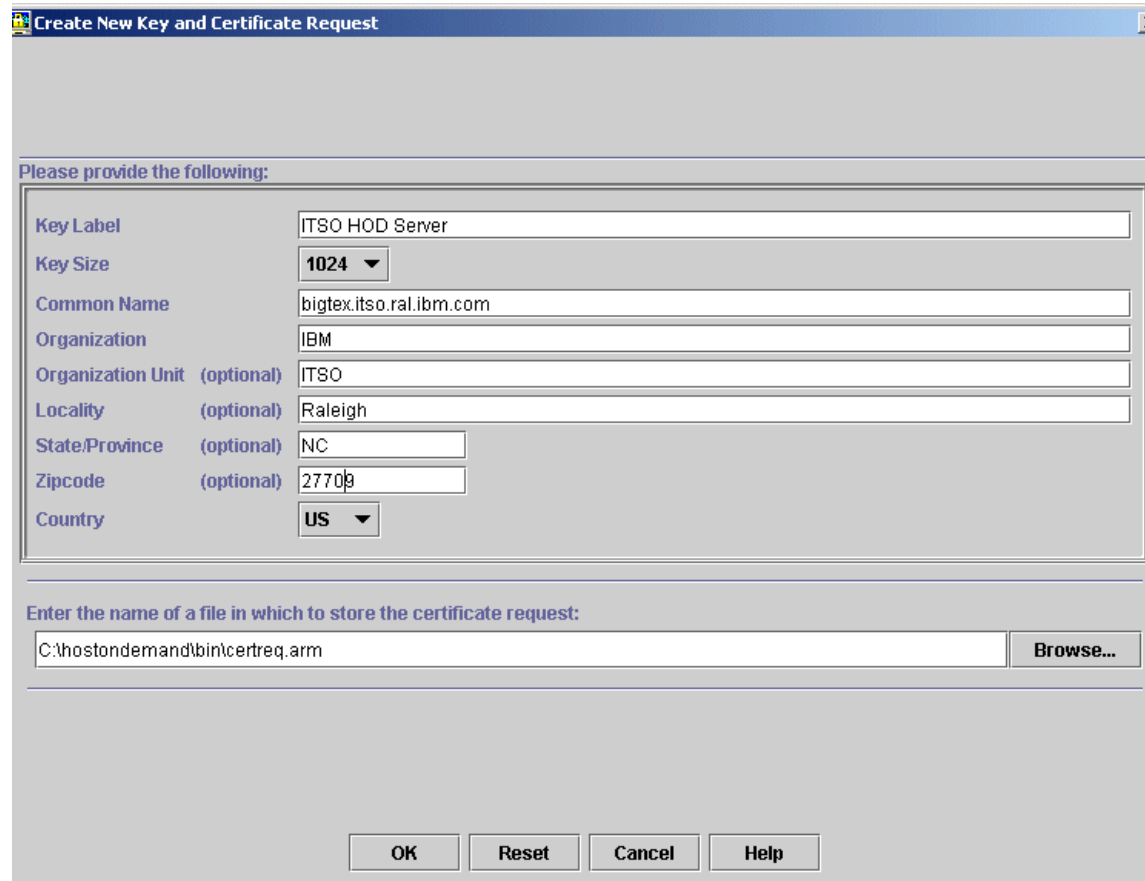
7. Select **Personal Certificate Requests** from the drop-down list.
8. Click **Create** and select **New Certificate request**. The Create New Certificate Request window appears.
9. Use Table 12-1 to enter your data in the appropriate fields and enter a name and location for this file, in our example \hostondemand\bin\certreq.arm.

Table 12-1 Certificate request information

Field name	Value
Key Label	This field is for identification use only, so use a meaningful text string.
Version	Use X509 V3.

Field name	Value
Key Size	This should default to the encryption level of your installation. The larger value is more secure, but you must take into account the capabilities of your browsers to decrypt.
Common Name	This should be the fully qualified DNS name of the server. It will be used if any client requests server authentication, and when resolved by the DNS server it must match the IP address the client uses to establish the session.
Organization	Fully identify the name of your organization, for example IBM Corp.
Organizational Unit	This optional field can further identify the server or department operating the server within the organization, for example ITSO.
Locality	This optional field should contain the city where the server is located.
State/Province	This is an optional parameter.
Zipcode	This is an optional parameter. Some Netscape browser versions have been known to crash when this field is used; therefore, it is recommended that you omit this field.
Country	This will default to the native country code.

When complete, the window should look like Figure 12-3.



The dialog box is titled "Create New Key and Certificate Request". It contains a section titled "Please provide the following:" with the following fields:

Key Label	ITSO HOD Server
Key Size	1024
Common Name	bigtex.itso.ral.ibm.com
Organization	IBM
Organization Unit (optional)	ITSO
Locality (optional)	Raleigh
State/Province (optional)	NC
Zipcode (optional)	27709
Country	US

Below this section is a field for the file name: "Enter the name of a file in which to store the certificate request:". The text "C:\hostondemand\bin\certreq.arm" is entered, and a "Browse..." button is to the right.

At the bottom are four buttons: "OK", "Reset", "Cancel", and "Help".

Figure 12-3 Create new certificate request

10. Click **OK**. You will see a window similar to the one shown in Figure 12-4.



Figure 12-4 Create new certificate request - completed

11. Note the file name and click **OK**.

12. Your certificate will appear in the list of Personal Certificate Requests.

13. Start a Web browser and access the CA's Web page. Follow the instructions provided to submit the certificate request. The following list provides the URLs of CAs:

- VeriSign

<http://www.verisign.com>

- Thawte

<http://www.thawte.com>

Depending on the CA you choose, you can either e-mail the certificate request just generated or incorporate the certificate request into the form or file provided by the CA.

While you are waiting for the CA to process your certificate request, you can enable SSL security for controlled testing purposes only by using a self-signed certificate as described in 12.3.5, "Create a self-signed certificate" on page 500.

12.3.3 Receive the CA's certificate

When the certificate that was created in 12.3.2, "Create a request for an unknown CA" on page 493 has been signed and returned by the CA, you must receive it.

If your CA is already in your list of trusted CAs, you can skip directly to 12.3.4, "Receive the certificate signed by the CA" on page 498. To check this:

1. Click **KeyDatabaseFile** -> **Open** after you have started the certificate management utility as described in 12.3.1, "Starting the certificate management utility" on page 492.
2. Select the file, probably HODServerKeyDb.kdb in \hostondemand\bin, and click **Open**.
3. Select **Signer Certificates** from the drop-down list.
4. If there are no entries with the name of that CA, click **Add** and the window shown in Figure 12-5 will appear.

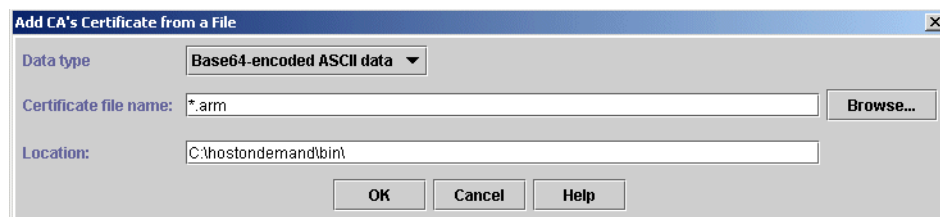


Figure 12-5 Receive CAs certificate

5. Enter the name of the file you have received and click **OK**.
6. Enter a meaningful label at the next window, for example TrustAuthorityCA and click **OK**.
7. The CA will be added to the list of Signer Certificates as shown in Figure 12-6.

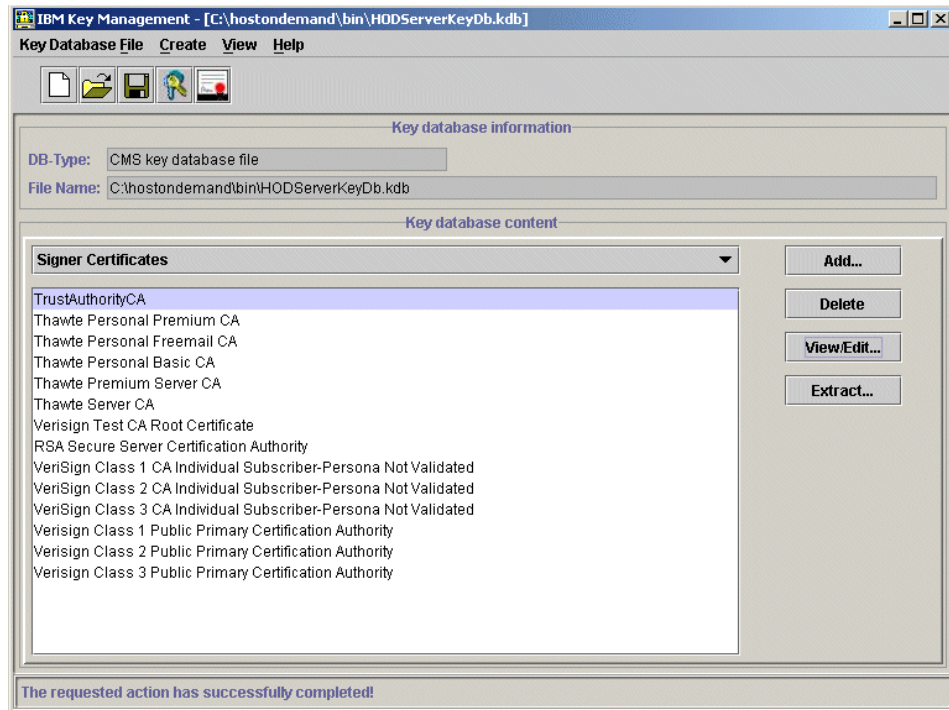


Figure 12-6 Receive CAs certificate - done

12.3.4 Receive the certificate signed by the CA

1. Assuming you have already opened the CMS keyring database file x:\hostondemand\bin\HODServerKeyDb.kdb, select **Personal Certificates** from the drop-down list.
2. Click **Receive** and the window shown in Figure 12-7 will appear.

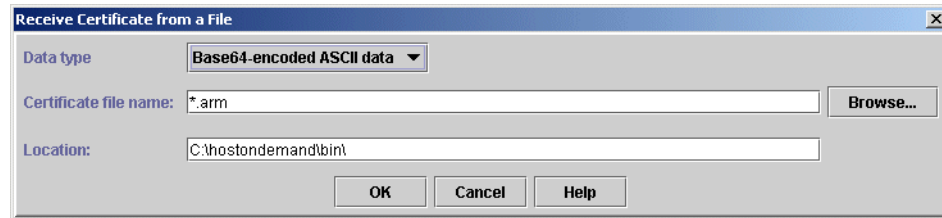


Figure 12-7 Receive signed certificate

3. Enter the name of the file you have received and click **OK**.
4. The CA will be added to the list of Personal Certificates as shown in Figure 12-8.

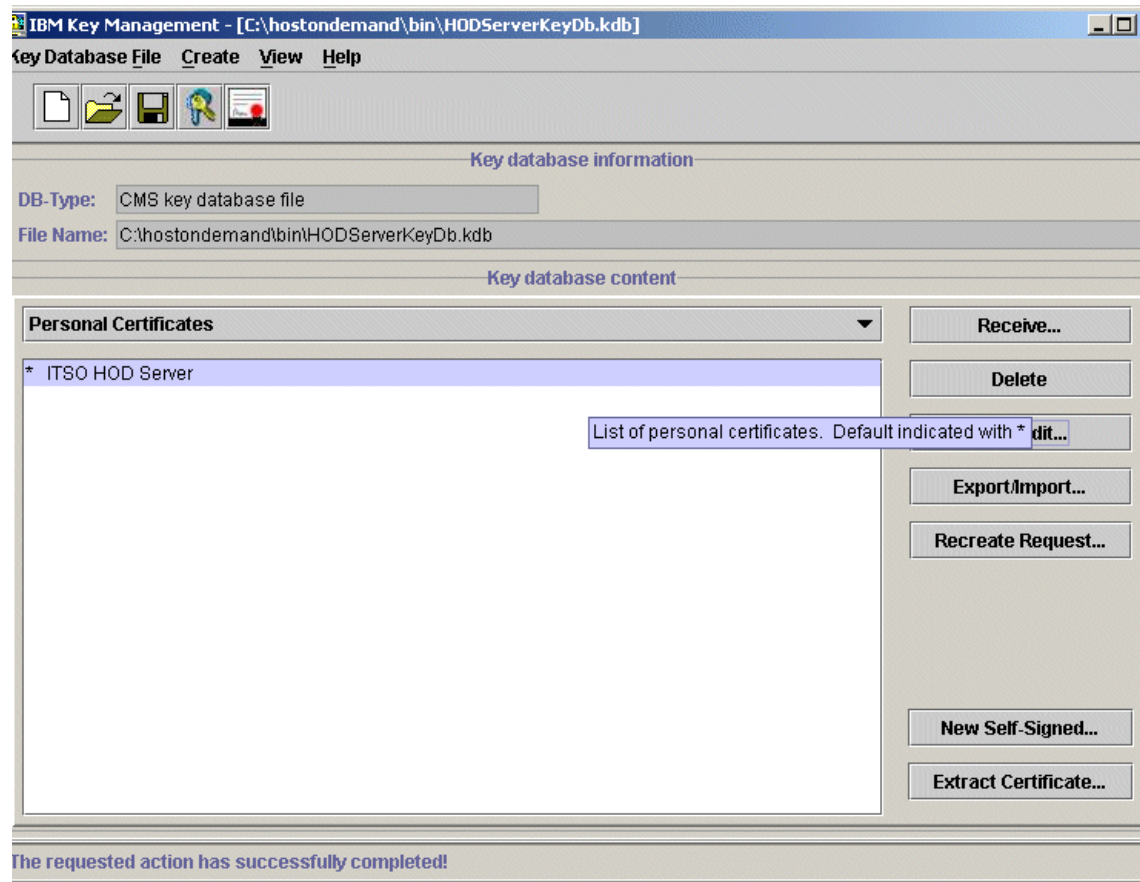


Figure 12-8 Receive signed certificate - done

5. In order for the Host On-Demand service manager to use this new certificate, you must stop and restart the Host On-Demand service manager.
6. Continue with 12.3.6, "Make a certificate available for the clients" on page 503.

12.3.5 Create a self-signed certificate

1. After you have started the utility, click **KeyDatabaseFile** -> **New**.

Important: If you want to use a self-signed certificate while you are waiting for a signed certificate to be returned from a CA, click **KeyDatabaseFile** -> **Open** instead, do *not* create a new database file. Skip to step 6.

2. Accept CMS keyring database file, HODServerKeyDb.kdb for the file name and x:\hostondemand\bin for the location. Click **OK**.
3. You may be asked if you want to replace an existing file. Click **Yes**.
4. The Password window appears. Enter your password twice and, if you wish, set an expiration time.
5. Select **Stash the Password to a file**. This will cause the password to be held, encrypted, in \hostondemand\bin\HODServerKeyDb.sth. The Host On-Demand server needs to be able to access it at runtime. Click **OK**.
6. Your Certificate Management window will look like Figure 12-9.

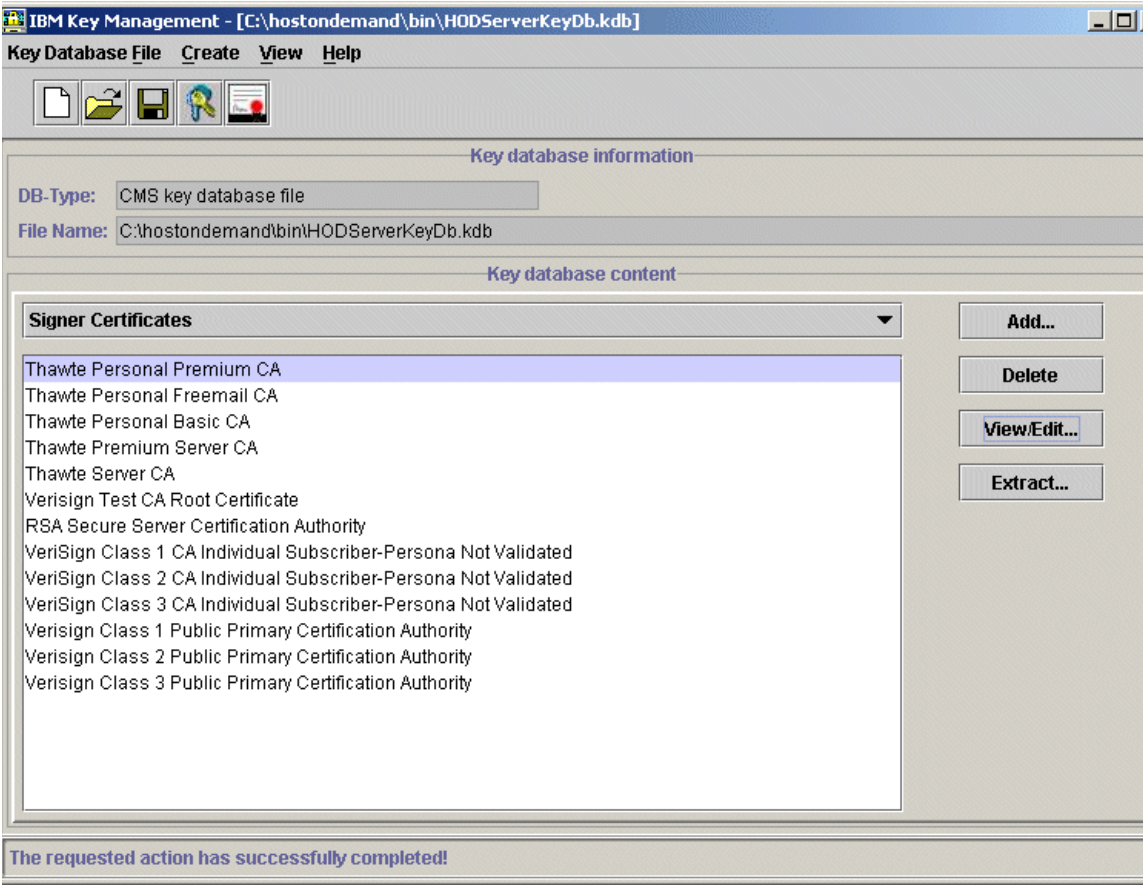


Figure 12-9 Create new self-signed certificate - start

- 7. Select **Personal Certificates** from the drop-down list.
- 8. Click **Create** and select **New Self-signed Certificate**. The Create New Self-Signed Certificate window appears.
- 9. Use Table 12-2 to enter your data in the appropriate fields.

Table 12-2 Self-signed certificate information

Field name	Value
Key Label	This field is for identification use only, so use a meaningful text string.
Version	Use X509 V3.

Field name	Value
Key Size	This should default to the encryption level of your installation. The larger value is more secure, but you must take into account the capabilities of your browsers to decrypt.
Common Name	This should be the fully qualified DNS name of the server. It will be used if any client requests server authentication, and when resolved by the DNS server it must match the IP address the client uses to establish the session.
Organization	Fully identify the name of your organization, for example IBM Corp.
Organizational Unit	This optional field can further identify the server or department operating the server within the organization, for example ITSO.
Locality	This optional field should contain the city where the server is located.
State/Province	This is an optional parameter.
Zipcode	This is an optional parameter. Some Netscape browser versions have been known to crash when this field is used; therefore, it is recommended that you omit this field.
Country	This will default to the native country code.
Validity	The maximum recommended value is 365 days.

10. When complete, the window should look like Figure 12-10.

Create New Self-Signed Certificate

Please provide the following:

Key Label	ITSO HOD Server
Version	X509 V3 ▼
Key Size	1024 ▼
Common Name	bigtex.itso.ral.ibm.com
Organization	IBM
Organization Unit (optional)	ITSO
Locality (optional)	Raleigh
State/Province (optional)	NC
Zipcode (optional)	27709
Country	US ▼
Validity Period	365 Days

OK Reset Cancel Help

Figure 12-10 Create new self-signed certificate - completed

11. Click **OK**. Your certificate will appear in the list of Personal Certificates.
12. Use View/Edit to check that the values are correct and that the certificate is the default.
13. In order for the Host On-Demand service manager to use this new certificate, you must stop and restart the Host On-Demand service manager.
14. Continue with 12.3.6, "Make a certificate available for the clients" on page 503.

12.3.6 Make a certificate available for the clients

1. Start the Certificate Management utility on the Host On-Demand server and open the keyring database file, HODServerKeyDb.kdb, which is located in the \hostondemand\bin directory.
2. For a certificate from an unknown CA, select **Signer Certificates** from the drop-down list.
For a self-signed certificate, select **Personal Certificates** from the drop-down list.

Your window will look like Figure 12-11.

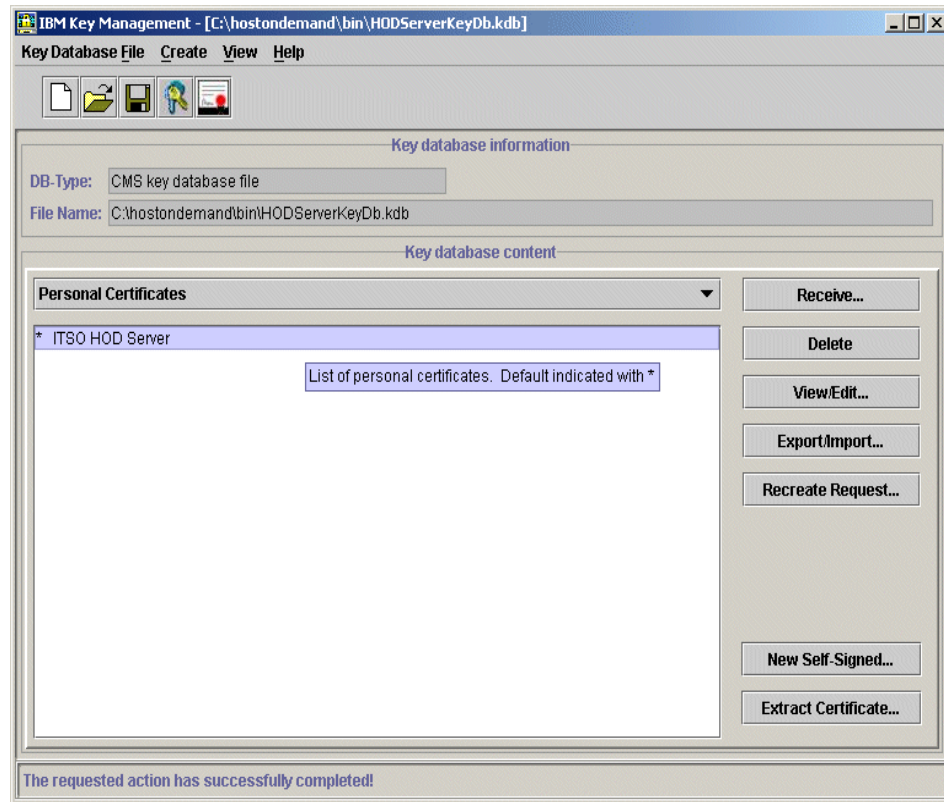


Figure 12-11 Distribute self-signed certificate - selection

3. Highlight the certificate you want to make available for the clients.
4. Click **Extract Certificate** and the Extract Certificate to a File window appears.

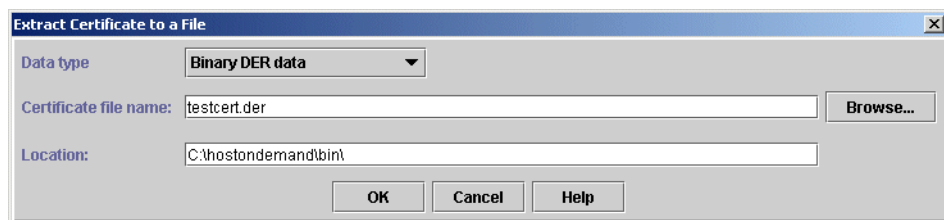


Figure 12-12 Extract certificate

5. Set the **Data type** to Binary DER.
6. Select a name and location to store this file, for example
 \hostondemand\HOD\private\testcert.der.

7. Click **OK** and the file will be created.
8. Close the keyring database file.
9. Click **KeyDatabaseFile** -> **New**.
10. Select **SSLLight key database class** from the Key database type drop-down list, accept CustomizedCAs.class for the file name and x:\hostondemand\HOD\ for the location. Click **OK**.
11. You may be asked if you want to replace an existing file. In that case, click **Yes**.
12. For a certificate from an unknown CA, select **Signer Certificates** from the drop-down list and click **Add**.

For a self-signed certificate, select **Personal Certificates** from the drop-down list and click **Receive**.
13. Select the Binary DER file from the Data type pull-down.
14. Enter the file name and location where you stored it, in our example \hostondemand\HOD\private\testcert.der, or click **Browse** to locate it.

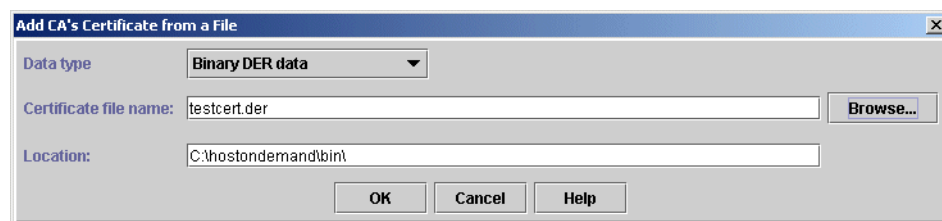


Figure 12-13 Adding a Self-signed certificate

15. Click **OK** to insert the certificate.
16. For a self-signed certificate, skip to step 17.

Enter a meaningful label, for example TrustAuthorityCA and click **OK**.
17. The certificate will be added to the selected list.
18. Save and close the CustomizedCAs.class file.

Note: You may also create and update the CustomizedCAs.class file using the Windows locally installed Host On-Demand client and copy the file to the published directory of the server.

12.4 The Certificate Wizard

The Certificate Wizard is a Java application, introduced in Host On-Demand Version 6, which helps the Administrator to create and maintain certificates and the associated databases.

For an example of how this is done, we will create a self-signed certificate and make it available for the clients, as we did in 12.3.5, “Create a self-signed certificate” on page 500 and 12.3.6, “Make a certificate available for the clients” on page 503, by using the Certificate Wizard.

12.4.1 Using the Certificate Wizard

1. On a Windows NT or Windows 2000 server, click **Start -> Programs -> IBM Host On-Demand -> Administration -> Certificate Wizard** and the Certificate Wizard introduction window shown in Figure 12-14 will appear.

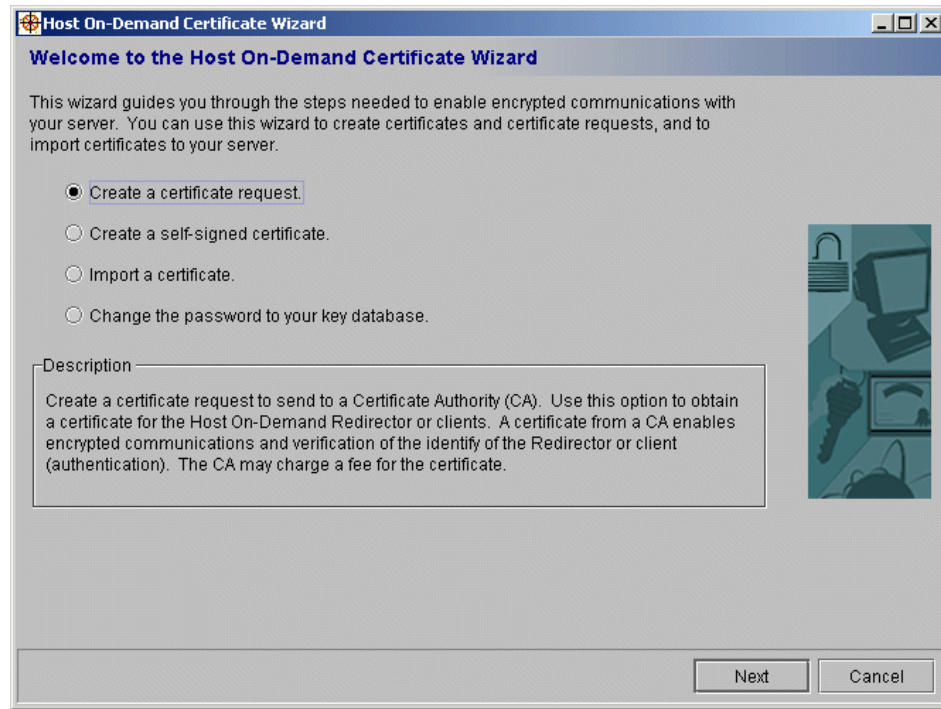
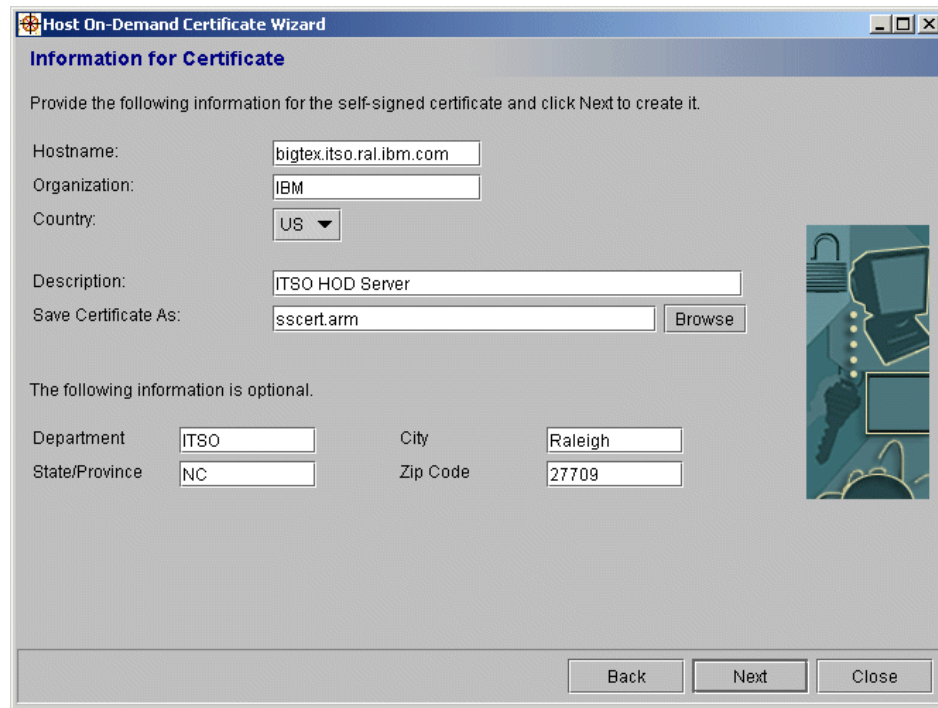


Figure 12-14 Certification Wizard Introduction window

The Change the password to your database option is only available if the wizard finds the file HODServerKeyDb.kdb in /[install]/bin/.

2. Select **Create a self-signed certificate** and click **Next**.
3. On the next window, select **Server** to specify the location of the database and click **Next**.
4. Enter your password and click **Next**.
5. On the next window, enter your data in the appropriate fields by using Table 12-2 on page 501. After that, your window will look as shown in Figure 12-15.



The image shows a Windows-style dialog box titled "Host On-Demand Certificate Wizard". The main heading is "Information for Certificate". Below this, a message states: "Provide the following information for the self-signed certificate and click Next to create it." The form contains several input fields: "Hostname:" with the value "bigtex.itso.ral.ibm.com", "Organization:" with the value "IBM", "Country:" with a dropdown menu showing "US", "Description:" with the value "ITSO HOD Server", and "Save Certificate As:" with the value "sscert.arm" and a "Browse" button. Below these, a section titled "The following information is optional." contains four more input fields: "Department" (ITSO), "City" (Raleigh), "State/Province" (NC), and "Zip Code" (27709). On the right side of the dialog, there is a vertical graphic showing a computer monitor, a keyboard, and a padlock. At the bottom right, there are three buttons: "Back", "Next", and "Close".

Hostname:	bigtex.itso.ral.ibm.com		
Organization:	IBM		
Country:	US		
Description:	ITSO HOD Server		
Save Certificate As:	sscert.arm		Browse
The following information is optional.			
Department	ITSO	City	Raleigh
State/Province	NC	Zip Code	27709

Figure 12-15 Certificate Wizard - Information

6. Click **Next**, the certificate will be created, it will be made available for the clients and the window shown in Figure 12-16 will appear.

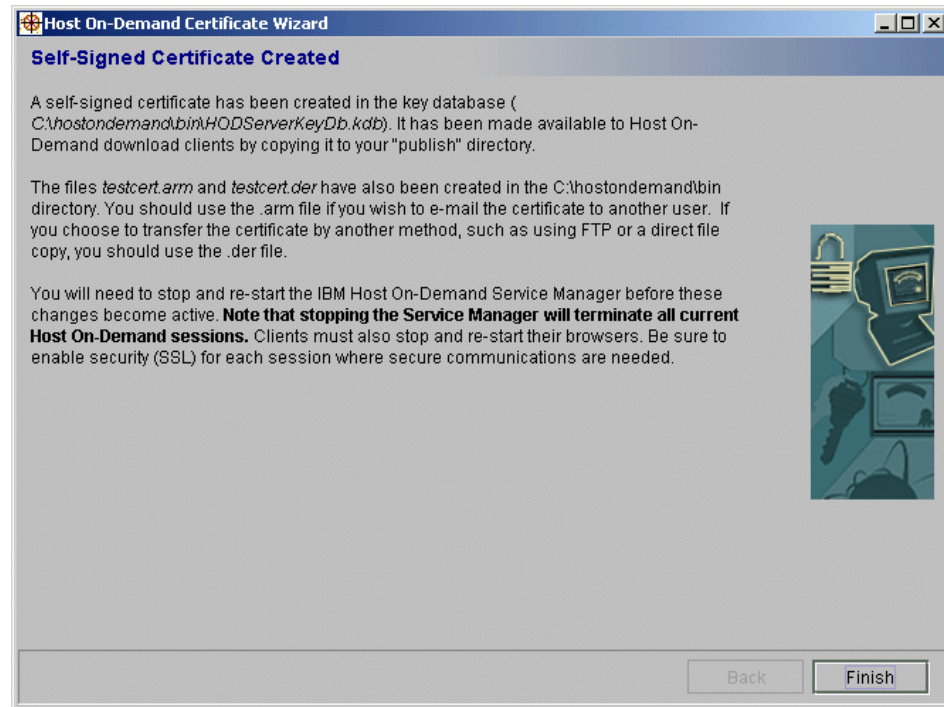


Figure 12-16 Certificate Wizard - Done

7. Click **Finish**.

12.5 Keyring utility

The keyring utility is used to obtain a site or server certificate from a Telnet Server (or Redirector) and install it into the `CustomizedCAs.class` file. It is provided for this express function for servers that do not have the certificate management utility. This utility is a Java keyring utility that is shipped with Host On-Demand. This is a lengthy command and it is easy to make errors. On iSeries and Windows systems a script file is shipped; however, you can create your own script file. The example below was created for the zSeries server that was discussed in "Make certificates available to clients" on page 119. We created a shell script, `javakeyrng`, with the command so it could be reissued if needed. The backslash is a continuation character; otherwise the command must be on one continuous line. The command for Java 1.3 is:

```
java -classpath ./usr/lpp/HOD/hostondemand/lib/sm.zip \
com.ibm.hodsslght.tools.keyrng CustomizedCAs connect ipaddr:port
```

Where `ipaddr` is the address of your TN3270 telnet server, and `port` is the SSL port you wish to connect.

The command for Java 1.1.8 is:

```
java -classpath ./usr/lpp/HOD/hostondemand/lib/sm.zip:$CLASSPATH \
com.ibm.hodsslight.tools.keyrng CustomizedCAs \
connect ipaddr:port
```

You will be prompted to enter the password for `CustomizedCAs.class` file. There is no way to make the password available to clients at runtime, so you *must not* give a password, just press Enter. The results of the command will look similar to the Java 1.3 example shown in Example 12-1.

Example 12-1 Java keyring utility output

```
CASEY @ SC48:/usr/lpp/HOD/hostondemand/HOD>javakeyrng
Password for CustomizedCAs.class:
Connecting to 9.12.6.126:6623
com.ibm.hodsslight.SSLException
    at com.ibm.hodsslight.SSLConnection.certificate(SSLConnection.java:979)
    at com.ibm.hodsslight.SSLClient.serverCertificate(SSLClient.java:272)
    at com.ibm.hodsslight.SSLClient.handshake(SSLClient.java:110)
    at
com.ibm.hodsslight.SSLConnection.handleData(SSLConnection.java(Compiled Code))
    at
com.ibm.hodsslight.SSLRecordLayer.receiveRecord(SSLRecordLayer.java:695)
    at com.ibm.hodsslight.SSLConnection.install(SSLConnection.java:212)
    at com.ibm.hodsslight.SSLClient.<init>(SSLClient.java:719)
    at com.ibm.hodsslight.SSLSocket.install(SSLSocket.java:117)
    at com.ibm.hodsslight.SSLSocket.<init>(SSLSocket.java:260)
    at com.ibm.hodsslight.tools.keyrng.main(keyrng.java)
com.ibm.hodsslight.SSLException
time created=Wed Aug 29 16:52:27 EDT 2001
category=4 TRUSTPOLICY
error=1017 PEERCERTIFICATECHAINNOTTRUSTED
int1 =0
e=null
```

----- Server Certificate Chain -----

Site Certificate - Number 0

```
Key : RSA/512 bits
Subject: wtsc48oe.itso.ibm.com, Research Triangle Park, ITS0, IBM, US
Issuer: ITS0Raleigh, Raleigh, ITS0, IBM, US
Valid from: Mon Aug 27 10:53:17 EDT 2001
Valid to: Tue Aug 27 11:03:17 EDT 2002
Finger print: 2A:84:BA:46:C0:73:7C:4F:6D:98:AD:B1:44:72:BA:F8
```

```
-----
Enter the number of the certificate to be added to CustomizedCAs.class (q to
quit): 0
Adding the Site Certificate - 0 to CustomizedCAs.class
Done.
```

When executing in a Java 1.3 environment, we found that the flow of execution changes, since Java 1.3 classifies certain exception conditions different from Java 1.1.8. The result is that program flow under Java 1.3 follows a different exception handling path, a path not traversed under Java 1.1.8. In any case, the Java exception shown in Example 12-1 can be ignored. You can verify the certificate was added to the CustomizedCAs.class file using the following command:

```
java -classpath ./usr/lpp/HOD/hostondemand/lib/sm.zip \
com.ibm.hodssligh.tools.keyrng CustomizedCAs verify
```

The add option does not require the TN3270 server to be available, since no socket call is issued. You have to specify the name of the certificate file as one of the input parameters. The command when using Java 1.3 is:

```
java -classpath ./usr/lpp/HOD/hostondemand/lib/sm.zip \
com.ibm.hodssligh.tools.keyrng CustomizedCAs \
add --certificatetype certificate.name
```

where --certificate type is either ca if you are adding a CA root certificate or site if you are adding a site or self-signed certificate. The certificate.name is the fully qualified name of the actual certificate, for instance, /u/casey/itso.cer.

The command for Java 1.1.8 is:

```
java -classpath ./usr/lpp/HOD/hostondemand/lib/sm.zip:$CLASSPATH \
com.ibm.hodssligh.tools.keyrng CustomizedCAs \
add --certificatetype certificate.name
```




Deployment strategies

IBM WebSphere Host On-Demand can provide cost effective and secure browser-based host access to users in both intranet and extranet-based environments. Host On-Demand is installed on a web server, simplifying administrative management and deployment. The Host On-Demand applet is downloaded to the client's browser providing user connectivity to critical host applications and data.

This chapter discusses issues involved with deploying Host On-Demand. Careful consideration of these topics is important for successful deployment:

1. Configuration models
2. Client type - cached or download
3. Client preload and Java considerations
4. Security requirements
5. Base platform(s) for Host On-Demand server

We advocate making these decisions with the goal of minimizing Host On-Demand administration and support requirements. This chapter will examine the factors that affect these choices and why a particular choice or choices would be made.

13.1 Host On-Demand configuration models

Default HTML files are shipped with Host On-Demand. Before clients can connect to sessions via the default pages, the administrator must logon to the Administration utility and configure user ID and session information. Some of the features available via the Deployment Wizard are not available with the default HTML pages. For example, using the Deployment Wizard the administrator can select the level of client Java or tailor the preload components to be downloaded to clients. As it is recommended to use the Deployment Wizard, we will now review the configuration models available.

The Deployment Wizard requires the administrator to choose one of three configuration models. Choosing how the Host On-Demand server will communicate configuration information to the Host On-Demand client involves understanding the three types of configuration models available:

- ▶ Configuration server-based model
- ▶ Combined model
- ▶ HTML-based model

It may be that using only one model will not work for all your Host On-Demand users. Using the Deployment Wizard, Chapter 14, “Deployment Wizard” on page 529, the administrator can easily select the configuration model to be deployed to the Host On-Demand client.

When deciding which one of the three configuration options to use you should consider the following:

- ▶ Do your users expect their individual preferences to be migrated with them from workstation to workstation?
- ▶ As the Host On-Demand administrator, how do you expect to access, manage and change Host On-Demand user IDs and groups?
- ▶ Can you configure all the firewalls in your network to allow the passing of Host On-Demand configuration information between the workstation and Host On-Demand configuration server?
- ▶ Will there be performance issues on the Host On-Demand server caused by the number of users you have defined logging into the Host On-Demand server and accessing their personalized configurations?
- ▶ Do you require one of the following services that require the Host On-Demand Service Manager, Redirector, License Use Management or OS400 Proxy support?

13.1.1 HTML-based model

In the HTML-based model, session information is contained in HTML files created by the Deployment Wizard. Changes made by the end user, such as changing a screen color or keyboard mapping (if allowed), are stored locally on the user's machine. The Host On-Demand Service manager/configuration server is not used to store or manage configuration data. After the end user has accessed the customized HTML page for the first time, information is then stored locally on the client's workstation. As a consequence, if the Host On-Demand server becomes unavailable but the web server is available, the end user will be able to connect to their host sessions. This model does not require a firewall port to be opened to access the configuration server.

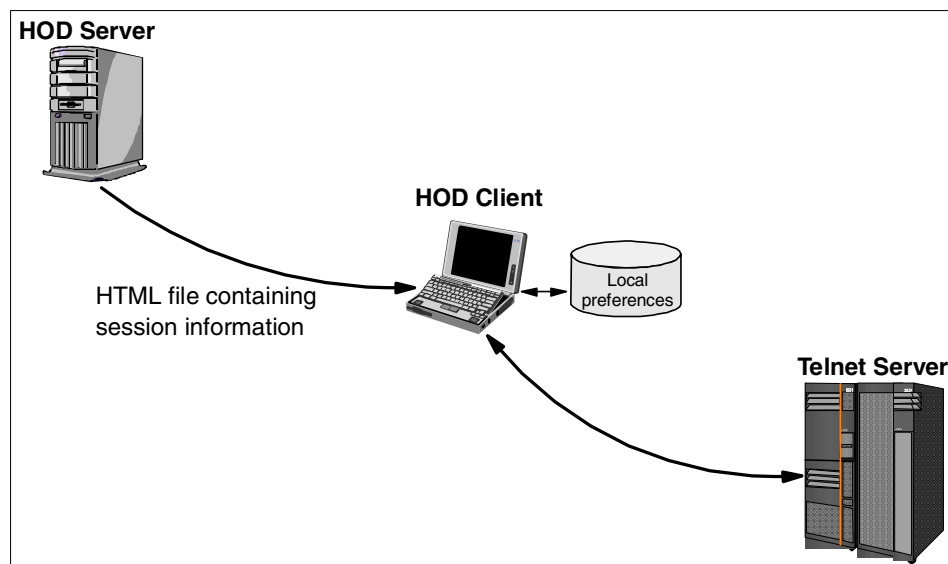


Figure 13-1 HTML-based model

Note: If License User Management is enabled, the client will attempt to communicate with the Host On-Demand Service manager. If there is an intervening firewall, it will need to be configured to allow access to the Host On-Demand Service Manager (port 8999 by default).

With the HTML-based model, the administrator sets up the initial session configuration windows and may choose to lock certain properties to prevent users from changing these fields. The administrator can also allow users to make session changes but not allow changes to be saved beyond the current session.

If the end user sets up personalized changes and the administrator later updates the customized HTML file via the Deployment Wizard, these updates will be merged with the updates made by the end user. However, if an end user has changed a field, changes stored in the end user's local file will override the updates made by the administrator. The administrator may override a user setting by setting the field value and locking the field.

Advantages:

- ▶ Access to the configuration server is not required (unless the Host On-Demand services, such as the Redirector, OS/400 proxy, or license use management, are enabled). Improved performance on the server may be realized, since these users are not accessing personalized configuration data.
- ▶ There is no need to open a separate port on the firewall as the configuration server is not accessed.
- ▶ There is no need to create and maintain Host On-Demand user IDs.
- ▶ End users are not required to logon to Host On-Demand.
- ▶ Performance may be better than using the configuration server because you do not have large numbers of users accessing personalized configurations on the configuration server.

Limitations:

- ▶ If users make changes to their personal configurations, those changes may not be available on other workstations without physically copying the files or placing them on a shared file system where they may be accessible from another workstation.

Note: The HTML-based model is the default model used by the Deployment Wizard.

13.1.2 Configuration server-based model

In this model, session configuration data is stored and managed on the Host On-Demand configuration server by the administrator using the Host On-Demand Administration Utility. End users must logon to the Host On-Demand configuration server with a user ID. A user ID may be shared among multiple end users. However this is not recommended if the administrator is going to allow end users to save their user preferences. User IDs can also be grouped together if they share common configuration data. It may be easier to maintain configurations at a group level, especially if there is a large user base.

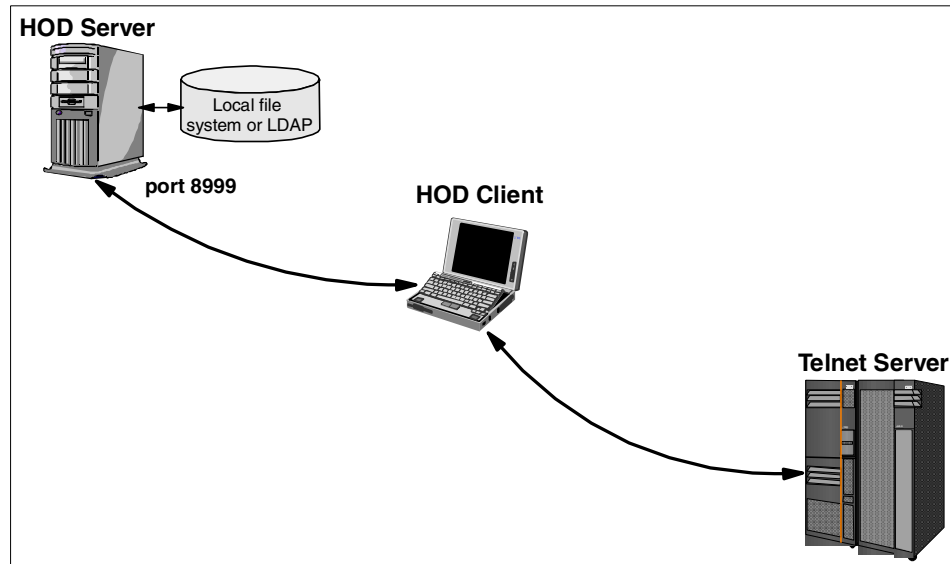


Figure 13-2 Configuration server-based model

If there is a firewall between the HOD server and the HOD client, the port used to communicate with the Host On-Demand configuration server (default is port 8999) must be opened in the firewall. If the Host On-Demand Configuration Servlet (refer to Chapter 9, “Configuration Servlet” on page 397) is implemented, the port to the configuration server is not required.

Individual user preferences within a group are maintained with the user ID. This implies that if a user ID is a member of more than one group, each group may have group-specific session data but the user-specific data remains constant across the groups. This also means that if you have multiple HOD users logging onto HOD with the same user ID, allowing them to save preferences may cause conflicts as each user tries to save their own preferences. In this situation it is best to disallow users from saving preferences.

Advantages:

- ▶ Users can access their own personalized configuration data from any machine that has network access to the configuration server.
- ▶ Administrators can maintain the sessions from local or remote sites.
- ▶ Users may be able to be organized into groups, which simplifies administration.
- ▶ On the Windows, AIX and z/OS platforms, Native Authentication may be used to reduce the number of userids and passwords users are required to

remember. See 11.11, “Native Authentication” on page 468 for additional details.

Limitations:

- ▶ Administrative overhead of maintaining the configuration server user IDs, and groups.
- ▶ Performance implications if a large number of users are accessing the configuration server.
- ▶ Users must logon to Host On-Demand
- ▶ Requires firewall port to Configuration Server to be opened or use of the Host On-Demand Config Servlet
- ▶ Must maintain and manager User IDs

Integrated Windows Domain Logon

For cases where the administrator chooses to have user settings stored on the Host On-Demand server, the user creation process can be automated if the clients are running on a Microsoft Windows operating system. If this option is selected, each user ID is identified to Host On-Demand by the user's Windows Domain user name. Users are not prompted for a user name or password, but instead, the Windows domain security is relied upon for authentication. When this option is selected, the administrator specifies a default group for users that have not chosen to customize any settings. The first time a user customizes a setting, that user becomes defined on the Host On-Demand server and the customizations are stored.

13.1.3 Combined model

In the combined model, the administrator sets up a group on the configuration server with the required sessions (similar to the configuration server-based model). The Deployment Wizard is then used to create the HTML file specifying the group on the configuration server that will be accessed to obtain session information. Access to session information on the Host On-Demand configuration server is required the first time a client accesses a combined model HTML file. Like the configuration server-based model, if there is a firewall between the HOD server and the HOD client, the port used to communicate with the Host On-Demand configuration server (default is port 8999) must be opened in the firewall. If the Host On-Demand Configuration Servlet is implemented, the port to the configuration server is not required.

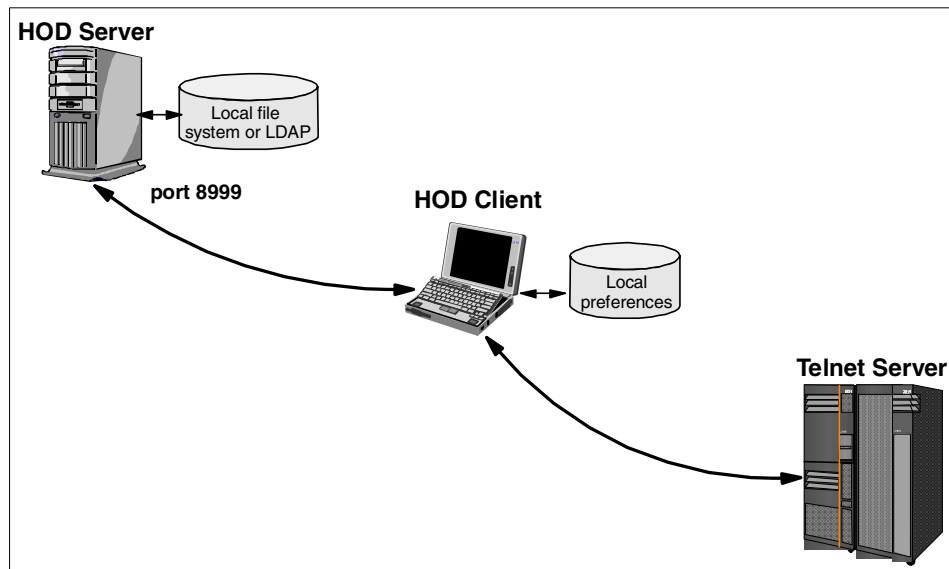


Figure 13-3 Combined model

After the initial access, if allowed by the administrator, user preferences are stored on the user's local machine (similar to the HTML model) and the Host On-Demand Service manager is not required, although if it is not running, users will see the screen as shown in Figure 13-4.

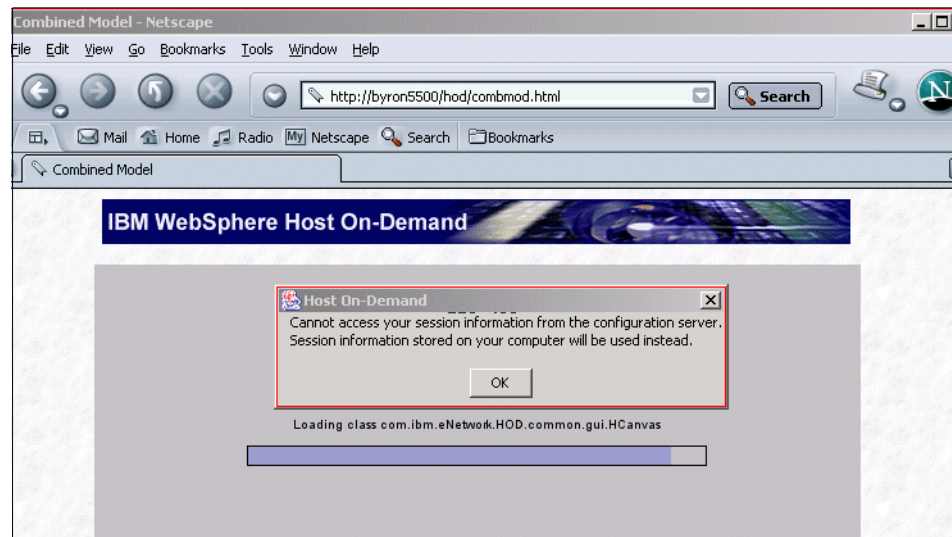


Figure 13-4 Combined model - no Service Manager

Advantages:

- ▶ At least one group must be created on the configuration server but there is no requirement to create or maintain user IDs.
- ▶ Clients will attempt to access the configuration server before each use of Host On-Demand, but if the Configuration Server is unavailable, Host On-Demand clients will run using saved copies of the session configuration information.
- ▶ Administrators can update the group session information locally or remotely, and have it deployed to all clients the next time Host On-Demand is used. The updates are merged with the user's preferences. Any fields changed by the administrator will override the user preference. However, user-specified preferences are maintained for fields not changed by the administrator.
- ▶ On the Windows, AIX and z/OS platforms, Native Authentication may be used to reduce the number of userids and passwords users are required to remember. See 11.11, "Native Authentication" on page 468 for additional details.

Limitations:

- ▶ Configuration information on a local machine is not available to other machines unless it is physically copied or by placing them on a shared file system where they may be accessed by another workstation.
- ▶ May increase administration overhead if there are a large number of "default" groups. Since there are no user IDs, the administrator must be aware of what "default" group a user is mapping to. Any changes to the group information will affect all users whose HTML files specify that group.
- ▶ Requires firewall port to Configuration Server to be opened or use of the Host On-Demand Config Servlet

Note: It is possible to for some users to access the configuration server and other users to implement the HTML-based or Combined model. The administrator selects the model when using the Deployment Wizard.

13.1.4 User Preferences

The administrator can restrict the settings that a user can change by locking the field. If the administrator does not lock a field but **Do not save preferences** is checked, changes made by the end user will not persist when they close the host session. The administrator can, after the initial deployment, change a default setting. Provided a user has not modified the setting, the new value will take

effect the next time the user starts the Host On-Demand session. To override any settings that have been modified by the user, the administrator can set the value for the field and lock the field. The new value will take effect the next time the user starts the Host On-Demand session.

You may wish to consider the following when deciding to allow users to save preferences:

- ▶ Are individual user preferences required at all or can you provide default settings that will meet the needs of your end users?
- ▶ Do you wish to have user preferences stored on the configuration server or stored locally on end user's machines?
- ▶ Will users be moving between different workstations and require their user preferences to follow them?

Your answers to these questions should assist you in determining if you need to allow users to save preferences.

Some examples of customizations that may be required by end users are:

- ▶ select a specific LU name or workstation ID
- ▶ defining macros
- ▶ Personal color or keyboard settings

Hint: Host On-Demand V7 provides the ability to code HTML overrides that allow a server side program to dynamically set various client side parameters. See Chapter 16, "Modifying session properties dynamically (HTML overrides)" on page 583. This new functionality may reduce your need to permit individual user preferences since you can now programmatically change user settings based on such criteria as IP address of the user.

13.2 Host On-Demand emulator clients

The HTML files created by the Deployment Wizard and loaded into the client browser launch the Host On-Demand emulator client. There are three different types of Host On-Demand emulator clients available:

- ▶ Download Client
- ▶ Cached Client
- ▶ Function On-Demand

Some of the issues to consider before deciding which emulator client to use are:

- ▶ Level of Java installed in your client's browsers

- ▶ Network connection speeds
- ▶ Users access to multiple Host On-Demand servers using different levels of Host On-Demand code
- ▶ Type of Host On-Demand configuration model you wish to deploy

More detailed information on these clients can be found in Chapter 5, “Clients” on page 163. That chapter also discusses several issues of concern when deploying these Host On-Demand clients to your users.

Specifically, as the HOD administrator you should be aware of these issues and how they will effect users of HOD clients:

- ▶ 5.8, “Improvements to Java 2 support” on page 188
- ▶ 5.9, “Java 2 practical issues” on page 193
- ▶ 5.10, “Client Java type: Java 1, Java 2, or Auto Detect” on page 199
- ▶ 5.12, “Download client and cached client implementation” on page 207
- ▶ 5.15, “Web browsers: Java 1 and Java 2 enabled” on page 221

13.3 Security requirements

One of the most fundamental considerations in building a secure Host On-Demand environment is to remember its web-based nature. Restricting access to the Host On-Demand web pages via web server security is the first line of defense.

Having registered users, where they are required to log in to use the HOD service, is not primarily for security, but to aid in administration, and also for storing of the user’s preferences. Client authentication can be used to provide a high-security environment with standard PKI tools, and in concert with an established security management framework (for example, RACF).

The Telnet server is the next line of defense. SSL can be used in order to prevent frame examination, and non-standard Telnet ports should be used to discourage discovery of the Telnet port you are using.

In its simplest form, Host On-Demand is simply a standard TN3270, TN5250 or VT client. While they may be suitable for intranet use, it’s important to note that these standard terminal types (whether they are Host On-Demand or other software) cannot be considered secure. TN3270, for example, passes all data in the clear. A simple frame trace of TN3270 traffic on a network is enough to recover data, user ID information, and even mainframe passwords. Host

On-Demand was the first Telnet emulator to offer SSL-encrypted 3270 or 5250 sessions, which would make a frame trace reconstruction of user data virtually impossible. This was first offered with Host On-Demand Version 4. With the use of the Host On-Demand Redirector, it is even possible to encrypt VT sessions.

In addition to the server authentication, where the client authenticates the server as valid, client authentication is also available. By enabling client authentication, Host On-Demand can be restricted to only those clients with a valid certificate.

Host On-Demand also has additional functions that build on top of the SSL-enablement: Native Authentication, express logon and Telnet-negotiated security. While these are not security functions, they build on the security already in Host On-Demand and can make a secure environment easier for the end user and the administrator. Here are the security functions available in Host On-Demand client:

- ▶ Delivery of the HTML, applets, and preferences via HTTPS.
- ▶ SSL-enabled host sessions (native TN3270, TN5250 or VT by use of the Redirector).
- ▶ Client authentication (requires the user to have a digital certificate recognized by the Telnet server).
- ▶ Telnet-negotiated security, the ability to negotiate a secure connection over the same port as a non-secure Telnet session (see 11.13, “Telnet-negotiated security” on page 483).
- ▶ Express logon requires SSL session with client authentication, and automatically logs the user into the zSeries host application without any additional prompts (see 11.8, “Express Logon Feature” on page 455).)

Each has certain infrastructure requirements that must be met. In deciding how much security is “enough,” the security needs must be balanced with the infrastructure and administrative requirements.

Using a separate Host On-Demand and/or Telnet server for high-security users is also a common solution for extranet and Internet environments. The security policies of many companies prohibit a direct connection from the Internet to their mainframe business systems. These companies establish servers in a secure segment of their network called the demilitarized zone (DMZ) that provides a buffer between the Internet and the operational systems inside. With Host On-Demand, the same security principles apply. So, a Host On-Demand server is set up inside the DMZ to serve extranet and internet users. Usually, this is an SSL-based (HTTPS) web server. Access to this server is usually restricted by some form of logon, with direct access first being authenticated by a gateway server of some kind.

If a Host On-Demand server is placed within a DMZ, then a Telnet server is the next requirement. The placement of a Telnet server within a DMZ is one solution. Having SSL enabled on the Telnet server is required in order to make it secure. As an alternative, Telnet traffic can also be redirected through the firewall via proxy servers, the Host On-Demand Redirector (only for low-volume traffic) or by using the Telnet proxy function of the IBM Communications Server for AIX or Linux.

The final stop for Host On-Demand security is the target host itself. Generally, these are well-protected machines with highly evolved security mechanisms such as RACF. At this point, it is the user that becomes the weakest link in the chain. It is distinctly possible that in order to get to a host application, an extranet user may have to know:

1. A user ID and password to an external web site.
2. A Host On-Demand user ID and password.
3. A RACF user ID and password.
4. In some cases, it's possible that users may even have to log on to individual applications.

A user faced with this gauntlet is likely to record this information somewhere: a spreadsheet, a text file or even the famous yellow sticky on the display terminal or under the keyboard.

13.3.1 Firewall considerations

When you allow external users outside your company intranet access to Host On-Demand and Host On-Demand's emulator capabilities, you need to configure your company's firewall to allow for this remote access. You will need at least one port, the telnet port, to be open on the firewall. The Host On-Demand configuration server port is optional.

If you do not use the Configuration Servlet (see Chapter 9, "Configuration Servlet" on page 397), port 8899 on the firewall must be opened and directed to the Host On-Demand server port 8899. This port is only needed if you are using the Configuration Server-based model, License User Management (LUM) or Combined Server model. See "Host On-Demand configuration models" on page 514.

A firewall is your gateway from the Internet to the intranet for Host On-Demand users on the Internet side of the firewall. Remember to configure your Host On-Demand clients to use either the IP address of your firewall or DNS name of your firewall when setting the destination address for the emulator session. This is because the firewall blocks access to your internal IP addresses. For example, see Figure 13-5.

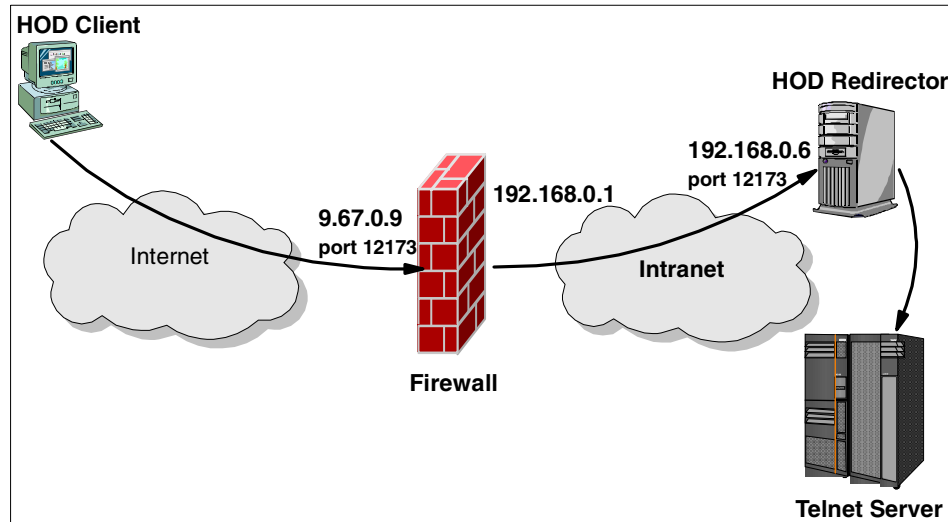


Figure 13-5 Network with firewalls

The IP address for your Host On-Demand redirector is 192.168.0.6 using port 12173 on the company intranet. The firewall address is 9.67.0.9 on the Internet. The destination address for the Host On-Demand sessions would be 9.67.0.9 and port 12173.

The firewall must be setup to forward all traffic that comes to it on port 12173 to 192.168.0.6 port 12173.

If you do not configure the firewall and Host On-Demand clients properly, the Host On-Demand clients will NOT connect to your intranet resources.

13.4 Host On-Demand Server Platform choices

Host On-Demand server code is designed to be platform independent, needing little more than a standard Java runtime environment and a web server to provide its function. However, with the introduction of such features as Native Authentication, express logon and support for portal server, the choice of server platform may be determined by the desire to use such features.

With Host On-Demand, the choice of the server platform is influenced by a combination of factors including:

- ▶ Number of Host On-Demand clients
- ▶ Reliability and availability requirements
- ▶ Security
- ▶ User location

If you have a large client base and users which require 24 x 7 access to the configuration server, these factors may influence the platform choice for the Host On-Demand server. Typically, installations with a large number of end users are installed on high availability systems such as UNIX-based platforms or the z/OS platform.

Security considerations

Security can dictate platform choice for Host On-Demand. There are two general factors that can influence platform choice with security. First is the use of Host On-Demand for extranet users. It is generally considered prudent to avoid having an extranet user with a direct Telnet connection from the Internet to a company's main systems. Given this rule, it is common practice to have a Host On-Demand server (with Telnet disabled) for extranet users running inside the company's DMZ. This server will usually run on either an Intel-based server (Windows NT or Windows 2000) or a UNIX server.

Even for a customer who will run Host On-Demand on a primary mainframe platform such as zSeries or iSeries, security concerns will often dictate the use of a second (distributed) platform.

Second, the use of the Host On-Demand Redirector to provide telnet SSL can also dictate the use of the base platform for Host On-Demand. SSL is supported only on Windows NT and AIX Redirectors and the use of SSL can impose a significant load on a server and must be considered carefully. Telnet negotiated security can also be used if connecting to a Telnet server that supports this option.

Third, the use of Host On-Demand's Native Authentication can also dictate the server platform choice. Native Authentication is dependent on the underlying operating system for validation of the user ID and password. If a company needs to validate Host On-Demand user IDs against RACF, that dictates deployment on a zSeries platform. Conversely, if a company wants to validate against a Windows NT domain structure, this would dictate deployment on a Windows NT server platform. Since Native Authentication is a Host On-Demand configuration Server deployment model, both cases would dictate a careful review of the need of registered Host On-Demand users.

13.4.1 User locations

Many of today's businesses are global. Even smaller regional businesses often need to communicate with business partners or suppliers from around the country if not around the world. Also, today's business world is very fluid; an environment where mergers and acquisitions are part of everyday life can result in some very interesting network campus arrangements. So, it is possible that geography can also play a large role in the Host On-Demand deployment strategy. In general, geography affects:

1. Campus groupings
2. WAN links
3. Time zones
4. Country (language) considerations

The first two factors are often interrelated. It is not unusual to find corporations spread across several major campuses that span their country or countries. Often each campus is serviced by individual "farms" of distributed systems. For example, a company that has offices in Philadelphia, Chicago and San Francisco will likely use separate Windows NT domain servers at each site. Furthermore, it's not unusual to see larger distributed platforms (for example, UNIX and mainframe servers) at several locations. For example, a company may have offices in five locations in various places in the United States, but have major data centers in only two of the locations.

Given a company's geographical situation, how can this affect their deployment of Host On-Demand? First, the geographical dispersion of the Host On-Demand user community needs to be taken into account when estimating the server size or possible traffic added by Host On-Demand clients accessing the Host On-Demand Server. IBM performance testing has indicated that Host On-Demand adds workload to a server under the following conditions:

- a. When a user logs on to a Host On-Demand server configured for LDAP or Native Authentication.
- b. During either the initial cached client download, or an update to the cached client, and for each invocation of the download client. The ability to tailor the client using componentization can significantly reduce the client download size, since only the required modules are downloaded.

The issue here is how the interactions between deployment choices and geography can impact Host On-Demand. In short, it's important to understand the geographical dispersion of users, since their use will skew the load on the Host On-Demand (and Telnet) server(s). For example, if a company is planning to support 3,000 total users, but they are spread across three time zones in four locations, due to the geographical constraints, it's likely the load on the server will

be balanced for each time zone. Also, if the deployment choice is made to use a HTML based configuration server model that does not require Host On-Demand users to logon to Host On-Demand server, then the load on the server will be similar to that of a normal web server.

13.5 Other considerations

An additional environment worth mentioning is Microsoft Windows Terminal Services either separately or with Citrix Metaframe. It is possible to use Host On-Demand with this software, but keep in mind that running the Host On-Demand server and cached client on the same machine is not supported. Therefore, you can either install Host On-Demand on the Terminal Services server and have the users use the download client, or install Host On-Demand on a different machine and have the users use the Host On-Demand cached client. The last configuration is recommended.



Deployment Wizard

The Deployment Wizard is a tool that is used by the administrator to create and edit customized HTML files. These files can contain a variety of information, depending on the options specified in the Deployment Wizard. The customized files are read by the applet that is downloaded to clients.

This tool is a Java application that runs only on a Windows platform. It is automatically installed when you install Host On-Demand server on a Windows system. In Host On-Demand Version 7 the Deployment Wizard can be installed as a stand alone program on a Windows platform. The stand alone Deployment Wizard can be installed from either a Host On-Demand for Windows CD or by downloading the image from the Host On-Demand server. The resulting Deployment Wizard files can then be distributed to the system where Host On-Demand is installed.

Using the Deployment Wizard is not required for an end user to launch a session. A set of default HTML files is shipped with Host On-Demand and these can be used by the end user to launch a session, provided the administrator has previously used the Host On-Demand Administration Utility to configure a session for the end user. However, by using the Deployment Wizard the administrator is able to take advantage of features such as: selecting the client level of Java; and tailoring the Host On-Demand applet size.

14.1 Planning

Some users and administrators may need to use customized HTML created by the Deployment Wizard, while others will find the default configuration HTML sufficient. To decide whether or not to use the Deployment Wizard, consider the following:

- ▶ Do you require end users to save their user preferences locally?
- ▶ Do you wish to select the level of client Java support?
- ▶ As the administrator would you like to tailor the download size of the applet to clients?
- ▶ Do you wish to create a customized html template to be used as the Host On-Demand web page?
- ▶ Do you wish to modify the default port specified at install time to communicate with the Configuration Server?

If the answer to any of the above is Yes, you should consider using the Deployment Wizard to create custom HTML files.

14.2 Starting the Deployment Wizard

If you have installed Host On-Demand on a Windows platform start the Deployment Wizard by selecting **Start -> Programs -> IBM Host On-Demand -> Administration -> Deployment Wizard**.

To install the stand alone Deployment Wizard, insert the Host On-Demand Windows installation CD then click **Install Deployment Wizard** from the Host On-Demand Welcome window as shown in Figure 14-1.

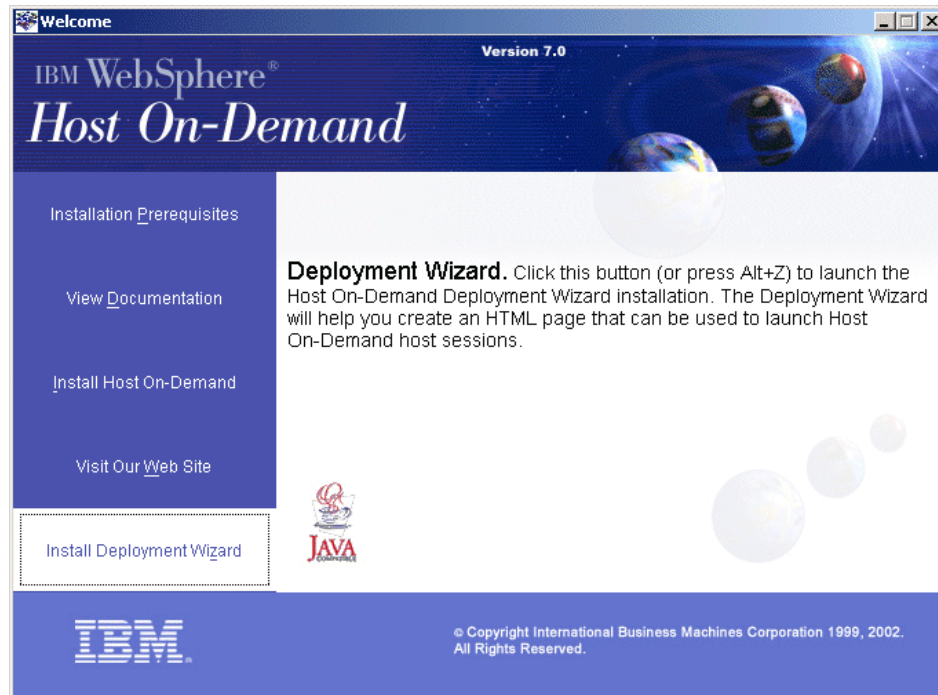


Figure 14-1 Host On-Demand Welcome

The Deployment Wizard can also be downloaded from your Host On-Demand server. Start your browser and point to the HODMain.html page. Click on **Deployment Wizard Installation Image for Windows** to download the file to your workstation. From the File Download window click **OK** to save the file to your workstation (see Figure 14-2).

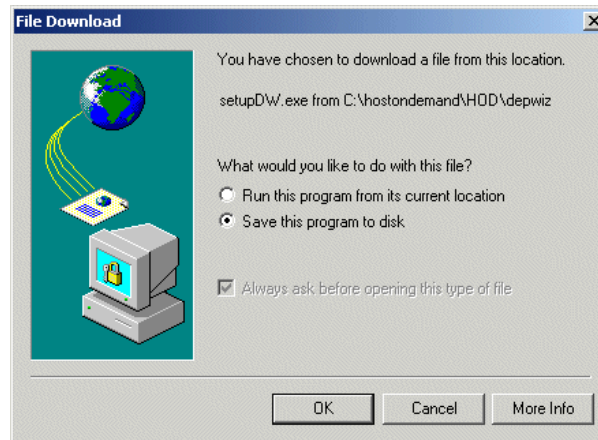


Figure 14-2 Deployment Wizard download

After saving setupDW.exe you can either run the program from the window as shown in Figure 14-3 or run **setupDW.exe**.

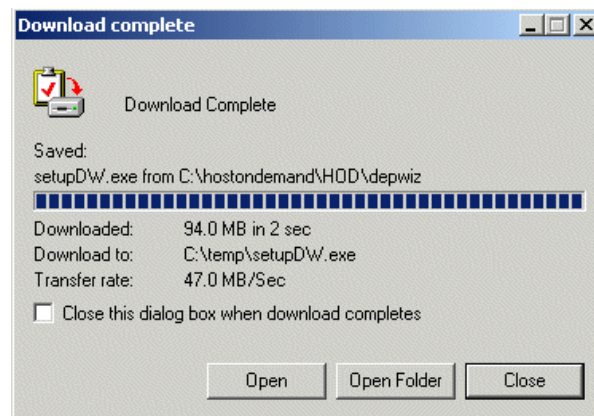


Figure 14-3 Deployment Wizard installation

Once installation is complete launch the Deployment Wizard from the **Start > Programs** desktop menu.

14.3 Using the Deployment Wizard

After starting the Deployment Wizard, the Welcome to the Host On-Demand Deployment Wizard window is displayed as shown in Figure 14-4.

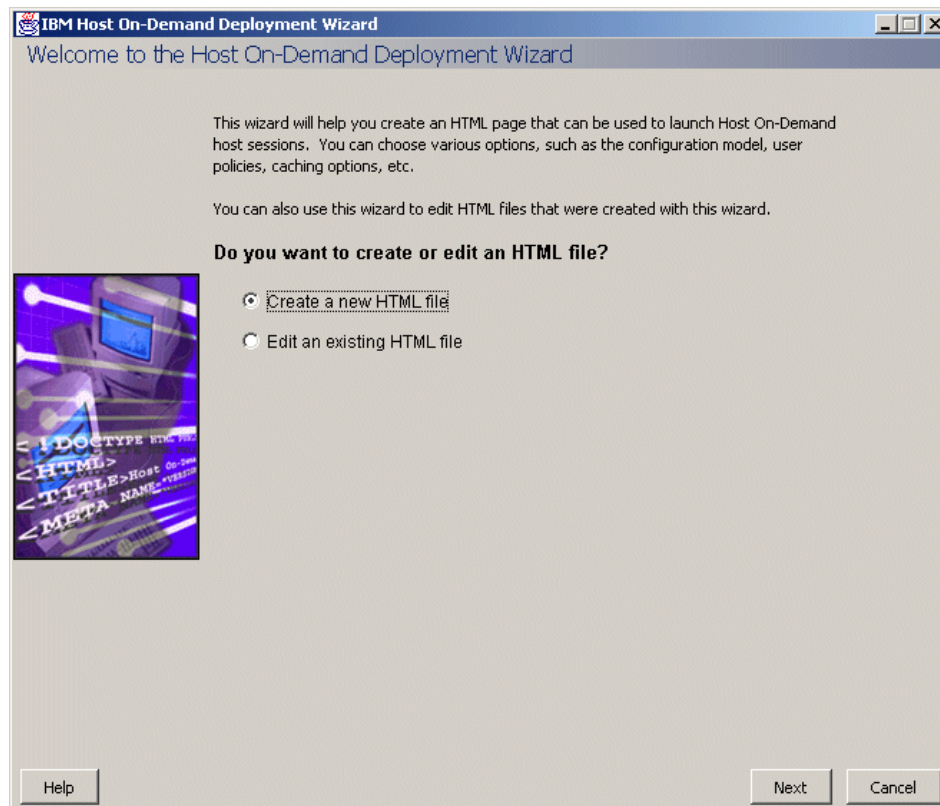


Figure 14-4 Deployment Wizard Welcome

The default setting is Create a new HTML file. If you want to edit existing HTML files that were created by the Deployment Wizard, select **Edit an existing HTML file**. It is recommended that you do not manually edit files created by the Deployment Wizard, nor should you use the Deployment Wizard to edit files that it did not create, doing so could render the file unusable.

If the server is on an iSeries system, see “Mapping a network drive to the iSeries” on page 162. A network drive connection allows custom HTML files to be opened and updated using the Deployment Wizard.

We will now demonstrate the use of the Deployment Wizard with an example illustrating how to create a custom HTML page for each of the configuration models.

14.3.1 HTML-based model example

With the HTML-based model the administrator does not have to logon to the configuration server to create or maintain Host On-Demand user IDs. All host session configuration information is contained in the files created by the Deployment Wizard. If the administrator allows users to save changes to their host session configuration settings, such as keyboard remappings, changes are stored on the user's local file system.

A description of the selected model is displayed in the center of the window as shown in Figure 14-5.

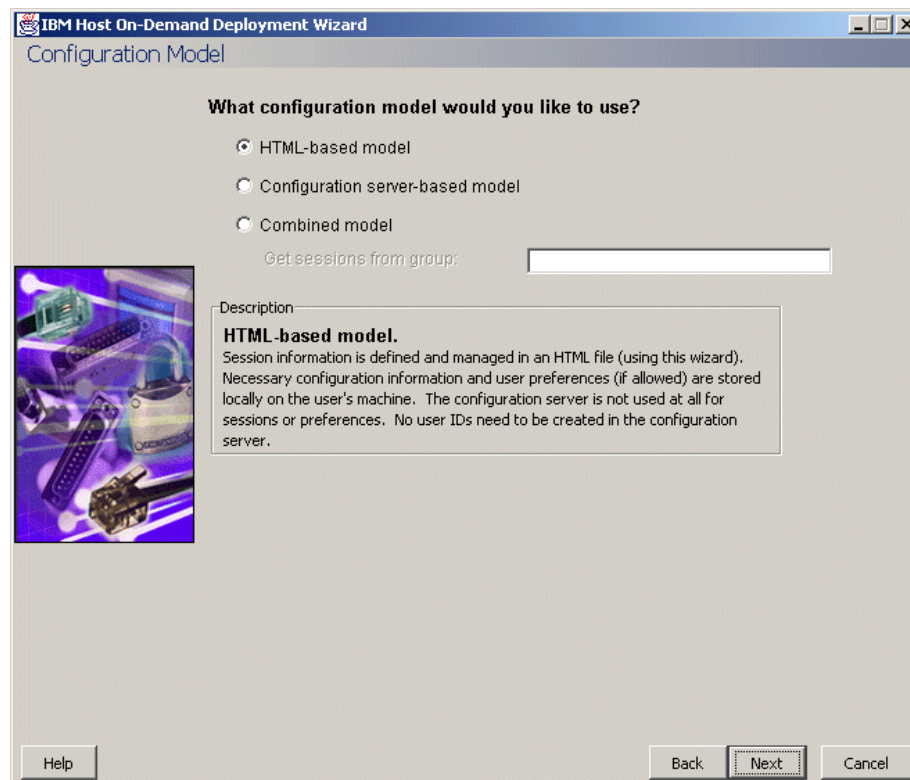


Figure 14-5 HTML-based model

Clicking **Next** will display the Host Sessions window. For this scenario, the administrator defines a basic 3270 session with a session name of "itso3270", specifies the destination address, and clicks **Add**. This adds the session to the table as shown in Figure 14-6.

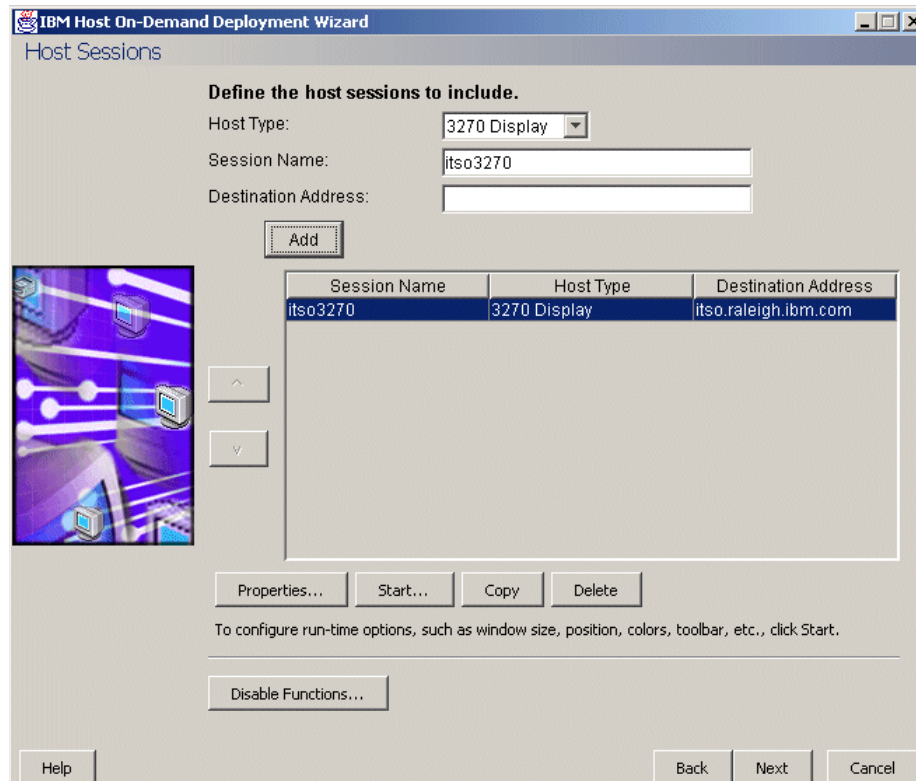


Figure 14-6 Host Sessions

More than one session can be added to the table if the administrator wants to provide access to other systems. A session icon will be created on the end user's client window for each session defined. If more than one session is created icons can be re-ordered by highlighting the session and clicking on the **up** or **down** arrow. However, once a user has accessed the HTML page the initial order will override subsequent changes made by the administrator.

Note: It is possible to add a very large number of sessions but there can only be a maximum of 26 sessions opened concurrently by a single user. This number can be further limited by the Maximum number of concurrent sessions per user field on the Advanced Options window (see Figure 14-12 on page 542).

To set a session's configuration properties, highlight the session and click **Properties**. This will display a properties window unique to the type of session being defined. For a 3270 Display session, the window shown in Figure 14-7 is displayed.

The image shows a dialog box titled "itso3270" with several tabs: "Connection", "Advanced", "Proxy Server", "Security", "Language", and "Screen". The "Advanced" tab is active. The dialog contains the following fields and controls:

- Session Name:** Text field containing "itso3270". To its right is an unchecked "Lock" checkbox.
- Destination Address:** Text field containing "itso.raleigh.ibm.com". To its right is a checked "Lock" checkbox.
- Destination Port:** Text field containing "23". To its right is a checked "Lock" checkbox.
- Enable SLP:** Radio buttons for "Yes" and "No", with "No" selected. To its right is a checked "Lock" checkbox.
- TN3270E:** Radio buttons for "Yes" and "No", with "Yes" selected. To its right is a checked "Lock" checkbox.
- LU or Pool Name:** Empty text field. To its right is a checked "Lock" checkbox.
- Screen Size:** Dropdown menu showing "24x80". To its right is an unchecked "Lock" checkbox.
- Host Code-Page:** Dropdown menu showing "037 United States". To its right is a checked "Lock" checkbox.
- Associated Printer Session:** Empty dropdown menu. To its right is an unchecked "Lock" checkbox.
- Close Printer With Session:** Radio buttons for "Yes" and "No", with "No" selected. To its right is an unchecked "Lock" checkbox.
- File Transfer Type:** Dropdown menu showing "Host File Transfer". To its right is an unchecked "Lock" checkbox.

Below the "File Transfer Type" dropdown is a button labeled "File Transfer Defaults...". At the bottom of the dialog are four buttons: "OK", "Cancel", "Keyboard...", and "Help".

Figure 14-7 Session properties

This window is similar to the window described in 7.1.7, "Configuring sessions" on page 285 where the fields are described in detail. The Deployment Wizard online help provides a description of each field. The online help also details fields and tabs that have been added in Host On-Demand Version 7 such as the Proxy Server tab. Note that some fields are locked by default, for example, Destination Address and Destination Port fields. It is recommended to lock these fields because if they are altered by a user they could affect the operability of the session. Administrators however may change these settings if they desire.

After configuring the session properties and clicking **OK**, control is returned to the Host Sessions window shown in Figure 14-6 on page 535.

The administrator can provide initial settings for runtime options (such as color or key remappings and screen size), by clicking **Start**. Doing so will start a host session, verifying connectivity for the session, and allowing the administrator to alter the runtime options. After the runtime options are modified, the administrator will close the session, storing the preferences. These preference settings will apply only to the highlighted session. For this scenario, the settings will not be modified.

Clicking **Copy** will copy the highlighted session creating a new session. The new session will have the name *n:xxxxx*, where *n* is an integer beginning with 1 and *xxxxx* is the name of the original session. When a session is copied, the session configuration attributes are copied as well as the runtime options.

Clicking **Delete** will remove a session from the table and all of its associated configuration data.

Clicking **Disable Functions** allows the administrator to restrict selected functions from the end users. In this scenario, the administrator is disabling all macro functions as shown in Figure 14-8. Disabling a function prevents the client from being able to view the function.

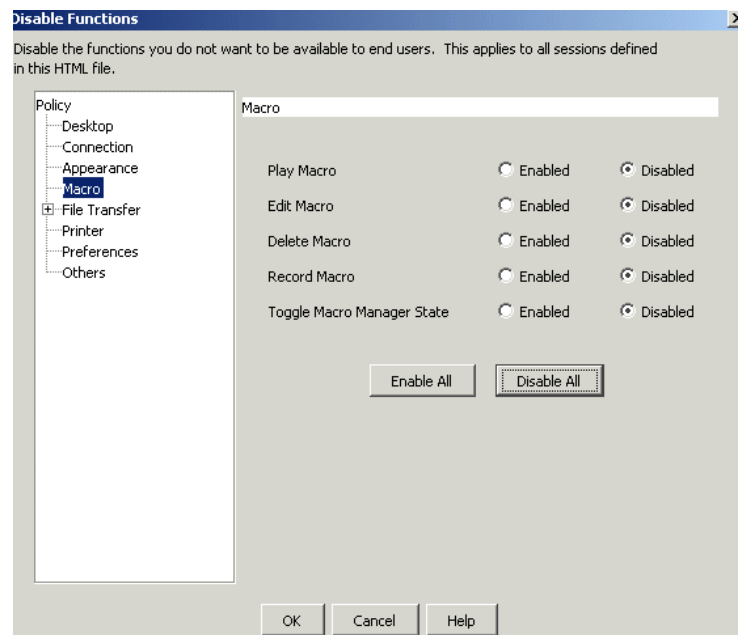


Figure 14-8 Disable functions

Select the various policy categories to locate and mark the functions you will disable. Click **OK** to return.

Note: Disabling functions applies to *all* sessions defined in the HTML file, not just the highlighted one.

Disabling functions is different from locking specific fields. The administrator can lock fields when configuring a session. Users cannot change the values for those fields. They appear grayed out on the client windows. When a function is disabled, it is removed from the toolbar or menus so users do not see it.

Details on the ability to disable functions is also found in the Administration Utility; see 7.1.8, “Disabling functions” on page 345.

To continue, click **Next** on the Host Sessions window and the Additional Options window shown in Figure 14-9 is displayed.

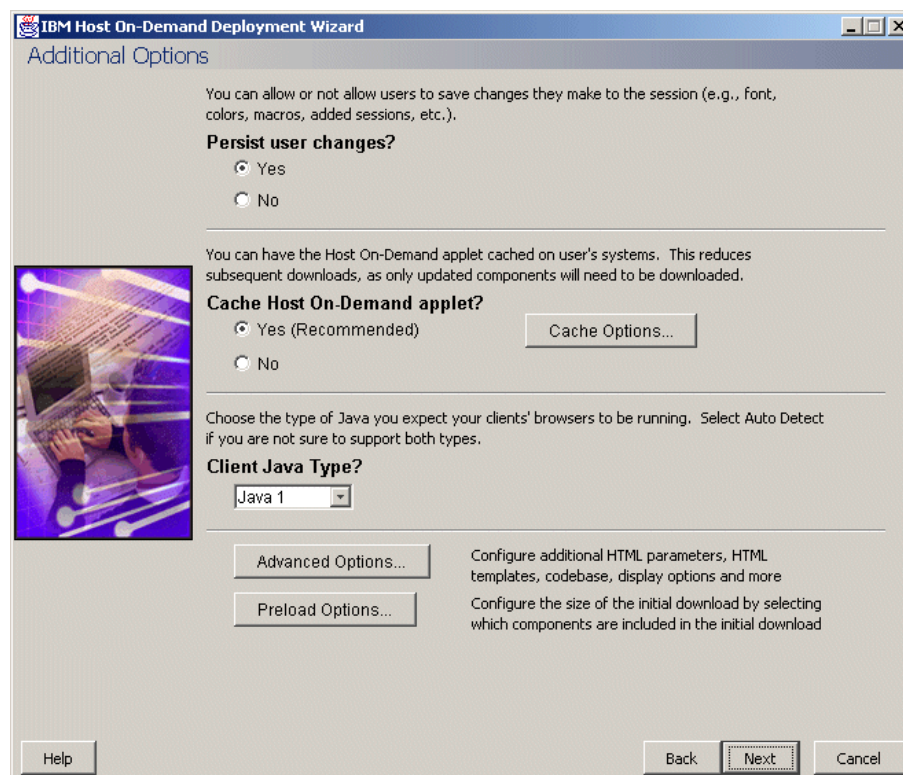


Figure 14-9 Additional options

This window has multiple functions. It allows the administrator to specify whether an end user can save user preferences, for example, keyboard re-mapping, made during an open session. If **No** is selected, end users are prevented from saving changes, ensuring that each time a session is opened, the original session properties are used. If **Yes** is selected, user changes will persist.

To reduce confusion, the administrator should inform the end users if saving changes is not allowed; otherwise, it may not be obvious to the end users why their changes are not being maintained from session to session. To control whether or not a user is allowed to make changes, use the Lock check boxes found on the Session Properties window (see Figure 14-7 on page 536). It is possible for the administrator to let the end users change the values of a session (by not locking the fields) but disallow the saving of these changes by selecting **No** to Persist user changes on the Additional Options window.

Browser considerations are detailed in “User preferences stored on local machines” on page 557. For this scenario, the administrator is allowing users to save changes.

The next section of the window lets the administrator determine if a cached or downloaded client will be created. If the cached client is selected, clicking **Cache Options** will further tailor the caching process. The administrator can select some load-balancing techniques that handle mass downloads when an upgrade is available and whether to allow the update to happen in the foreground or background. In Host On-Demand Version 7 the administrator can restrict the number of users that can upgrade in a certain time period. Note that Cache options are only valid if you are upgrading from Host On-Demand Version 5 or higher.

In this scenario, on the Cache Client Upgrade Option tab, shown in Figure 14-10 on page 540, the administrator specifies that 20% of the users can upgrade however only 10% of users can upgrade during 8am and 5pm. Specifying a percentage during a time period enables the administrator to control the amount of network traffic during their peak business hours.

If the administrator selects less than 100% in the Percent of users who can upgrade by default field, the administrator will need to re-edit the HTML file at a later stage to increase the percentage. This is required to ensure that all end users are upgraded. For example, in Figure 14-10 the administrator enters 20% as the Percent of users who can upgrade by default. The Host On-Demand applet randomly allocates a number which is used to determine if a user can upgrade. It is possible that even after accessing the HTML page several times a user may still not be upgraded. The initial percentage of users who can upgrade and the rate of increase will depend on your network infrastructure. For example the administrator initially selects 20%. In this scenario, users access the Host

On-Demand page daily and after a week the administrator is advised that the majority of users have been upgraded. The administrator re-edits the HTML file and modifies the upgrade percent to 100 to ensure all the remaining users will be upgraded.

The administrator can select whether the upgrade can take place in the foreground or background, where it will not disrupt the current open session. In this scenario the administrator will allow the end user to decide as shown in Figure 14-10. Note, selecting Upgrade in foreground will freeze the end users window until the download is complete hence this option is not recommended if your end users have low speed connections to the server such as dial-up connections.

Restriction: Java 2 cached clients cannot upgrade in the background.

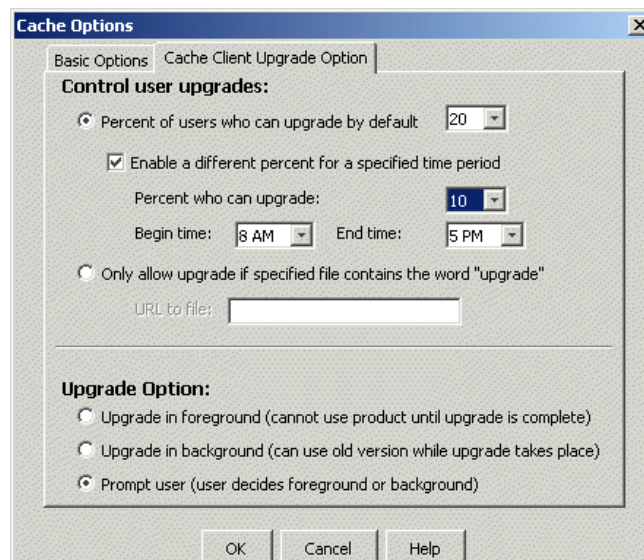


Figure 14-10 Cache options

After specifying the cache options click **OK** to return to the Additional Options window shown in Figure 14-9 on page 538.

From the Additional Options window the administrator can select Java 1 (the default), Java 2 or Auto Detect in the Client Java Type field. The value in this field refers to the Java level in the client's browser. When Auto Detect is selected end users will experience a delay while Host On-Demand attempts to detect if a Java 2 plug-in is available. A message will be displayed as shown in Figure 14-11.

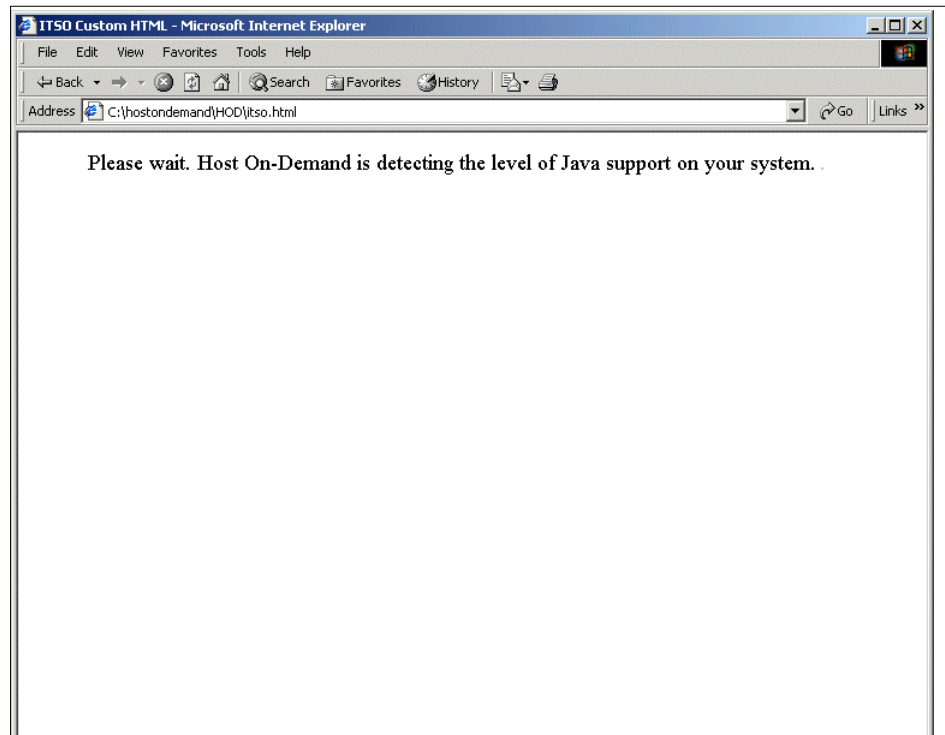


Figure 14-11 Java detect

If your end users only have Java-1 enabled browsers we recommend that you use the default setting which will bypass the detection process. If your end users are all using Java-2 enabled browsers select **Java 2**. If you are unsure of the level of Java in the end user's browser or have both Java 1 and Java-2 clients then select **Auto Detect**. For more information on Java 2 see Chapter 4, *IBM WebSphere Host On-Demand Version 7.0 Planning, Installing, and Configuring Host On-Demand*, SC31-6301-00.

Clicking the **Advanced Options** button shown on the Additional Options window brings up the window shown in Figure 14-12. The Display window allows the administrator to select the format of the sessions displayed to end users and the number of open sessions allowed per user. In this scenario, we show that the administrator restricts the number of open sessions per user to 5.

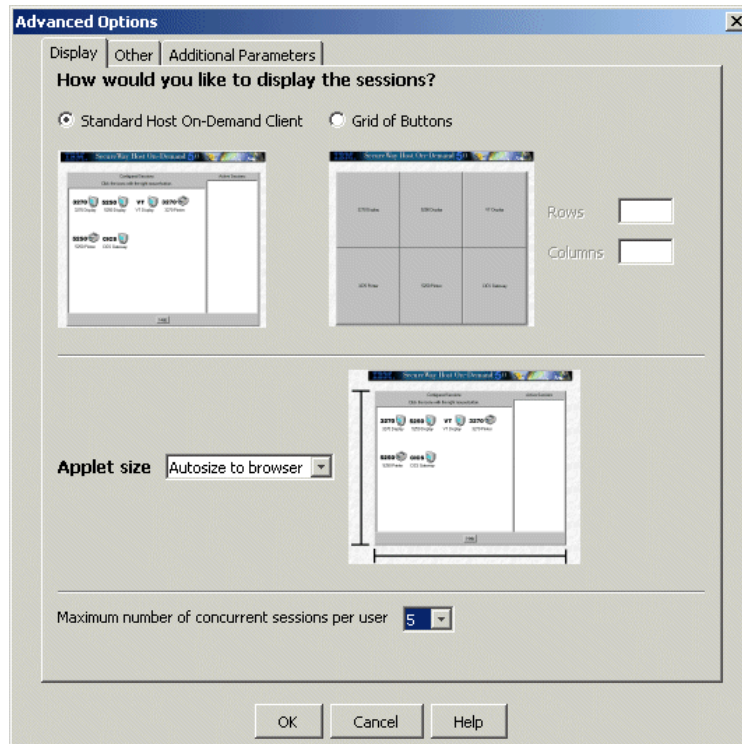


Figure 14-12 Advanced options

Clicking **Other** on the Advanced Options brings up the window as shown in Figure 14-13. In Host On-Demand Version 7 administrators can publish files generated from the Deployment Wizard to a location other than the Host On-Demand server publish directory. This function makes future upgrades easier and allows system administrators to restrict access to the publish directory. In this scenario the administrator selects to use this feature and enters the alias for the Host On-Demand publish directory in the Codebase field as shown in Figure 14-13.

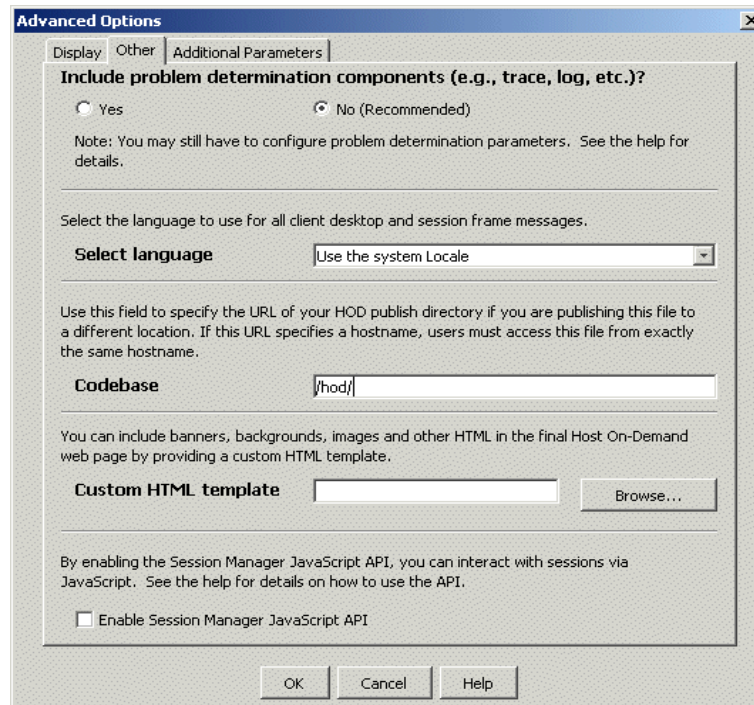


Figure 14-13 Other options

In this example a relative path has been entered. You can also enter a fully qualified URL in the Codebase field. Note, if you use a fully qualified URL, the web server name must match exactly with what the user enters in their web browser, even if the DNS entries resolve to the same ip address. For example, the administrator enters the Codebase field as follows:

```
http://itsoweb.raleigh.ibm.com/hod/
```

The customized HTML file created by the administrator is called itso.html. In this example we will assume that itsoweb.raleigh.ibm.com and webhod7.ibm.com resolve to the same ip address. If a user enters the following on their browser it will fail as the server DNS names do not match.

```
http://webhod7.ibm.com/hod/itso.html
```

In a like manner if you use an ip address in the fully qualified URL, the client cannot use a DNS name in their browser URL. We recommend using a relative URL if possible as it also provides the ability to load balance across multiple Host On-Demand servers.

Important: The Codebase parameter refers to the installed Host On-Demand server publish directory, not the directory where you will publish the Deployment Wizard files.

In Host On-Demand Version 7 the administrator can choose to select a customized HTML template they have previously created. For more information on Customizing HTML templates see Chapter 15, “Custom HTML templates” on page 567.

Administrators can set some additional parameters to be passed to Host On-Demand by clicking on the **Additional Parameters** tab. This window allows the administrator to set any HTML parameters that will apply to the HTML files being created. The values specified for the parameters in the HTML file take precedence over the values that may be specified in the config.properties file. In this scenario we have altered the directory where local preferences will be saved by setting the Save parameter as shown in shown in Figure 14-14. See the online help or “Additional HTML parameters” on page 564 for more HTML parameters allowed.

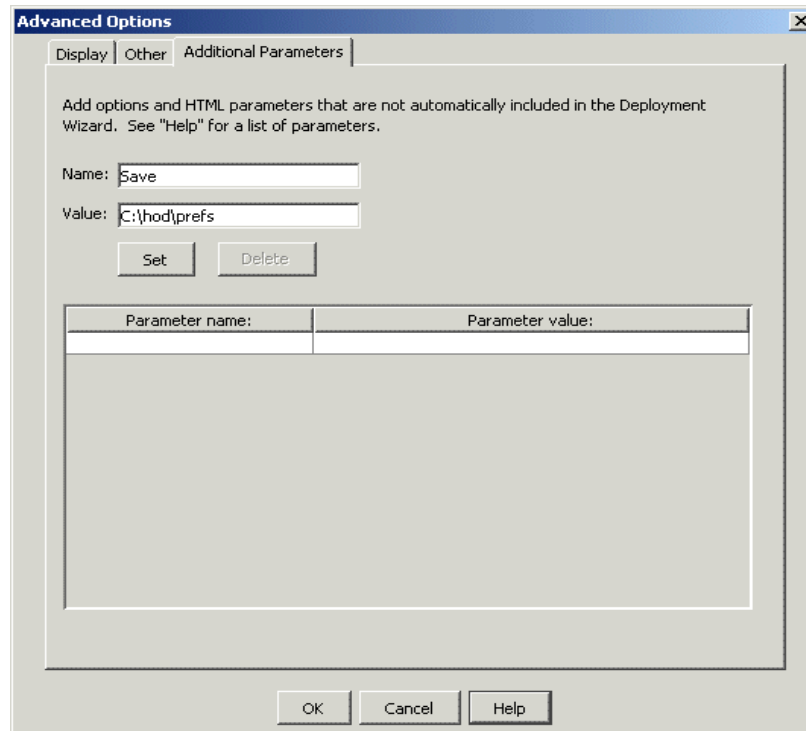


Figure 14-14 Additional parameters

Click **Set** and then **OK** to return to the Additional Options window shown in Figure 14-9 on page 538.

Clicking **Preload Options** from the Additional Options window allows the administrator to select components which will be downloaded when the HTML file is accessed. By default everything, except for specific language code pages, is downloaded. To reduce the size and time of the download, the administrator might choose to exclude certain components.

In this example, it is expected that the end users will only be using 3270 display/prINTER sessions. Therefore, to reduce the size of the download, the administrator clicks **Deselect All** on the 5250 display as shown in Figure 14-15. If these functions are ever needed by the end user, they will be downloaded upon subsequent access.

Restriction: If you are using the download client with a Java-2 enabled browser functional components will not be downloaded as needed. You will need to ensure the preload component list contains all the components required by the client.

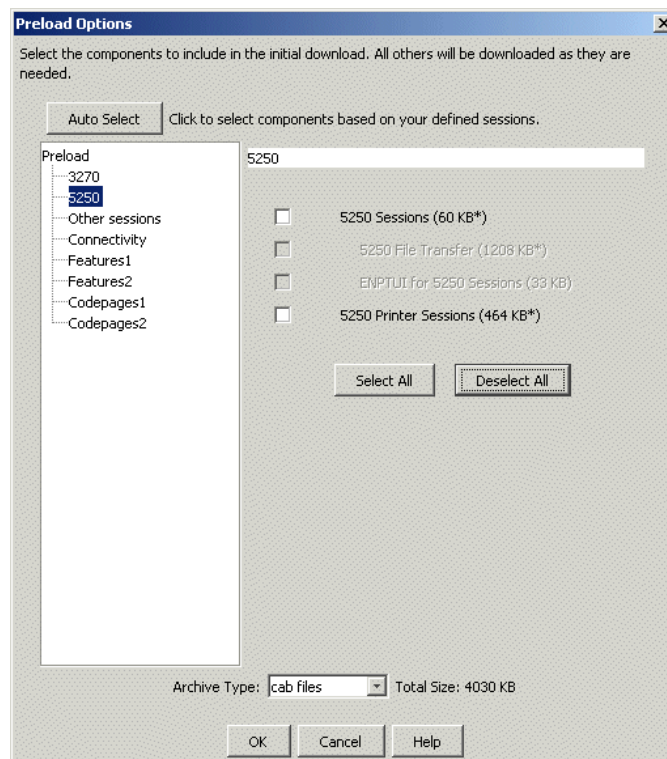


Figure 14-15 Preload options

Some sessions share components and selecting one but excluding another may not necessarily reduce the size of the download. For example, if the 3270 display session is excluded but the 3270 printer session is included, the size of the download is the same as if both were included, since the 3270 printer session needs some of the same files. In general, to reduce download size, select only those components that will be used. The preload options also apply when running the download client where the applet is downloaded every time the page is accessed, not just the first time.

Tip: When creating customized pages using the HTML model you can use the **Auto Select** button. The Deployment Wizard will automatically select the components required based on the sessions configured and options checked.

After selecting the required download components, click **OK** to return to the Additional Options window shown in Figure 14-9.

After completing the Additional Options window and clicking **Next**, the File Name and Output Format window is displayed as shown in Figure 14-16. This window displays a summary of the options selected thus far. The administrator supplies a page title, file name and selects the directory where the Deployment Wizard output will be stored. If the Deployment Wizard is running on the server, the directory will be the Host On-Demand publish directory. The administrator can change the directory by clicking the on the **Browse** button to browse the file system. With Host On-Demand Version 7 administrators can also select the type of output to be created by the Deployment Wizard. Output HTML is the default as shown in Figure 14-16. In this example the Deployment Wizard files will be created in C:\hostondemand\HOD.

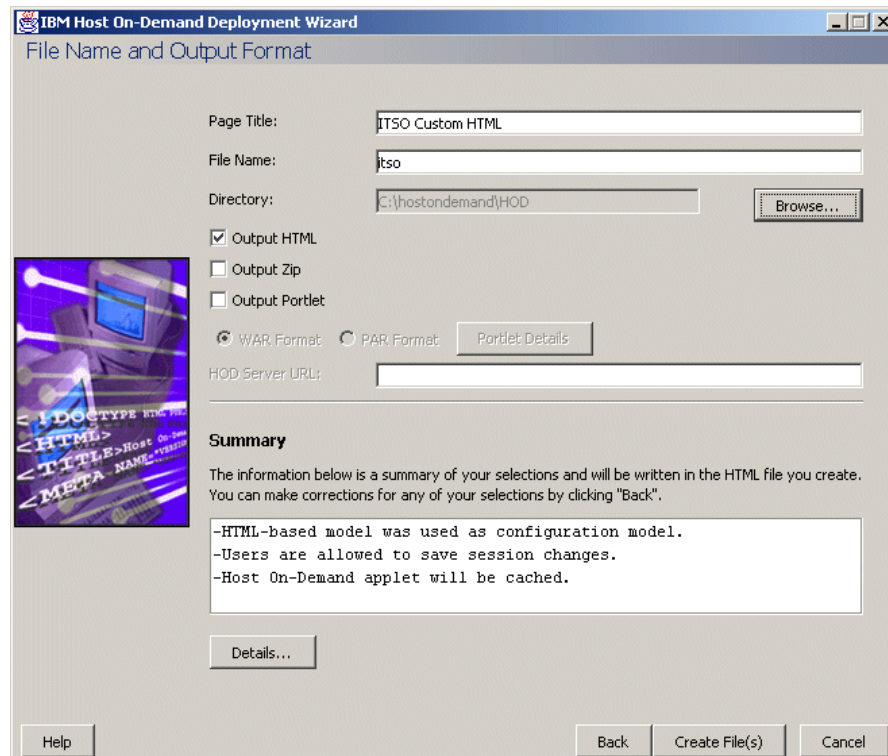


Figure 14-16 Wizard output

In Host On-Demand Version 7 administrators can choose to create a zip file by selecting Output Zip. This option creates a zip file of the Deployment Wizard-generated files. The DWunzip tool is used to install the files to the Host On-Demand server, see “Distributing Deployment Wizard files” on page 560 for instructions. This format is recommended if you are running the Deployment Wizard on a different server or platform from your Host On-Demand server.

Host On-Demand Version 7 can run as a portlet on Portal Server, a component of IBM WebSphere Portal. If Output Portlet is selected the administrator will need to configure additional parameters. See Chapter 17, “Host On-Demand Portlets” on page 617 for further information on using Host On-Demand as a portlet.

Finally, the administrator selects **Create File(s)** and the Deployment Wizard output is created and stored in the specified directory. See “Files created by the Deployment Wizard” on page 562 for a description of the files that are created.

If the Deployment Wizard is installed on a server different from the Host On-Demand server the files must be transferred to the system where Host On-Demand is installed. See “Distributing Deployment Wizard files” on page 560 for instructions.

After the files have been distributed to the Host On-Demand server end users will indicate the address of the machine where the Web server and Host On-Demand publish directory is installed, specifying the new customized HTML. For example:

```
http://TheHODWebserver/hod/itso.html
```

14.3.2 Configuration Server-based model example

In the configuration server-based model session information is maintained on the configuration server using the Host On-Demand Administration utility. Configuration information is defined in group or user IDs created by the administrator. If the administrator allows users to save preferences, changes are stored in the user ID on the server. This model can be useful in the scenario where clients access host sessions from a number of terminals and they are unable to retrieve their local user preferences. With large number of users however there can be a high administrative requirement to create and maintain all the user IDs.

After starting the Deployment Wizard, and selecting **Create a new HTML file**, the administrator selects the Configuration Server-based model (see Figure 14-17).

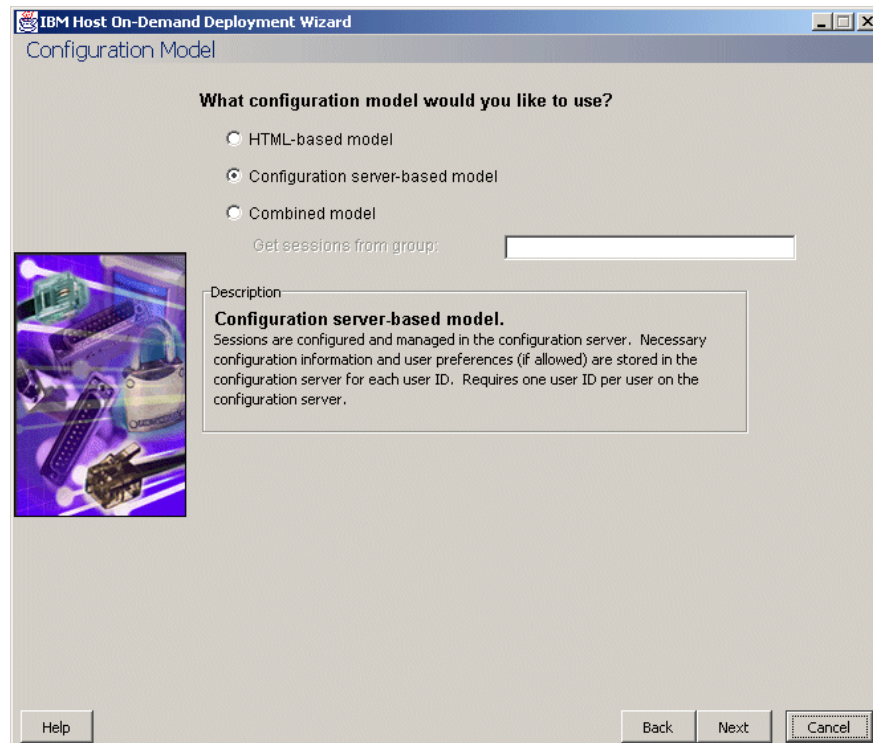


Figure 14-17 Configuration server-based model

The next window, shown in Figure 14-18, determines if the end users will log on using their Host On-Demand user IDs or their Windows user IDs. Since some of the users in this scenario may be on non-Windows platforms, the administrator chooses to use the Host On-Demand user IDs. This requires the administrator to set up user IDs using the Administration Utility prior to end users accessing the customized HTML page.

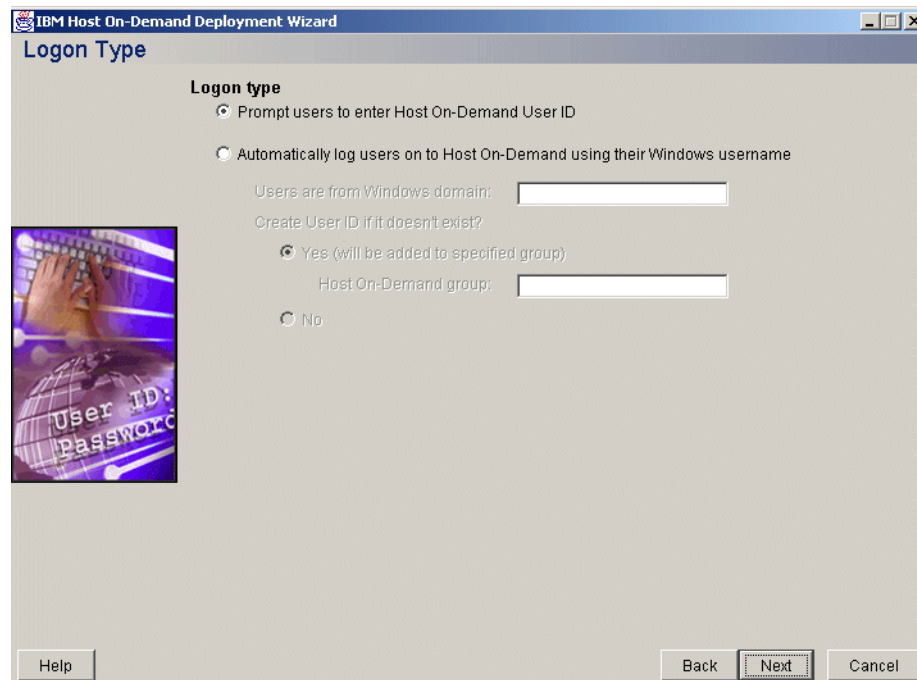


Figure 14-18 Logon type

If some of the end users run on the Windows platform and have similar connectivity requirements, the administrator may choose to group these users together and create a customized HTML file for them whereby they are automatically logged on to Host On-Demand using their Windows user ID. Using the Integrated Windows domain logon, Host On-Demand offers several advantages:

- ▶ Host On-Demand user IDs can be created automatically.
- ▶ Users can bypass the Host On-Demand logon window.

See 11.12, "Integrated Windows domain logon" on page 475 for more information.

After completing this window and clicking **Next**, the window shown in Figure 14-19 is displayed.

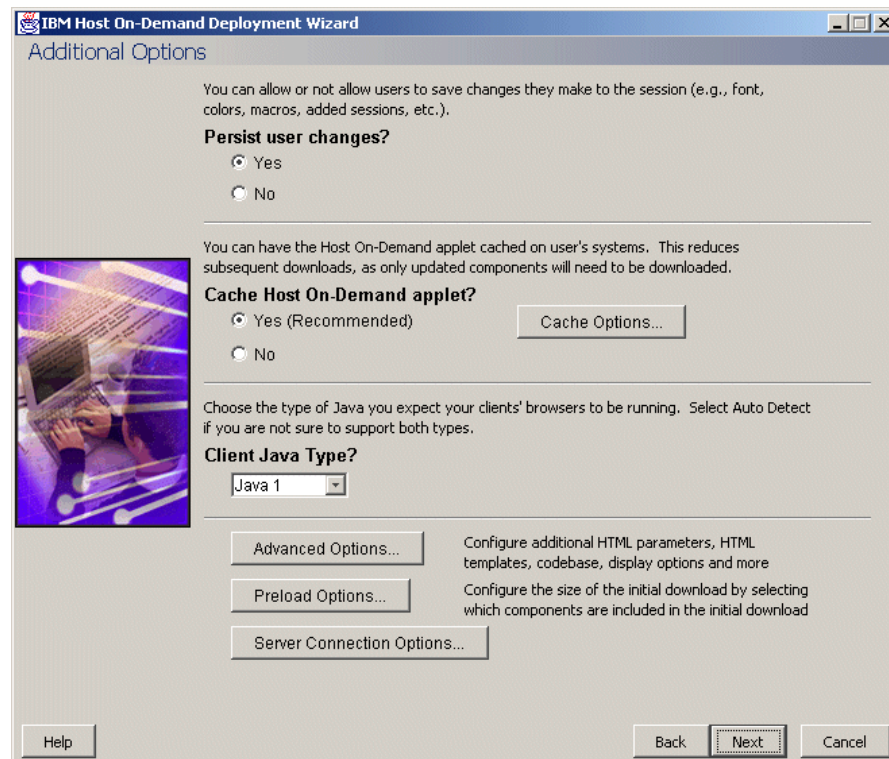


Figure 14-19 Server connection options

This window is similar to the Additional Options window shown in Figure 14-9 on page 538; however, because this session will use the Configuration Server-based model, a new button, **Server Connection Options**, is displayed. Clicking **Server Connection Options** displays the window shown in Figure 14-20.

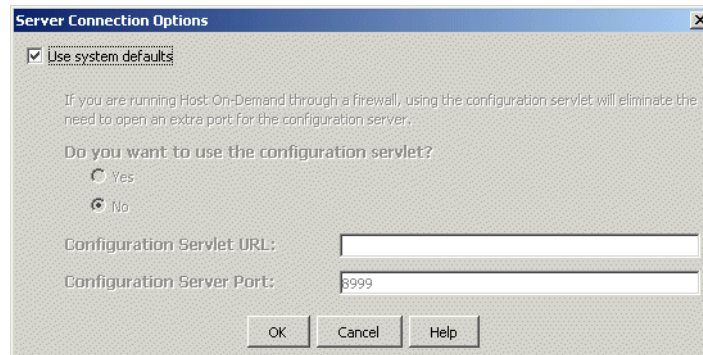


Figure 14-20 Server connections - system defaults

If **Use system defaults** is checked, the parameters in the config.properties file are used by the Host On-Demand applet.

During Host On-Demand installation, the config.properties file is created in the Host On-Demand publish directory except on z/OS servers. On z/OS servers, this file is named config.properties.ascii and must be created manually. See “config.properties.ascii” on page 101 for more information. The config.properties file is read by all Host On-Demand applets and provides a global way of setting HTML parameters. If the default connection port of 8999 was chosen during installation, this file would contain the parameter/value of:

```
ConfigServerPort=8999
```

If the Configuration Servlet was installed and a connection port was specified, the config.properties file would contain the following:

```
ConfigServletURL=hostname_or_IPaddress\HODConfig\hod
```

If you have changed the default settings you can enter your new values via the Deployment Wizard as shown in Figure 14-21. However, if you wish to use these setting for all your client's host sessions we recommend that you update the config.properties file on your Host On-Demand server.

To override the default parameters uncheck the **Use system defaults** button as shown in Figure 14-21.

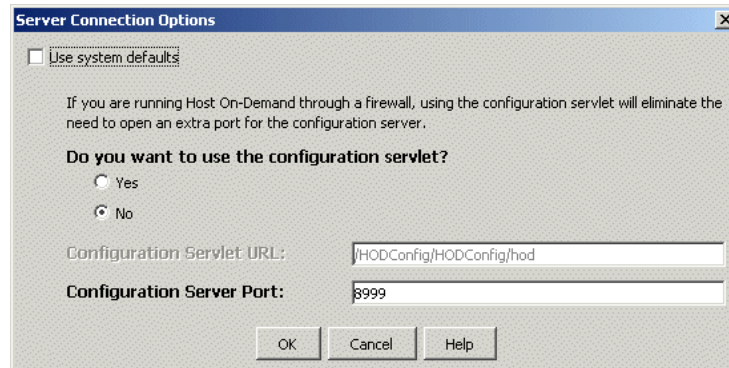


Figure 14-21 Server connection options - no system defaults

To override the Configuration Servlet values, select **Yes** and supply the URL to where the Configuration Servlet is installed. To override the Configuration Server port, select **No** to Do you wish to use the configuration servlet and specify the new port.

Note that specifying a new Configuration Server Port value on this window does not change the port that the Configuration Server is “listening” on. It simply provides a ConfigServerPort parameter in this HTML file that overrides the ConfigServerPort specified in config.properties file. Only end users of this HTML file will pick up this parameter. To change the Host On-Demand listening port the administrator must modify the NSMprop file. See “Changing the Service Manager’s configuration port” in the online help for details.

Returning to our scenario, we will assume that using the system defaults is sufficient. After completing the Additional Options window and clicking **Next**, the File Output and Format window is displayed as shown in Figure 14-16 on page 548. Note that the Host Sessions window shown in Figure 14-6 on page 535 will not be shown since, with the Configuration Server-based model, the administrator uses the Administration Utility to define host sessions and their properties. The information is kept on the Configuration Server and not in the customized HTML files created by the Deployment Wizard.

After the files have been distributed to the Host On-Demand server, end users will access the HTML page in the same method as used by the HTML model. For example:

```
http://TheHODWebserver/hod/configmodel.html
```

where configmodel is the name specified in the File Name field (see Figure 14-16 on page 548).

14.3.3 Combined model example

In the combined model host session information is defined on the configuration server (like the configuration server-based model) and if allowed by the administrator, user preferences are stored on the user's machine (like the HTML-based mode). Unlike the configuration server-based model, Host On-Demand user IDs are not created on the configuration server. Configuration information is stored in group definitions on the server created via the administration utility. If the default session information within the group is ever changed by the administrator, the users of the HTML file will automatically receive the updates.

To illustrate the combined model in our scenario we have chosen to use a Host On-Demand server installed on a z/OS platform. Since the Host On-Demand Configuration Server was installed on a z/OS system, the Deployment Wizard must be run on a Windows machine. In this example the administrator will allow end users to save their user preferences.

After choosing to create a new HTML file, the window in Figure 14-22 is shown.

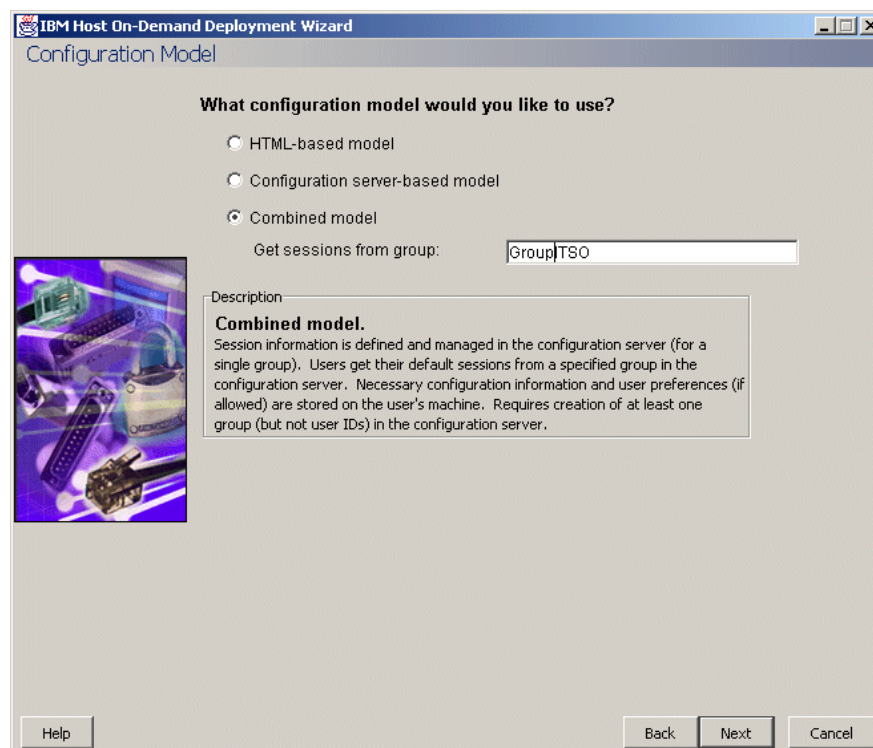


Figure 14-22 Combined model

The administrator selects the combined model and must also supply a group that is defined on the Configuration Server, GroupITSO. GroupITSO must be defined on the Host On-Demand server prior to end users accessing the customized HTML page because configuration information is obtained from this group. The administrator proceeds by clicking **Next** and the Additional Options window (shown in Figure 14-9 on page 538) is displayed.

Additional parameters can be selected in the same manner as the HTML and Configuration server-based model. Click **Next** to display the File Name and Output Format window. In this example the administrator selects **Output zip** from the Output format window as shown in Figure 14-23. The file GroupITSO.zip will be stored in the directory entered by the administrator: C:\DWizard. See “Distributing Deployment Wizard files” on page 560 for information on deploying the custom HTML files.

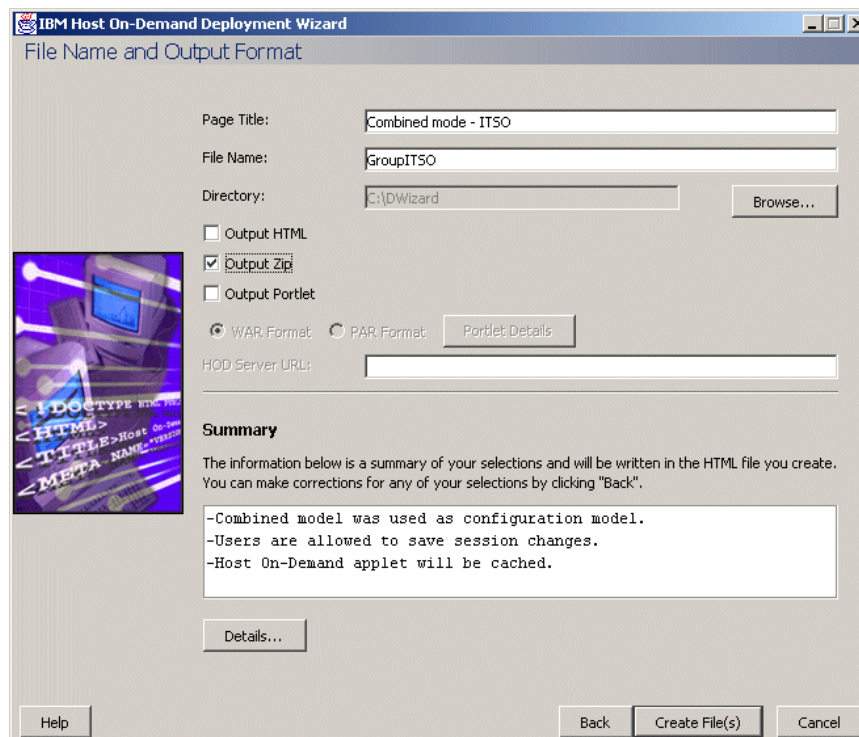


Figure 14-23 Output zip

Once the files are on the z/OS system and the group containing the configuration information, GroupITSO, has been created by the administrator, end users can access these HTML files. For example:

<http://TheHODWebserver/hod/GroupITSO.html>

If the user makes any changes to the configuration information, these changes are stored on the user's local machine. As long as the applet is running, the Configuration Server on the z/OS host system should not be accessed again to obtain configuration information.

If the administrator makes a change to the configuration information in GroupITSO, the next time the user downloads the client, the new change will be picked up and merged with the user's preferences. For example:

1. The administrator initially sets up a 3270 session definition with a destination address of `sys1.raleigh.com` and a session name of `Raleigh System`.
2. The user downloads the applet and changes the session name to `My test system` (if this field is not locked).
3. Later, the system is moved and the destination address changes. The administrator logs on to the Administration Utility and changes the definition in GroupITSO to have a destination address of `sys2.raleigh.com`.
4. The next time the user downloads the applet, the new destination address will be used and saved but the session name will remain since the user had previously specified, `My test system`. (This is true as long as the administrator did not also lock the session name field when the destination address was changed.)

If the z/OS Web server is running, but the Configuration Server becomes unavailable, and the HTML files have been used at least once, the configuration settings on the user's local machine will be used to connect to the host system. A message is displayed to the end user and the Host On-Demand applet continues as usual.

If the HTML files have not been used, there is no locally stored user preferences or session configuration information that can be used to launch a session; therefore, a message is displayed to the end user and the Host On-Demand applet stops running.

14.3.4 User preferences stored on local machines

In the HTML-based model and combined model the default is to allow user changes to persist. User preferences are stored in files on the user's local machine. If the administrator has not allowed user changes to persist there are no local files stored. There are some special considerations to be aware of when these models are used.

Browser considerations

Depending on which browser is used to access Host On-Demand, the files are stored in different locations. (See 14.5.3, “Files stored on local machine” on page 563 for more information). This can cause confusion if an end user decides to use a different browser after making changes. The changes will not be picked up by the new browser. If as the administrator you are unable to restrict users to using a single browser it is recommended to use the Save HTML parameter when creating the HTML files with the Deployment Wizard. (See Figure 14-14 on page 545). This parameter allows the administrator to specify where the files are to be stored locally regardless of what browser the end user chooses.

Deleting sessions

An end user who has their user preferences stored on a local machine will not be allowed to delete the only copy of a session from the Host On-Demand Configured Sessions window. A message is displayed with this information. This is illustrated in Figure 14-24 where we tried to delete the session 3270 Display.

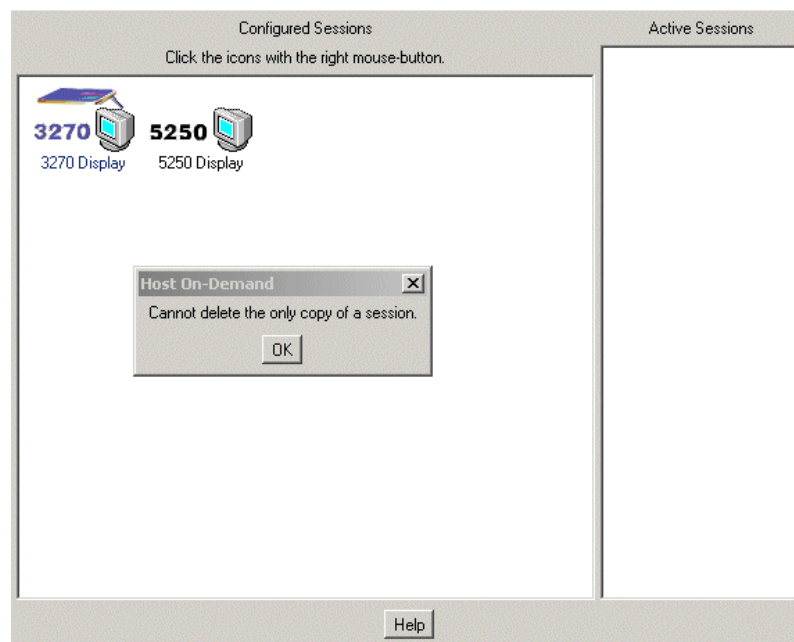


Figure 14-24 Deleting session error

Order of precedence for changes

When configuration information is kept in different places, there is an order of precedence as to which value is used when a change occurs in one place but not the other.

Initially, all values are created by the administrator. If the HTML-based model is used, the values are stored in the `cfgn.cf` file created by the Deployment Wizard. If the combined model is used, the values are stored in the Configuration Server. The initial download of the applet will use these values.

A user can change many of these values as long as the administrator has not locked the field. The user preferences are stored locally if the administrator has allowed changes to persist.

On subsequent downloads of the applet, configuration information is determined using the following hierarchy:

1. If a field is locked by the administrator, the value from the administrator is used.
2. If the user has made and kept changes locally, the values from the local user preferences are used.
3. If the administrator makes a change to a field that the user has not yet modified or adds new configuration values, the values from the administrator are used.

Note, once a user has modified a field this value will override a subsequent change by the administrator. If as the administrator you wish to maintain control over certain fields it is recommended you use the lock function which will disable users from being able to make changes to the field.

Restoring the default preferences

To remove user changes and restore the default session properties, users can delete the files from their local machine that contains the user-specified changes. See 14.5.3, “Files stored on local machine” on page 563 to determine what files to delete. For example, to restore the defaults of a session called “3270 Payroll” that was set up in a customized HTML file called `ITSO.html`, look for a folder called `ITSO` on the local machine. If you delete this folder, all sessions defined in `ITSO.html` will be restored to their default settings. To restore a single session, open the folder `ITSO` and delete the `cfgn.df` file that contains the text string “name=3270 Payro11”.

14.4 Distributing Deployment Wizard files

Once the custom HTML files have been created by the Deployment Wizard, they must be distributed to the production server. If Output Zip is checked the procedures is as follows:

1. Transfer the zip file created by the Deployment Wizard, in binary, to the server publish directory. This will either be the Host On-Demand publish directory or your user publish directory.
2. Edit the DWunzip file located on your server if necessary. You will need to edit the file if you have changed the default publish directory or if you have transferred the zip file to your user publish directory.
3. If you are running DWunzip on a UNIX-based or z/OS platform you will need to ensure DWunzip has execute permission.
4. Run DWunzip by entering DWunzip xxxxx where xxxxx is the name of the zip file created by the Deployment Wizard and transferred to the server.

Note: Output Zip is recommended if your Host On-Demand server is on a different server or platform than your Deployment Wizard. The DWunzip tool ensures that files are stored in the appropriate directory. If your server is on a non- Windows platform, the DWunzip tool sets file permissions and ownership. If running DWunzip on a z/OS platform the necessary file extensions are also added.

In the Combined Model example in theFigure 14-23 on page 556 we entered GroupITSO in the File Name file and checked Output Zip. The output directory was modified to C:\DWizard. After clicking Create File(s) the window in Figure 14-25 is displayed. This window displays the name of the zip file created and allows the administrator to view online help for using the DWunzip tool.

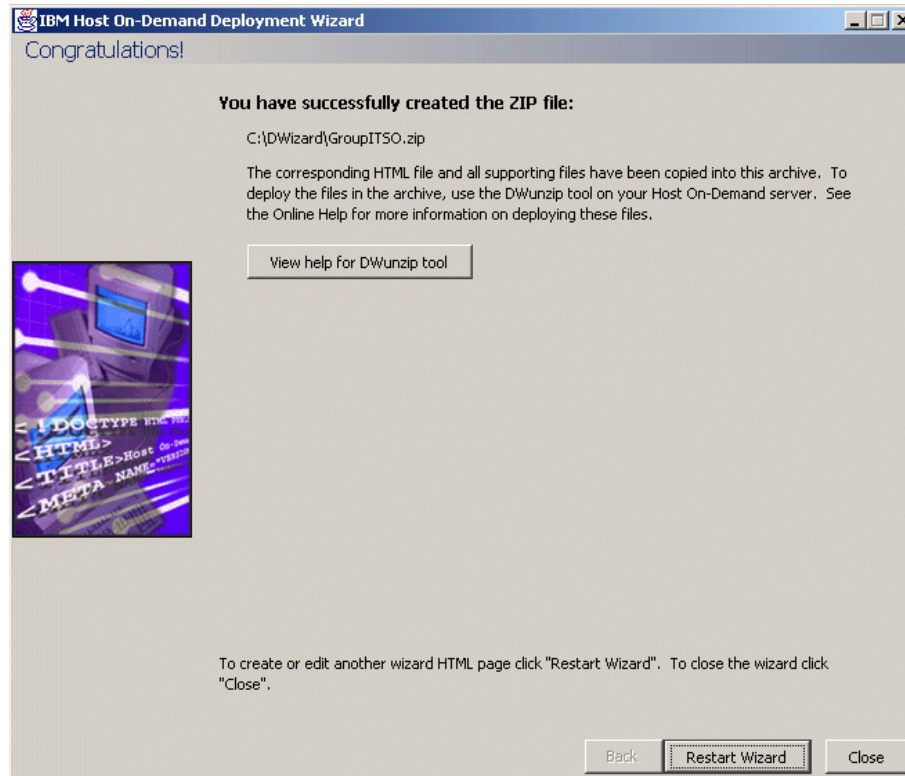


Figure 14-25 Dwunzip

If Output HTML was checked the basic procedure is as follows:

- ▶ Refer to “Files created by the Deployment Wizard” on page 562 to help you determine the files that must be transferred and the file format, either text or binary.
- ▶ FTP the files to your configuration server platform.
- ▶ If deploying the files to a UNIX platform you will need to ensure file permissions are set correctly.
- ▶ If the server is on a z/OS system, see Chapter 3.4, “Deployment Wizard considerations” on page 96 for special considerations.

14.5 Files created by the Deployment Wizard

There are several files that may be created by the Deployment Wizard. Not all files are created for every type of client, but the directory and file structure is identical for download and cached clients.

14.5.1 Files stored in publish directory

Table 14-1 lists the files that are created and must be stored in the server's Host On-Demand publish directory or your user publish directory. The default publish directory is:

\hostondemand\HOD

Table 14-1 Files stored in publish directory

File Name	Type	Description
xxxxx.html	text	HTML file created by Deployment Wizard; used by the end user to launch the client.
z_xxxx.html	text	HTML file created by the Deployment Wizard if Java 2 or Auto Detect is selected.
where xxxxx is the File Name specified on the Output Format window of the Deployment Wizard. If Java 1 is selected z_xxxx.html will not be created.		

14.5.2 Files stored in customized subdirectory

For every customized HTML that is created, a corresponding subdirectory is created with the same name under \HODData. The default is:

\hostondemand\HOD\HODData\xxxxx

These files contain the session configuration information created by the administrator.

Table 14-2 lists the files created by the Deployment Wizard and stored in \HODData\xxxxx.

Table 14-2 Files stored in \HODData\xxxxx

File Name	Type	Description
cfgn.cf	text	One for each session defined; contains configuration information set by administrator.
params.txt	text	Contains some configuration parameters for the setup of the client session window.

File Name	Type	Description
policy.obj	binary	Contains information about the Disabled Functions; see Figure 14-8 on page 537.
preloads.obj	binary	Contains information about the objects to preload as defined on the Preload Options window; see Figure 14-15 on page 546.
udparams.txt	text	User-defined HTML parameters.
wlinfo.txt	text	Contains the responses to each window in the Deployment Wizard; used only by the Deployment Wizard.
cfgn.cf and policy.obj will only be created if HTML-based model is selected		

14.5.3 Files stored on local machine

If the HTML-based model or combined model was used in the Deployment Wizard and Persist user changes was checked, user preferences are kept on the local machine of the user.

Depending on which browser was being used to access Host On-Demand, the files that contain the user preferences can be in different directories. Table 14-3 lists the directories where local files will be stored.

Table 14-3 Default local directories

Platform and Browser	Directory
Windows 98 & IE Windows 98 & Netscape 4 Windows 98 & Netscape 6	\Windows\Java\username\HODData\xxxxx \Netscape\Users\username\HODData\xxxxx \Windows\username\HODData\xxxxx
Windows 2000/XP & IE Windows 2000/XP & Netscape 4 Windows 2000/XP & Netscape 6	\Documents&Settings\username\HODData\xxxxx \Netscape\Users\username\HODData\xxxxx \Documents&Settings\username\HODData\xxxxx
Windows NT & IE Windows NT & Netscape 4 Windows NT & Netscape 6	\WINNT\Profiles\username\HODData\xxxxx \Netscape\Users\username\HODData\xxxxx \WINNT\Profiles\username\HODData\xxxxx
OS/2 & Netscape 4 OS/2 & Netscape 6	\Netscape\Users\username\HODData\xxxxx \java13\JRE install dir\HODData\xxxxx
UNIX & Netscape 4 UNIX & Netscape 6	/home/username/.netscape/HODData/xxxxx /home/username/HODData/xxxxx

where xxxxx is the name of the custom HTML file created and username is the user ID of the logged-on user. With Netscape 4.x on Windows it represents the Netscape profile ID.

If the Save HTML parameter was specified when the HTML files were created, the files will be located in this directory, regardless of which browser is used:

<save base directory>\userID\HODData\xxxxx

Table 14-4 lists the files stored in this directory.

Table 14-4 Files stored on local machine

File Name	Type	Description
cfgn.cf	text	One for each session defined; contains configuration info set by administrator
cfgn.df	text	Difference file; contains config changes made by user; one for each session changed by user
metadata	text	Info needed for Host On-Demand processing
version	text	Info needed for Host On-Demand processing

14.6 Additional HTML parameters

The following are some additional HTML parameters that the administrator can specify in the window shown in Table 14-5 on page 565. Do not use a text editor to manually add these parameters to the HTML file or they will be lost the next time the file is opened with the Deployment Wizard. Additional details on each parameter can be found in the online help for this window.

Table 14-5 Additional HTML parameters

Parameter	Value	description
3270InputAreaIndication	Underdot,DisplayAndNonDisplay UnderDot,NonDisplay UnderDot,Display UnderLine,Display UnderLine,DisplayAndNonDisplay UnderLine,NonDisplay 3DLowered,DisplayAndNonDisplay 3DLowered,Display 3DLowered,NonDisplay 3DRaised,DisplayAndNonDisplay 3DRaised,Display 3DRaised,NonDisplay	Indicates that the unprotected fields in a 3270 session be indicated in a particular method. Underdot - Causes a dot to be placed under every character position in the fields indicated by the second value. Underline - Causes an underline to be placed under every character position in the fields indicated by the second value. 3D - Causes a 3D rectangle to be displayed in the fields indicated by the second value. Display - Only unprotected displayable fields will have the selected indication applied. NonDisplay - Only unprotected nondisplayable fields will have the selected indication applied. DisplayAndNonDisplay - All unprotected fields will have the selected indication applied.
CustomKeyFunctionX	function identifier function data	Customizes the list of functions that a key or key combination can be mapped to using keyboard remapping.
Disable	lum	Disables license use counting and License Use Management server reporting.
DoNotPrefillUser	true	Causes the logon window to come up with a blank userID field. If this parameter is absent, the default is to fill in the user ID with the user name requested from the system.
HideHODDesktop	true	Hides the Host On-Demand desktop and sessions tabs once an embedded sessions starts. Recommended only if one host session is configured which will be auto-started.

Parameter	Value	description
Save	base save directory	Specifies the location where the user preferences are to be stored. Only applies to Combined and HTML-based models.
ShareCachedClient	true	Enables Windows 2000 and XP multi-user machines (using IE and Microsoft JVM) to share an image of the cached client. See the online documentation for details regarding Restricted Users.
DebugCode	65535	Enables debugging information to be written to the Java console. Java console must be enabled.



Custom HTML templates

In this chapter we cover the support added in Host On-Demand to preserve customizations you make to a Host On-Demand Web page generated by the Deployment Wizard. This feature allows you to place the customizations in a separate file so that they are not lost if the Deployment Wizard is again used later on to modify the page.

15.1 Purpose of this feature

A custom HTML template is a file in which you can preserve HTML elements that modify the main HTML output file of the Deployment Wizard.

As you know, Deployment Wizard creates several output files (refer to 14.5, “Files created by the Deployment Wizard” on page 562). The main HTML output file is the file which contains the HTML and JavaScript elements that display the visible features of the Host On-Demand desktop page, shown in Figure 15-1.



Figure 15-1 Default Host On-Demand desktop

Figure 15-1 shows the default Host On-Demand Desktop page with its light-gray textured background, its IBM WebSphere Host On-Demand banner at the top, and its centrally located Host On-Demand desktop window containing the icons of configured sessions.

You might want to modify the main HTML output file of the Deployment Wizard in order to change the appearance of the Host On-Demand desktop page (for example, by replacing the Host On-Demand banner with a banner displaying your company's name), or to add HTML elements such as hyperlinks and forms to the page, or to add programming elements such as JavaScript or JavaServer Page elements.

Before Host On-Demand V7.0, you could make such changes by following these steps:

1. Running the Deployment Wizard to create output files
2. Manually editing the main HTML output file of the Deployment Wizard to add HTML elements, JavaScript elements, or JavaServer Pages elements

But there was a problem with this method. If later on for any reason you were to run the Deployment Wizard again to modify the main HTML output file (for example, in order to add an additional 3270 session), then the customized changes that you had added to the main HTML output file would be lost.

Specifically, the Deployment Wizard would create a new main HTML output file that would not contain your customizations. You would have to manually edit the new main HTML output file to add your customizations, just as you did the first time.

The custom HTML template feature alleviates this problem by allowing you to place your HTML, JavaScript, and JavaServer Pages customizations in a separate file where they can be preserved. In order to work properly, the custom HTML template file must be based on a skeleton of text markers and basic HTML elements that are defined in the file Wizard.html in the Host On-Demand publish directory.

Wizard.html also serves another purpose: if the user does not specify a custom HTML template file, Deployment Wizard uses the contents of Wizard.html as the skeleton for the main HTML output file.

15.1.1 Note: the Host On-Demand logon page is also affected

The customizations in a custom HTML template file will affect not only the Host On-Demand desktop page but also the accompanying Host On-Demand logon page, if your configuration model and options cause this page to be displayed.

15.1.2 Supported configuration models

You can use a custom HTML template file with any of the three Deployment Wizard configuration models:

- ▶ HTML-based model
- ▶ Configuration-server-based model
- ▶ Combined model

15.1.3 Client Java Types

You can use a custom HTML template file with any of the three Deployment Wizard Client Java Types:

- ▶ Java 1
- ▶ Java 2
- ▶ Auto Detect

15.2 Name of the main HTML output file

The name of the main HTML output file created by Deployment Wizard is an implementation-specific detail that may change in later releases. However, in order to enable you to verify or debug elements that you have included in a custom HTML template, here is some information about the name of the main HTML output file in Host On-Demand Version 7.0.

The information is simple.

Table 15-1 Main HTML output file name

On the last page of the Deployment Wizard, in the File Name field, if you specify MyCompany	The name of the main HTML output file is:
If the Client Java Type is Java 1	MyCompany.html
If the Client Java Type is Java 2 or Auto Detect	z_MyCompany.html

15.3 Specifying a custom HTML template file

This section deals with specifying a custom HTML template file in the Deployment Wizard. For the basic information on this topic see the Host On-Demand online documentation.

15.3.1 File management

To specify a custom HTML template in the Deployment Wizard, go to the Additional Options page, click **Advanced Options...**, then click the **Other** tab. Enter the file name in the Custom HTML template field. (If you need a more detailed description of how to perform these steps, consult the section on the Deployment Wizard in the Host On-Demand online help.) Figure 15-2 shows the Advanced Options window with the Custom HTML template field.

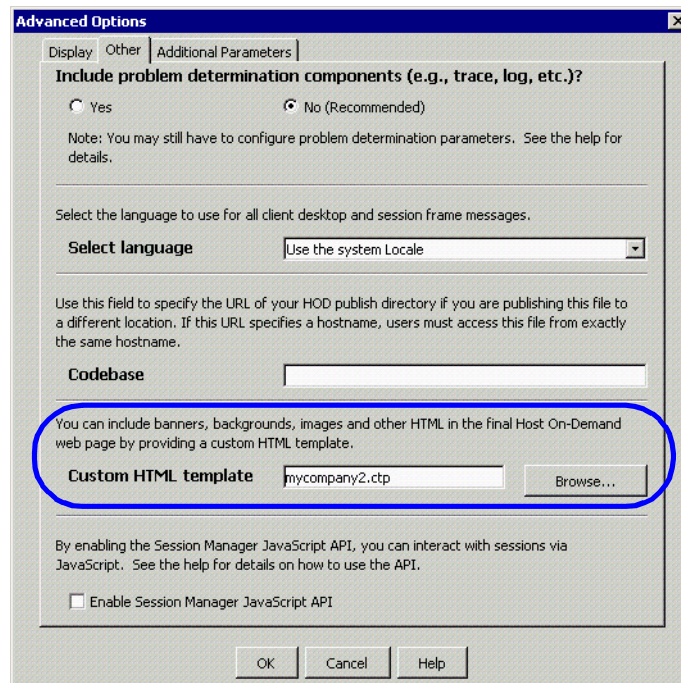


Figure 15-2 Advanced Options window, Other tab, Custom HTML template field

You can enter a file name, a relative path, or an absolute path. If you enter a file name or a relative path then Deployment Wizard starts looking for the file in the Host On-Demand publish directory.

It is probably better practice to enter an absolute path and maintain your custom template files in a separate directory outside of the Host On-Demand directory structure. This helps you keep custom HTML template files safe and separate from Deployment Wizard HTML output files.

You might want to identify all custom HTML template files by giving them a unique file extension or sequence of characters in the file name, for example, MyCompany.ctp or tpIMyCompany.html.

15.3.2 Do not modify Wizard.html

You can modify Wizard.html directly instead of specifying a custom HTML template, but this is not a good idea. Instead, copy the contents of Wizard.html to a target file such as mytemplate.html, and modify the target file.

In fact, you should make a backup copy of Wizard.html in case you happen to yield to the temptation of taking a shortcut by modifying Wizard.html directly, or in case Wizard.html is corrupted accidentally.

Tip: Keep Wizard.html in pristine condition so that you can use it as a template for your custom HTML template files, and so that it can continue to be used as the default template when no custom template is specified.

If you do modify Wizard.html directly, do not specify a custom HTML template in Deployment Wizard when you create the output files.

15.3.3 Custom HTML template vs. main Deployment Wizard output file

The main Deployment Wizard output file is the file whose name you specify in the File Name field on the File Name and Output Format page of the Deployment Wizard.

You also specify a main Deployment Wizard output file in the File Name: field on the Edit Existing HTML file page when you are starting out in the Deployment Wizard. On this page do not get confused and enter the name of a custom HTML template file. If you have associated a custom HTML template file with this main output file previously, Deployment Wizard will look in the main output file and find the name of the custom HTML template file.

15.3.4 When error checking is performed

Deployment Wizard does not search for the custom HTML template file or inspect its contents until you click the **Create File(s)** button on the File Name and Output Format page. Therefore Deployment Wizard will not inform you of any errors in the custom HTML template file until this point.

Here are some of the errors you might encounter:

- ▶ Deployment Wizard cannot find the custom HTML template, based on the path you specified.
- ▶ The custom HTML template file is not in valid UTF-8 format.
- ▶ The text markers in the custom HTML template file have been corrupted

15.4 UTF-8 encoding

Your custom HTML template file should be in UTF-8 format, a standard format for storing Unicode characters. Wizard.html is in UTF-8 format. UTF-8 uses a unique sequence of 1, 2, 3, or 4 bytes to encode each character in the Unicode character set. Therefore UTF-8 can handle all the Unicode characters, including all the characters of double-byte-character-set languages.

The Notepad editor included with Microsoft Windows 2000 and with Microsoft Windows XP is UTF-8 capable (but the Notepad editor included with Windows NT is not). When Notepad's Save As message box comes up, click the **Encoding:** combo box and select **UTF-8**.

If the UTF-8 formatting of a custom HTML template file becomes corrupted, then Deployment Editor will refuse to process the file.

15.4.1 Using an ASCII editor instead of a UTF-8 editor

Do not use an ordinary ASCII editor (one without the capability to save text in UTF-8 format) to save your custom HTML template file.

You can get by with using an ASCII-only editor if you use only characters in the lower-127-character range of the ASCII table, which includes English lowercase and uppercase letters, numbers, and many punctuation symbols. This approach works because for these characters UTF-8 uses the same 1-byte encodings as ASCII does. But if you use any character in the upper-129-character range of the ASCII table (these are 1-byte ASCII characters which UTF-8 uses 2 or more bytes to encode), then Deployment Wizard may not be able to process the custom HTML template file or may process it differently than you expect.

Given the risk of your accidentally introducing a character in the upper-129-character ASCII range, as well as for other reasons, you should not use an ASCII-only editor. Instead use a UTF-8 capable editor.

Two related points:

- ▶ 3-byte UTF-8 file signature

If you use an ASCII-only editor to view a UTF-8 formatted file such as Wizard.html you will see the 3 characters `ï»¿` at the beginning of the file listing. These are the ASCII representation of a three-byte signature indicating to a UTF-8-capable editor that the file is in UTF-8 format. Do not delete these characters!

- ▶ UTF-8 warning message box

The following information is implementation dependent and may change in future releases.

You may see a message box warning you that your custom HTML template file may not be in UTF-8 format. This message box appears when the 3-byte UTF-8 file signature expected at the beginning of the file is missing. See Figure 15-3.

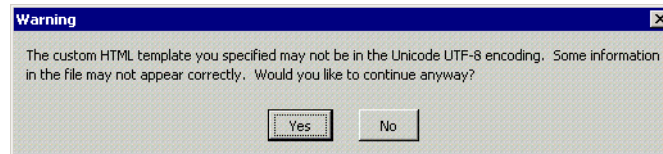


Figure 15-3 Warning message about format of custom HTML template file

This message box allows you to tell Deployment Wizard to continue and process the file anyway. However, when Deployment Wizard attempts to process the file, it will still expect the contents of the file to be encoded in UTF-8 format.

15.5 Parts of the custom HTML template file

This section discusses the different parts of a default custom HTML template file and the significance of each part. In this section JavaServer Pages is abbreviated JSP.

15.5.1 Default custom HTML template file

The following example shows a default custom HTML template file, which has exactly the same contents as Wizard.html.

Example 15-1 Contents of a default custom HTML template file

```
<!doctype html public "-//W3C//DTD HTML 3.2 Final//EN">
<HTML>
<HEAD>
<META http-equiv="content-type" content="text/html; charset=UTF-8">
<!-- SUMMARY -->
</HEAD>

<BODY BACKGROUND="<DW_CODE_BASE>hodbkgnd.gif">
<CENTER>
<IMG src="<DW_CODE_BASE>hodlogo.gif" ALT="hodlogo.gif">
<P>
```

```

<!-- STARTAPPLETPARMS -->
<!-- ENDAPPLETPARMS -->

<!-- SCRIPTS -->
<!-- APPLET -->

</CENTER>
</BODY>
</HTML>

```

15.5.2 Text markers

Do not delete or modify the text markers. Deployment Wizard uses these markers to indicate where certain types of information must be included in the main HTML output file.

The text markers are:

```

<!-- SUMMARY -->
<!-- STARTAPPLETPARMS -->
<!-- ENDAPPLETPARMS -->
<!-- SCRIPTS -->
<!-- APPLET -->

```

15.5.3 DOCTYPE declaration

The DOCTYPE declaration provides HTML version information and other information about the subsequent <HTML> element. Usually you will not need to modify this declaration.

The DOCTYPE declaration is copied to the main HTML output file without modification. Therefore any changes you make in this declaration in the custom HTML template file will be reflected in the main HTML output file.

Adding JavaScript and JSP after the DOCTYPE declaration

You can add JSP and JavaScript statements after the DOCTYPE declaration. The following example shows an unmodified DOCTYPE declaration followed by a JSP block. This is one of the JSP blocks used in an example of overriding HTML parameters in *Planning, Installing, and Configuring Host On-Demand*.

The JSP block below includes JSP method calls, JavaScript statements, and Session Manager JavaScript API method calls.

Example 15-2 A JSP block added after the DOCTYPE declaration

```

<!doctype html public "-//W3C//DTD HTML 3.2 Final//EN">
<%

```

```
// Get a session or create one if necessary and store the hostname
// entered in the form into the session.
HttpSession session = request.getSession(true);
String hostname = request.getParameter("form.hostname");

if (hostname!=null) {
    session.putValue("session.hostname", hostname);
}
%>
```

15.5.4 HEAD Element

You can include HTML elements, JavaScript elements, and JSP elements within the HEAD element. Consult an HTML reference work to determine which HTML elements are valid within this element.

Place all the elements that you add to the HEAD element ahead of the `<!-- SUMMARY -->` marker. This is a suggestion for ease of use.

Information about particular elements or markers:

► **TITLE**

Do not include a TITLE element. Deployment Wizard will generate a TITLE element in the main HTML output file using the text you enter in the Page Title: field of the File Name and Output Format page.

► **META**

You can modify the META element in the default custom HTML template file. This element is copied to the main HTML output file without modification.

You can add additional META elements.

► **<!-- SUMMARY -->**

This marker indicates where Deployment Wizard will include, in the main HTML output file, a lengthy HTML comment summarizing the options selected in the Deployment Wizard pages.

Example 15-3 shows a HEAD element with changes. Note that:

- The original META element has not been modified, even though it could have been.
- A new JavaScript element contains a method called `calculateCelsius()`
- Two additional META elements specify author and copyright information.
- All the changes have been included before the `<!-- SUMMARY -->` marker.

Example 15-3 Example modifications to HEAD element

```

<HEAD>
<META http-equiv="content-type" content="text/html; charset=UTF-8">
<SCRIPT LANGUAGE="JAVASCRIPT">
function calculateCelsius(theForm)
{
    var fahrenheit = parseFloat(theForm.fahrenheitText.value);
    theForm.celsiusText.value = (5.0/9.0 * (fahrenheit - 32.0)).toString();
}
</SCRIPT>
<META NAME=author CONTENT="John Smith">
<META NAME=copyright CONTENT="Copyright 2002, My Company">
<!-- SUMMARY -->
</HEAD>

```

15.5.5 BODY element

You can include HTML elements, JavaScript elements, and JSP elements within the BODY element. Consult an HTML reference work to determine which HTML elements are valid within this element.

You can modify the attributes of the BODY element itself. For example, you might want to specify a different image for the background or a solid background color instead of an image.

In the default custom HTML template file the <DW_CODE_BASE> variable in the BACKGROUND attribute indicates that the file is located in the images subdirectory of the Host On-Demand publish directory. See Example 15-1 on page 574.

15.5.6 Within BODY before <!-- STARTAPPLETPARMS -->

Placement

Elements in this section will be placed in the main HTML output file ahead of the APPLET element that launches the Host On-Demand applet. Consequently displayable elements in this section will be displayed on the Host On-Demand desktop page ahead of the Host On-Demand desktop window.

CENTER, IMG, P

In the default custom HTML template file (see Example 15-1 on page 574) the following lines appear after the line containing the <BODY> tag:

Example 15-4 Three lines following the line containing the <BODY> tag

```
<CENTER>
```

```
<IMG src="<DW_CODE_BASE>hodlogo.gif" ALT="hodlogo.gif">
<P>
```

In connection with these lines, please note:

► **IMG**

You can delete or modify this element.

► **CENTER**

Remember that the <CENTER> tag requires a </CENTER> closing tag later in the file. In the default custom HTML template file, the </CENTER> closing tag occurs on the third line from the bottom of the file.

You will probably want to retain this element. In a default HTML template file this element causes the Host On-Demand desktop window to be displayed centered in the browser page.

► **P**

You will probably want to retain this element or something like it. In a default HTML template file this element prevents the IBM WebSphere Host On-Demand banner from being displayed on the same line as the Host On-Demand desktop window.

Example

Example 15-5 note that:

- The BODY element has been modified to specify non-default colors for background, text, and links.
- The IMG element has been deleted.
- An H1 element is used to display the company name, MyCompany, Inc.
- The CENTER element is retained, so that the link and the Host On-Demand desktop will be centered.
- An A element is used to display a link to the company home page.
- The P element is retained, so that the Host On-Demand desktop will be displayed below the link element instead of on the same line.

Example 15-5 Sample section before <!-- STARTAPPLETPARMS -->

```
<BODY BGCOLOR="GRAY" TEXT="WHITE" VLINK="WHITE" LINK="LIME" ALINK="YELLOW">
<h1 align=center><i><font size=7 face="Impact" color="WHITE">MyCompany,
Inc.</font></i></h1>
<CENTER>
<A HREF="http://www.mycompanyinc.com">MyCompany Home Page</A>
<P>
```

```
<!-- STARTAPPLETPARMS -->
```

The changes in the above example added to a default custom html template file create the following customized browser page:

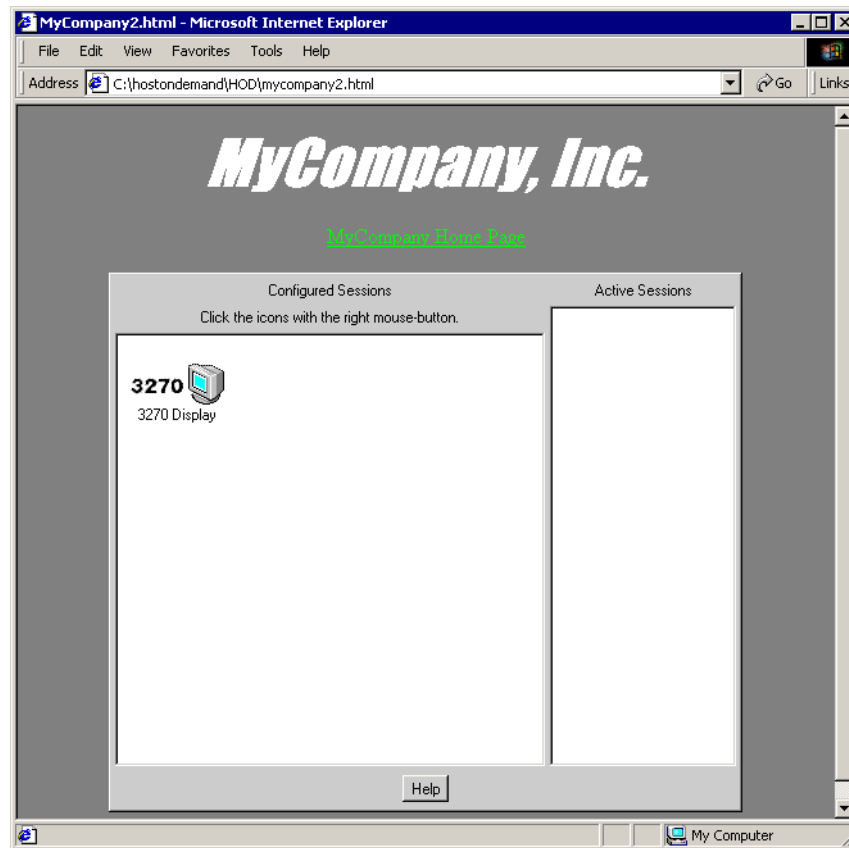


Figure 15-4 Customized browser page with <h1> heading and link

15.5.7 Within BODY between <!-- STARTAPPLETPARMS --> and <!-- ENDPAPPLETPARMS -->

This section should be used only to specify session parameters. Do not specify session parameters anywhere else in the custom HTML template file

The session parameters are documented in the Host On-Demand online help and include the following types of parameters:

- ▶ Session1 parameters
- ▶ Session2 parameters
- ▶ 3270 and 5250 host print parameters
- ▶ The Disable parameter
- ▶ Cached client parameters
- ▶ The “additional parameters” in Deployment Wizard (such as CustomKeyFunctionX and DoNotPrefillUser)
- ▶ The TraceOptions parameter
- ▶ The IPMonitor parameter

To specify a session parameter, use the format `<PARAM NAME=name VALUE=value>`. The following example shows a Disable parameter and an IgnoreWellKnownTrustedCAs parameter.

Example 15-6 Sample section specifying session parameters

```
<!-- STARTAPPLETPARMS -->
<PARAM NAME="Disable"      VALUE="cutpaste;filexfer3207;filexfer5250">
<PARAM NAME="IgnoreWellKnownTrustedCAs" VALUE="true">
<!-- ENDAPPLETPARMS -->
```

15.5.8 Within BODY, between <!-- ENDAPPLETPARMS --> and <!-- APPLET -->

Do not add anything here. Specifically, do not add any elements between `<!-- ENDAPPLETPARMS -->` and `<-- SCRIPTS-->`, and do not add any elements between `<!--SCRIPTS -->` and `<!-- APPLET -->`.

The `<!-- SCRIPT -->` marker indicates where Deployment Wizard will add required JavaScript code. Do not put your custom JavaScript code here.

The `<!-- APPLET -->` marker indicates where the Deployment Wizard will add the APPLET element used to launch the Host On-Demand applet.

In short, this section of your custom HTML template file should be the same as it is in the default HTML template file. The following example illustrates:

Example 15-7 Sample section after <!-- ENDAPPLETPARMS -->

```
<!-- ENDAPPLETPARMS -->

<!-- SCRIPTS -->
<!-- APPLET -->
```

15.5.9 Within BODY, after <!-- APPLET -->

Placement

Elements in this section will be placed in the main HTML output file after the APPLET element that launches the Host On-Demand applet. Consequently displayable elements in this section will be displayed on the Host On-Demand desktop page after the Host On-Demand desktop window.

CENTER, P

If you do not have a <CENTER> tag in the section before <!-- STARTAPPLETPARMS --> then you do not need a closing </CENTER> tag in this section.

If you include a displayable element in this section then you should put a P element ahead of it, so that the displayable element is displayed below the Host On-Demand desktop window instead of on the same line.

Example

In the following example note that:

1. A FORM element has been added which displays checkboxes, a Clear button, and a Submit button.
2. The P element causes the FORM element to be displayed below the Host On-Demand desktop window instead of on the same line.

Example 15-8 Sample section after <!-- APPLET -->

```

<!-- APPLET -->
<P>
<FORM METHOD="POST" ACTION="someplace/foo.cgi"
ENCTYPE="application/x-www-form-urlencoded">
What additional applets would you like to run?<BR>
<INPUT TYPE="checkbox" NAME="choice"
      VALUE="TravelExp">MyCompany Travel Expenses<BR>
<INPUT TYPE="checkbox" NAME="choice"
      VALUE="CapEquip">MyCompany Capital Equipment<BR>
<INPUT TYPE="checkbox" NAME="choice"
      VALUE="PhoneLog">MyCompany Phone Log<BR>
<INPUT TYPE="reset" VALUE="Clear Form">
<INPUT TYPE="submit" VALUE="Submit">
</FORM>

</CENTER>
</BODY>
</HTML>

```

The changes in the above example added to a default custom html template file create the following customized browser page:

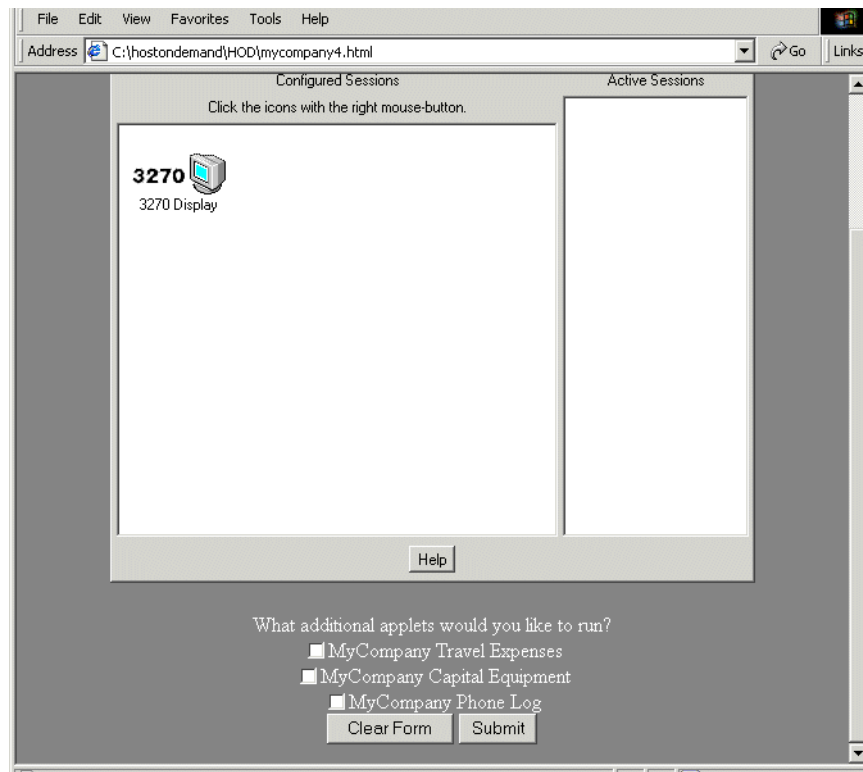


Figure 15-5 Customized browser page with FORM



Modifying session properties dynamically (HTML overrides)

Modifying session properties dynamically means overriding one or more fixed Host On-Demand session property values at the time client users access the HTML files. These session property values can be overridden based on information such as the client's IP address or the time of day. Administrators modify session properties dynamically by completing two main tasks:

- ▶ Modifying the HTML code that is generated from the Host On-Demand Deployment Wizard
- ▶ Deploying a program that runs on the Web server

You can override several session properties, including the LU (logical unit) name, the workstation ID of the client, the port, as well as the host name. For a complete list, refer to “List of session properties that can be overridden” on page 585.

In this chapter, we describe the benefits of modifying session properties dynamically, provide information that will help you decide if using this feature would be useful for your company, and give you step-by-step examples of two common real-life scenarios that can help you take advantage of this time-saving feature.

16.1 The Benefits of modifying session properties dynamically

By allowing you to override session properties at the time users access Host On-Demand HTML files, you can redefine session properties without having to reconfigure the HTML files themselves. Since reconfiguring the HTML files can be very time consuming, this feature is attractive because it allows you to make changes in a fraction of the time normally associated with redefining session properties for users.

When you override session properties, the override values always take precedence over both the initial session properties that were defined by the administrator as well as any user updates to the session properties. When the administrator decides to remove an override value for a session property, the initial configuration for the property becomes active again. This is because the HTML override value is never stored. This type of control allows you to make changes whenever necessary without losing your original settings. Also, the override value is locked, so the user cannot change it.

Administrators can choose from a variety of ways to modify session properties dynamically. Although it always involves modifying the HTML file that you create using the Deployment Wizard, you can write the program that runs on the Web server using a number of different programming languages, including JavaServer Pages (JSPs), servlets, Perl, REXX, or Active Server Pages (ASPs). Although this requires a basic understanding of Common Gateway Interface (CGI) applications, you don't have to be an expert in any of these programming languages. The two example scenarios that we provide in the following sections use JSPs.

16.2 The Need to dynamically modify session properties

Although there are many reasons why administrators may want certain session properties to be overridden at the time users retrieve Host On-Demand sessions, the most common examples involve either overriding a certain property value, such as LU name based on the client's IP address, or providing a form in which

the client can enter information, such as their host name or terminal size. In this chapter, we provide two real-life scenarios based on these common administrator issues that may help you determine if your company should take advantage of this feature.

16.3 List of session properties that can be overridden

Table 16-1 shows which session properties can be overridden, describes them, and provides acceptable values for each session property parameter

Table 16-1 Session properties.

Parameter name	Description	Valid values
Host	Host name or IP address of the target server. Appears as “Destination address” on property windows. Applies to all session types.	Host name or IP address.
Port	The port number on which the target server is listening. Appears as “Destination port” on property windows. Applies to all session types.	Any valid TCP/IP port number.
CodePage	The codepage of the server to which the session will connect. Appears as “Host Code-Page” on property windows. Applies to all session types except FTP.	The numeric portion (for example, 037) of the supported host codepage listed in the session property window.
SessionID	The short name you want to assign to this session (appears in the OIA). It must be unique to this configuration. Appears as “Session ID” on property windows. Applies to all session types.	One character: A-Z.

Parameter name	Description	Valid values
LUName	The name of the LU or LU Pool, defined at the target server, to which you want this session to connect. Appears as "LU or Pool Name" on property windows. Applies to 3270 Display and 3270 Printer session types.	The name of an LU or LU Pool.
WorkstationID	The name of this workstation. Appears as "Workstation ID" on property windows. Applies to 5250 Display and 5250 Print session types.	A unique name for this workstation.
ScreenSize	Defines the number of rows and columns on the screen. Appears as "Screen Size" on property windows. Applies to 3270 Display, 5250 Display, and VT Display session types.	value=rows x columns 2=24x80 (3270, 5250, VT) 3=32x80 (3270) 4=43x80 (3270) 5=27x132 (3270, 5250) 6=24x132 (VT) 7=36x80 (VT) 8=36x132 (VT) 9=48x80 (VT) 10=48x132 (VT) 11=72x80 (VT) 12=72x132 (VT) 13=144x80 (VT) 14=144x132 (VT) 15=25x80 (VT) 16=25x132 (VT)
SLPScope	Service Location Protocol (SLP) Scope. Appears as "Scope" under "SLP Options" on property windows. Applies to 3270 Display, 3270 Printer, 5250 Display, and 5250 Printer session types.	Contact your administrator to get the correct value for this field.

Parameter name	Description	Valid values
SLPAS400Name	Connects a session to a specific iSeries. Appears as “AS/400 Name (SLP)” on property windows. Applies to 5250 Display and 5250 Printer session types.	The fully qualified SNA CP name (for example, USIBMNM.RAS400B).
SSLCertificateSource	The certificate can be kept in the client's browser or dedicated security device, such as a smart card; or, it can be kept in a local or network-accessed file. Appears as “Certificate Source” on property windows. Applies to 3270 Display, 3270 Printer, 5250 Display, 5250 Printer, and VT Display session types.	The value is SSL_CERTIFICATE_IN_C SP for a certificate in a browser or security device. The value is SSL_CERTIFICATE_IN_U RL for a certificate in a URL or file.
SSLCertificateURL	Specifies the default location of the client certificate. Appears as “URL or Path and Filename” in property windows. Applies to 3270 Display, 3270 Printer, 5250 Display, 5250 Printer, and VT Display session types.	The URL protocols you can use depend on the capabilities of your browser. Most browsers support HTTP, HTTPS, FTP, and FTPS.
FTPUser	Specifies the user ID the session uses when connecting to the FTP server. Appears as “User ID” on property windows. Applies to FTP session types.	A valid user ID.
FTPPassword	Specifies the password the session uses when connecting to the FTP server. Appears as “Password” on property windows. Applies to FTP session types.	A valid password.

Parameter name	Description	Valid values
UseFTPAnonymousLogon	Enables the session to log in to an FTP server using anonymous as the user ID. Appears as "Anonymous Login" on property windows. Applies to FTP session types.	Yes or No.
FTPEmailAddress	Specifies the e-mail address to use when connecting to the FTP server while using Anonymous Login. Appears as "E-mail Address" on property windows. Applies to FTP session types.	A valid e-mail address.
Netname	The name of the terminal resource to be installed or reserved. If this field is blank, the selected terminal type is not predictable. Applies to CICS sessions only.	A valid terminal resource name.

16.4 Scenario 1: Overriding the LU name based on the client's IP address

A company with over 1000 Host On-Demand clients prefers its users to have the same LU name every time they log on to their 3270 Display and Printer sessions to better allocate resources in their mainframe environment. However, the company's administrator does not want to spend unnecessary time having to specify the LU names directly in the session definitions. Instead, he decides to modify the session properties dynamically so that the LU name is determined from the IP address of the client at the time users access the HTML file.

In this scenario, the administrator creates a text file called `luname.table` that pairs LU names with IP addresses so that the proper LU names are assigned to the clients based on their corresponding IP addresses. He creates a JSP file that reads this text file into a properties variable.

In the following section, we provide the steps that this administrator takes to modify the session properties, including editing the HTML code and deploying a JSP on the Web server. Following the steps, we show you an example of the completed JSP code.

Note: The following instructions are specific for this company's environment, which includes Windows 2000 operating system and IBM Websphere Application Server Advanced Edition Version 4.0. Other environments may require different steps.

16.4.1 Steps to modify the session properties

Follow these steps in order to override the LU name based on the client's IP address.

1. Use the Deployment Wizard to set up your initial HTML file. On the Additional Options window, click Advanced Options and go to the Other tab. Type the relative path `/hod/` in the Codebase field (Figure 16-1). Save the HTML file to the default Host On-Demand publish directory `hostondemand/HOD`. Your HTML file is now in the same directory with Host On-Demand's archive files.

Important: Codebase refers to the installed Host On-Demand publish directory, not the directory where you publish your Deployment Wizard files. Although you can enter a fully qualified URL in the Codebase field, we strongly recommend that you enter the relative path /hod/ for the default publish directory when modifying session properties dynamically. If you enter a fully qualified URL, any users who specify the host name in a different manner than you specified as the Codebase will not be able to access the files, even if the DNS entries resolve to the same IP address.

For more information about Codebase and which files are created by the Deployment Wizard, refer to Chapter 14., “Deployment Wizard” on page 529.

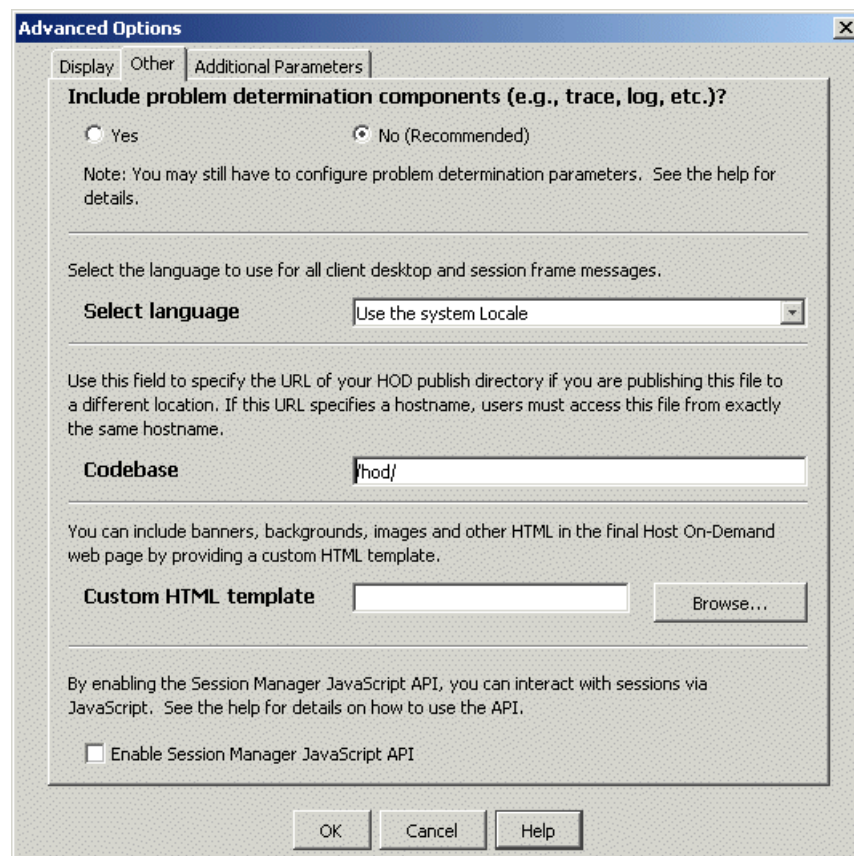


Figure 16-1 Advanced Options window of the Deployment Wizard

Note the following points:

- The Deployment Wizard generates some of the HTML code using JavaScript, and the HTML parameters are specified within a JavaScript array or by means of JavaScript document.write statements.
 - The format of the HTML code generated by the Deployment Wizard depends on whether you select Java 1, Java 2, or Auto Detect for the Client Java Type and whether or not you cache the Host On-Demand applet on your users' machines. Both of these options appear on the Additional Options window of the Deployment Wizard. (This scenario uses a cached Java 1 HTML file.)
2. Read a file called luname.table, which you will create in the next step, into a properties variable by adding the following lines to your JSP code:


```
<%java.util.Properties lunames = new java.util.Properties();
    lunames.load(new java.io.FileInputStream("c:\\luname.table"));%>
```

(This code assumes that the luname.table file is located on your C: drive.)
See [1](#) on page 604 to see where these lines are located in the JSP code.
 3. Create a file called luname.table file that contains pairs of LU names and IP addresses. The user's browser requests the IP address of the user, and the corresponding LU name is looked up from this luname.table file and read into a properties variable that can be used by the JSP file. The format of the lines should be ipaddress=luname, for example, 9.33.67.5=luname.
 4. Enable HTML overrides by including the EnableHTMLOverrides parameter in your HTML file and setting it to a value of true, as in the following example.


```
document.write('<PARAM NAME="EnableHTMLOverrides" VALUE="true">');
```

Refer to [2](#) on page 605 to see where this line is located in the JSP code.
 5. List the sessions to be overridden by including the TargetedSessionList parameter and setting the value as the exact names of the sessions that you are modifying dynamically. Since you may have multiple sessions associated with your HTML file, you need to specify which sessions you are modifying.

In the following example, the administrator set the value as the list of session names—3270 Display and 5250 Display—and separated them with commas.


```
document.write('<PARAM NAME="TargetedSessionList"
    VALUE="3270 Display,5250 Display">');
```

Refer to [3](#) on page 605 to see where these lines are located in the JSP code.
 6. Specify the override itself by including an HTML parameter for each of the properties that you are overriding. The name of the HTML parameter should be the name of the session property, and the value should be a value for the desired override.

In this scenario, by inserting the following code, the administrator changed the LUName session parameter for the session called 3270 Display. The LU name is set to the LU name corresponding to the IP address in c:\\luname.table.

```
document.write('<PARAM NAME="Luname" VALUE="3270  
Display=<%=lunames.get(request.getRemoteAddr())%>">');
```

The request.getRemoteAddr() is Java code for getting the IP address of the client requesting the JSP. This IP address is used to look up the corresponding LU name in the variable containing the luname.table file.

Refer to [4](#) on page 605 to see where these lines are located in the JSP code.

7. Add a ConfigBase parameter to the JSP file.

```
document.write('<PARAM NAME="ConfigBase" Value="http://host_name/hod/">');
```

Important: Similar to defining /hod/ as the Codebase in Step 1, the ConfigBase parameter is necessary because you will eventually deploy your JSP file to a location that is different than the default publish directory, and the Host On-Demand applet needs to know how to find the session configuration files located in the hostondemand/HOD/HODData directory. These files are created at the same time you save your Deployment Wizard HTML file to the publish directory.

Unlike Codebase, the ConfigBase parameter requires a fully qualified URL. ConfigBase is a term that is specific to Host On-Demand.

For more information, refer to Section 14.5, “Files created by the Deployment Wizard” on page 562.

Refer to [5](#) on page 605 to see where this line is located in the JSP code.

8. Compare your new JSP code to make sure that it is identical to the code in the file provided in “Completed HTML file after edits” on page 604. If not, make the necessary changes. If so, change the file extension from HTML to JSP and save it to your machine.

9. Start the WebSphere Application Server Advanced Edition Version Application Assembly Tool by selecting **Start > Programs > IBM WebSphere > Application Server V4.0 AE > Application Assembly Tool**. The Application Assembly Tool helps you package your application according to the J2EE specifications.

In this scenario, since we are packaging a JSP file, we need to create a Web module using the Create Web Module Wizard (Figure 16-2). Highlight the wizard and click OK.

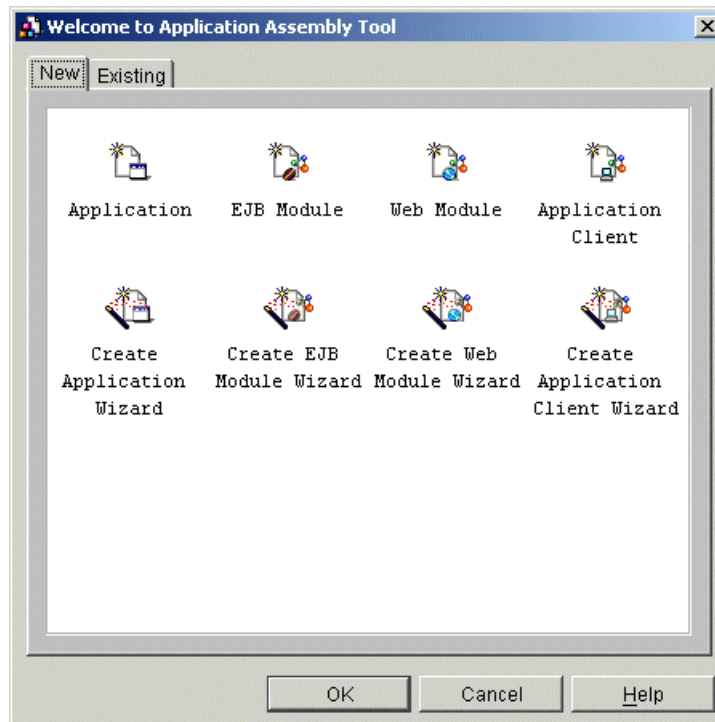


Figure 16-2 Application Assembly Tool

10. On the Specifying Web Module Properties window (Figure 16-3), type the name of your WAR (Web archive) file in the File name field. This WAR file represents your Web module. Although you will see the default name Web_Name_Module1.war in this field, you may want to change it to another name. Note that the File name field is the only field on this window that requires you to enter information (as indicated by the asterisk). Once you name your WAR file, click Next.

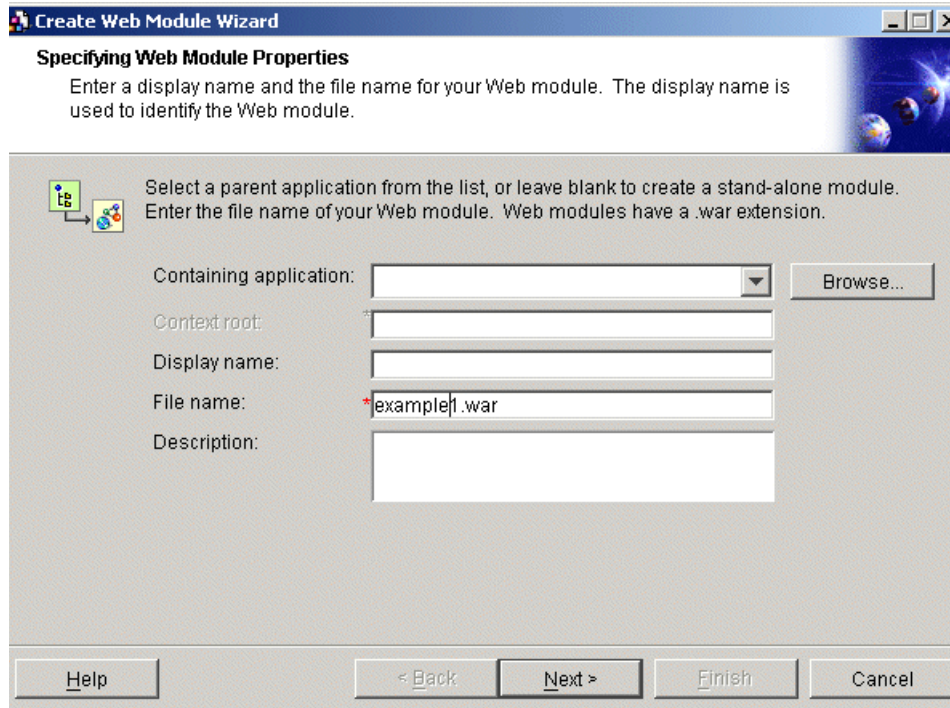


Figure 16-3 Specifying Web Module Properties

11. Click through four windows until you get to the Adding Web Components window, as shown in Figure 16-4. Click New. This launches a second wizard called Create Web Component Wizard. Now, you have two open wizards on your work space at the same time.

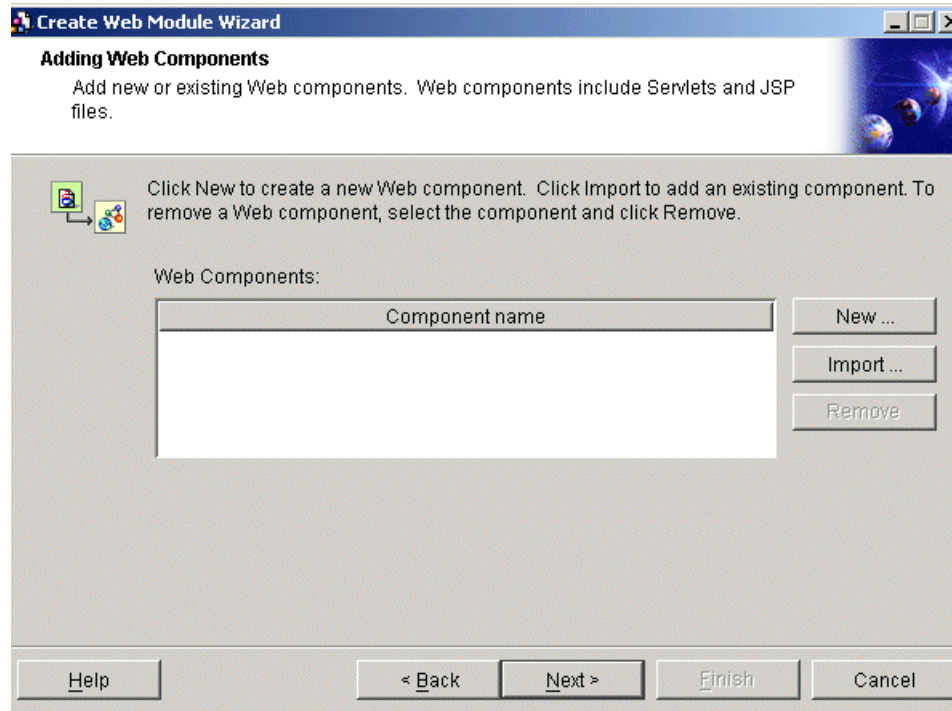
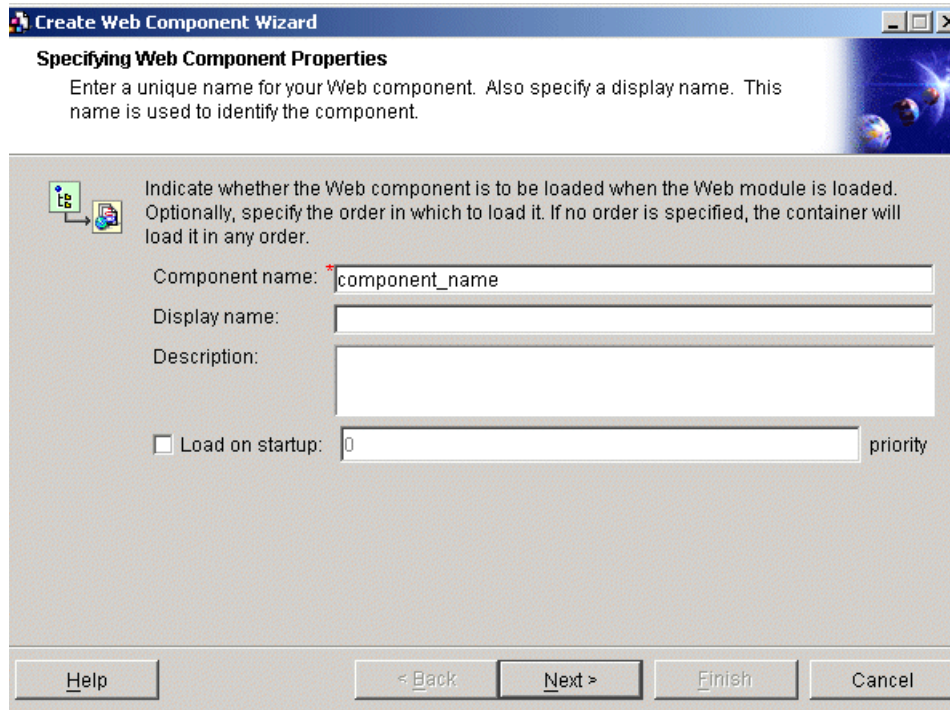


Figure 16-4 Adding Web Components

12. The first window of the Create Web Component Wizard is called Specifying Web Component Properties (Figure 16-5). In the Component name field, type the name of the Web component. The name of this component must be unique. You can also provide a display name and component description, but these fields are not required.



The screenshot shows a Windows-style dialog box titled "Create Web Component Wizard". The main heading is "Specifying Web Component Properties". Below the heading is a text box with the instruction: "Enter a unique name for your Web component. Also specify a display name. This name is used to identify the component." To the right of this text is a small graphic of a globe. Below the instruction, there is a section with a small icon of a web browser and a text box that says: "Indicate whether the Web component is to be loaded when the Web module is loaded. Optionally, specify the order in which to load it. If no order is specified, the container will load it in any order." Below this, there are four input fields: "Component name:" with a red asterisk and the text "component_name" entered; "Display name:" which is empty; "Description:" which is empty; and "Load on startup:" which is a checkbox that is unchecked, followed by a text box containing the number "0" and the word "priority" to its right. At the bottom of the dialog are five buttons: "Help", "< Back", "Next >", "Finish", and "Cancel".

Figure 16-5 Specifying Web Component Properties

13. On the Specifying Web Component Type window (Figure 16-6), select JSP and then click Browse. After the Select file for JSP file appears, click Browse to select the root directory of the JSP file that you created in Step 8. Once you find and highlight the root directory, click Select. The JSP file contained in the directory appears in the right frame of the window. Highlight the JSP file and click OK. After you click OK, the name of the JSP file appears in the JSP field in the wizard window.

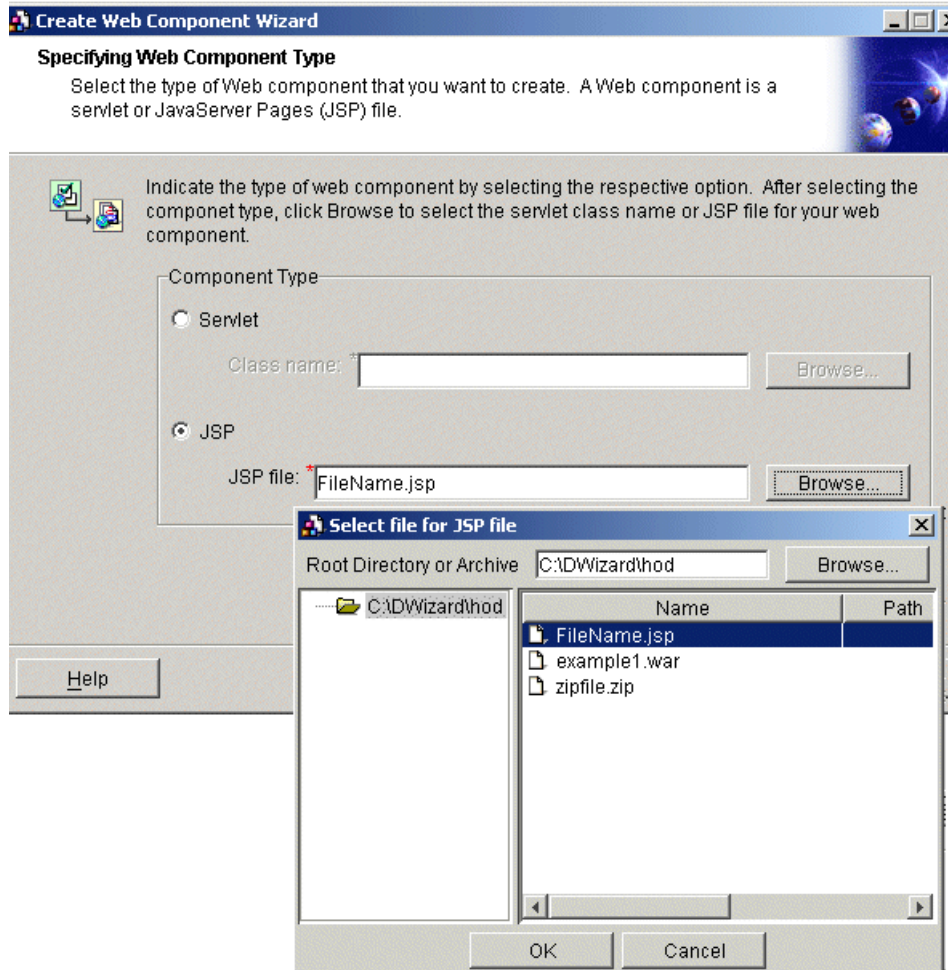


Figure 16-6 Specifying Web Component Type

14. Click Next until you reach the final window of the Create Web Component Wizard. Click Finish. Click Next until you reach the final window of the Create Web Module Wizard. Click Finish.

After you close both wizards, go to **File > Save As** in the Application Assembly Tool window, and type the name of your WAR file in the File name field. Be sure that the file extension of your file is .war. Save your WAR file in the WebSphere\Appserver\InstallableApps directory (Figure 16-7).

Once you save your file, a window appears that tells you that it was saved successfully. Click OK, and close out of the Application Assembly Tool.

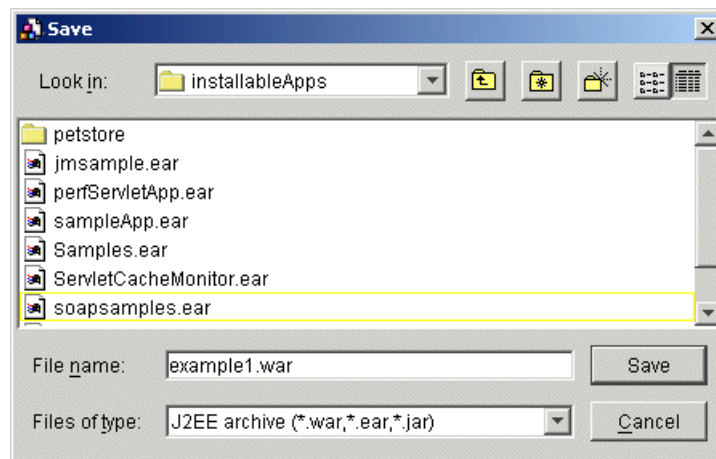


Figure 16-7 Save WAR file

15. Open the Administrative Console by selecting **Start > Programs > IBM WebSphere > Application Server V4.0 AE > Administrator's Console**. Be sure that the administration server is running, or you will get an error message. To start the administration server, go to **Programs > IBM WebSphere > Application Server V4.0AE > Start Admin Server**.

16. Once the Administrative Console is open, go to the Console drop-down menu and select **Wizards > Install Enterprise Application**. In the Specifying the Application or Module window (Figure 16-8), select Install stand-alone module (*war, *jar). By default, the name of your local host appears in the Browse for file on node field.

Type the Path, Application name, and Context root for web module:

- For Path, browse to the WebSphere\Appserver\InstallableApps directory and select the WAR file that you saved in Step 14.
- For Application name, type the name of the application. This will be the name of the EAR file that WebSphere creates later.
- For the Context root for web module, type the name of the context root. The first character must be a backwards slash (/). The context root is the part of the session URL that comes immediately after the host name, for example, `http://host_name/context_root_name/FileName.jsp`.

Click Next.

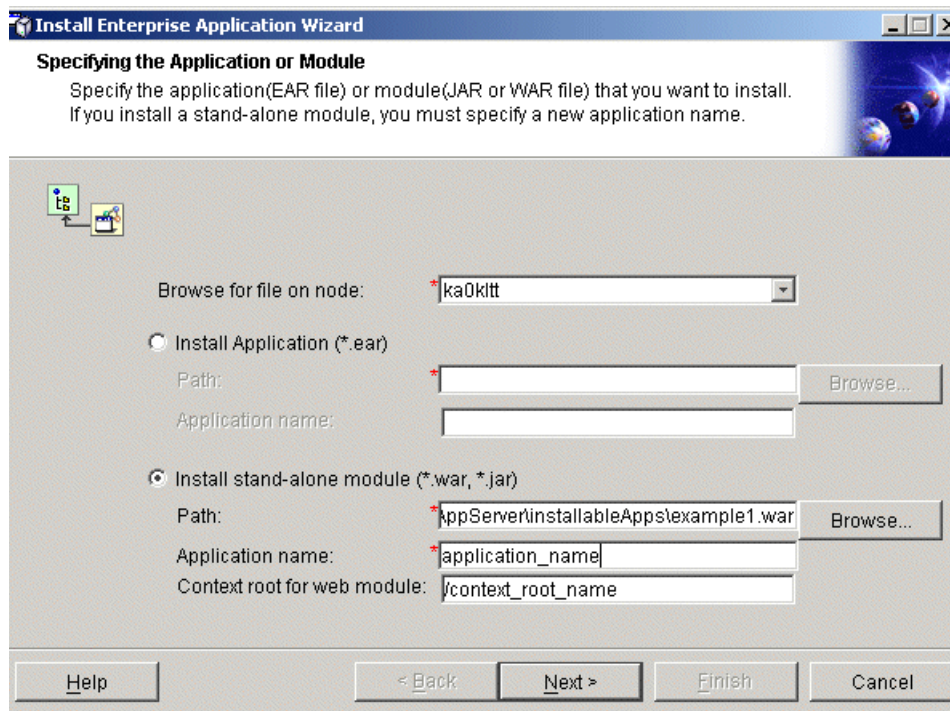


Figure 16-8 Specifying the Application or Module

17. Continue clicking through several windows until you reach the last window of the Install Enterprise Application Wizard (Figure 16-9). At this point, you are installing the Web module, which will be placed into an EAR file and expanded in the installedApps directory. This is the final stage before the application is actually deployed. Click Finish.

If completed successfully, an information dialog appears. Click OK and close the Administrative Console.

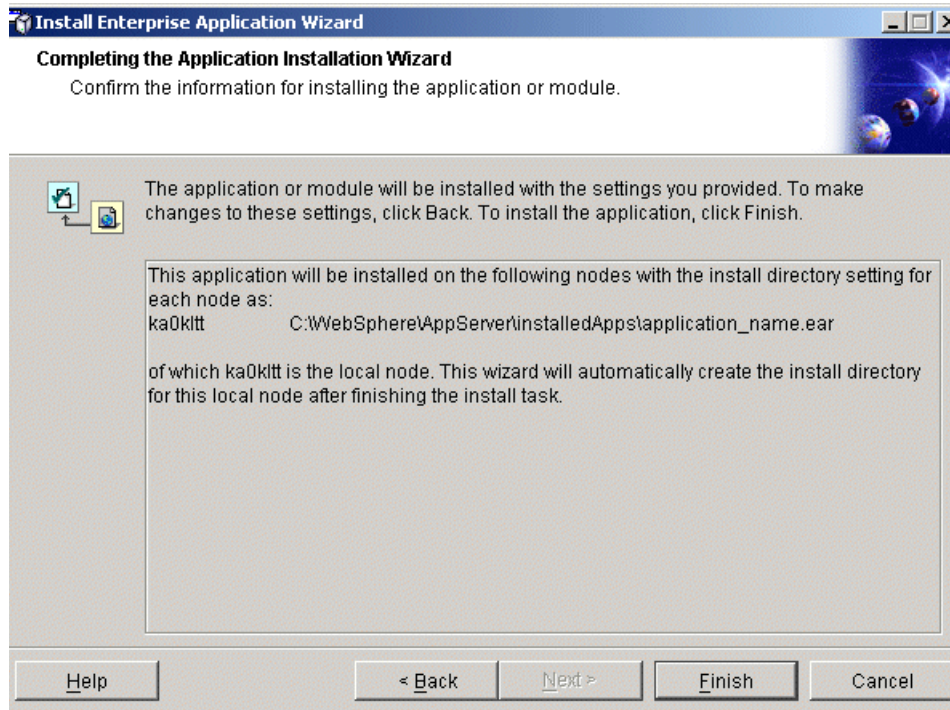


Figure 16-9 Completing the Application Installation Wizard

18. Since you have changed certain WebSphere configuration properties, you must regenerate the Web server plug-in configuration. Open the Administrative Console by selecting **Start > Programs > IBM WebSphere > Application Server V4.0 AE > Administrator's Console**.

In the left navigation bar, select the plus sign (+) to expand Nodes. Right click on your local host and select Regen Webserver Plug-in (Figure 16-10). You will not see any visual indication that the plug-in is regenerating, but you will know when it completes the regeneration process by viewing the Event Message Log in the bottom frame of the Administrative Console. Regenerating the plug-in usually takes a few minutes.

Important: If your Web server is on a separate machine, then you will have to move the plugin-cfg.xml file to the Web server machine after the plug-in regeneration. This file is located in the WAS_ROOT/AppServer/config directory.

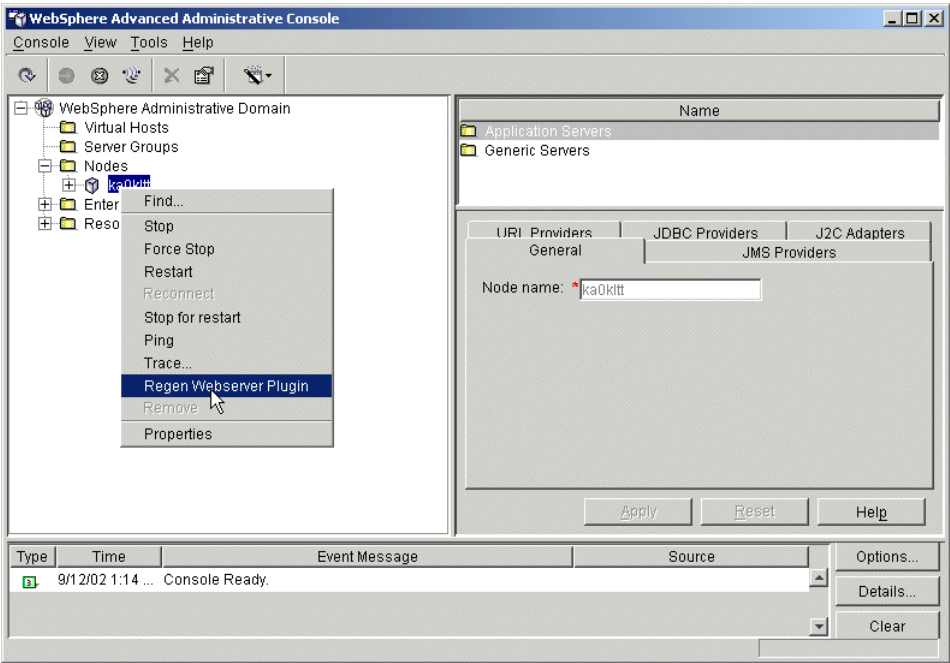


Figure 16-10 Regen Webserver Plug-in

19. Stop and restart WebSphere and your Web server. To stop WebSphere, go to **Start > Settings > Control Panel > Administrative Tools > Services > IBM WS AdminServer 4.0**. Right click on IBM WS AdminServer 4.0 and select Stop. Once the administration server stops, right click on IBM WS AdminServer 4.0 and select Start. It usually takes a few minutes to complete this process.
20. Finally, point your browser to `http://your_host/context_root_name/FileName.jsp` and retrieve the Host On-Demand session. In the URL, your_host is the name of your local host, context_root_name is the name of the context root, and FileName.jsp is the name of your JSP file. Once the session icon appears, right click on the icon and select Properties. In the Connection tab (Figure 16-11), check to make sure that the LU name that you specified in luname.table for your IP address is in the LU or Pool Name field on the window.

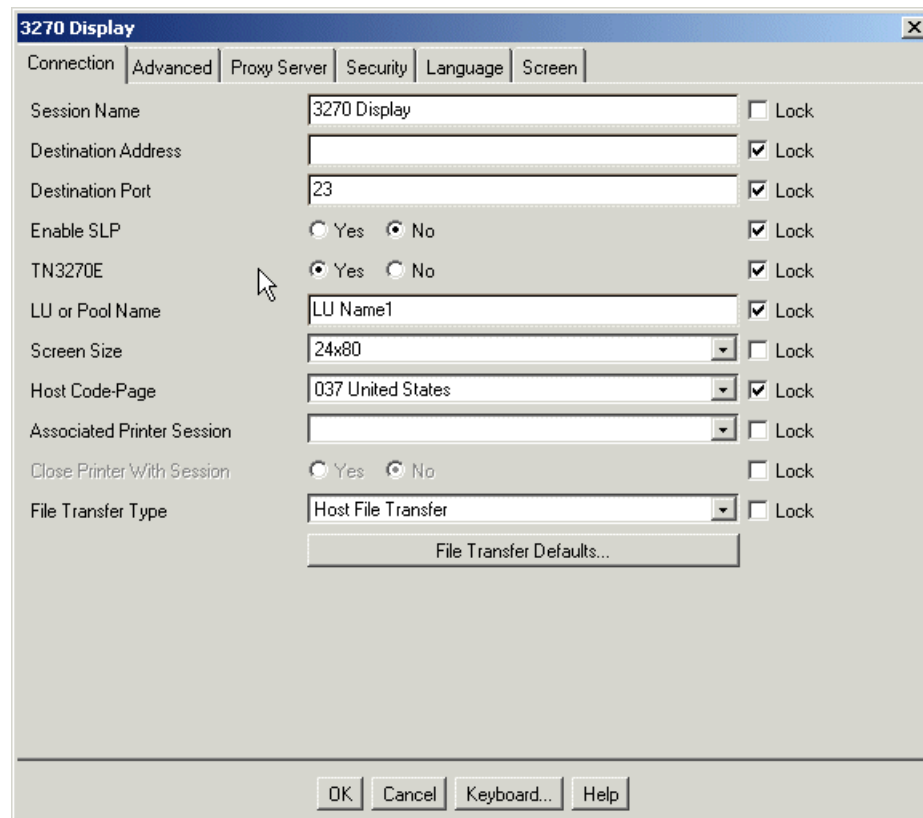


Figure 16-11 Connection tab of Deployment Wizard session properties

16.4.2 Troubleshooting Scenario 1

If your HTML override is not successful, take the following steps to help determine the cause:

1. Use a static value in your HTML file to see if your HTML overrides mechanism is working. For example, instead of coding the HTML file so that the IP address of the user and the corresponding LU name are looked up from the `luname.table` file (Step 2), replace the table lookup with a static LU name value, as in the following example:

```
document.write('<PARAM NAME="Luname" VALUE="3270  
Display=Static_LU_name">');
```

Refer to [4](#) on page 605 to see where these lines are located in the JSP file.

Once you complete the steps to override the HTML parameter, open the session properties and make sure that the static LU name is in the LU or Pool Name field of the Connection tab.

2. Add the debug parameter `DebugCode` to your HTML file generated by the Deployment Wizard and set it to a value of 65535:

```
document.write('<PARAM NAME="DebugCode" VALUE="65535">');
```

This parameter sends useful debugging information to the Java Console, such as whether or not your Codebase, Configbase, and TargetedSessionList parameters and session properties were read correctly.

Refer to [6](#) on page 606 to see where this line is located in the JSP file.

3. Check your access log file for messages that contain the code '404'. This is the standard code for "Page Not Found." This access log file is generated by your Web server and is located in your Web server directory.

16.4.3 Completed HTML file after edits

The following code is the original HTML code generated from the Deployment Wizard plus the additions the administrator made to modify session properties dynamically. The lines added to the original file are displayed in bold. The callout boxes in the left margin show you the corresponding step from the previous section.

```
<!doctype html public "-//W3C//DTD HTML 3.2 Final//EN">
```

```
<%
```

1

```
Properties lunames = new Properties();  
lunames.load(new FileInputStream("c:\\luname.table"));  
%>
```

```
<!-- HOD WIZARD HTML -->
```

```
<HTML>
```

```
<HEAD>
```

```
<META content="text/html; charset=UTF-8">
```

```
<!-- TITLE Begin -->
```

```
<TITLE>Example1</TITLE>
```

```
<!-- TITLE End -->
```

```
<!-- SUMMARY Begin -->
```

```
<!--
```

```
Configuration Model
```

```
    What configuration model would you like to use?
```

```
    -HTML-based model
```

```
Sessions created
```

```
    -3270 Display
```

```
    -5250 Display
```

```
Additional Options
```

```
    -Allow users to save session changes? = True
```

```
    -Cached = True
```

```
    -Java Type = java1
```

```
Disable Functions
```

```
Preload Options
```

```
    -5250 Sessions = True
```

```
    -Change Session Properties = True
```

```
    -FTP Sessions = True
```

```
    -3270 Sessions = True
```

```
Server Connection Options
```

```
Cache Options
```

```
    Basic Options
```

```
    -Debug = False
```

```
    -Height (in pixels) = 250
```

```
    -Width (in pixels) = 550
```

```
    Cache Client Upgrade Option
```

```
    -Percent of users who can upgrade by default = 100
```

```

        -Prompt user (user decides foreground or background)
Advanced Options
    Display
        -Standard Host On-Demand Client
        -Applet size = Autosize to browser
        -Maximum sessions = 26
    Other
        -Locale = Use the system Locale
        -Debug = False
        -HTML Template = Default
Additional Parameters
    -None
-->
<!-- SUMMARY End -->
</HEAD>

<BODY BACKGROUND="hodbkgnd.gif">
<CENTER>
<IMG src="hodlogo.gif" ALT="hodlogo.gif">
<P>

<SCRIPT LANGUAGE="JavaScript">
function writeAppletParameters()
{
    document.write("");
}
</SCRIPT>

<SCRIPT LANGUAGE="JAVASCRIPT" SRC="CachedJ1.js"></SCRIPT>
<SCRIPT LANGUAGE="JAVASCRIPT">
var hod_Height='80%';
var hod_Width='80%';
document.write('<APPLET ARCHIVE="CachedAppletSupporter.jar" MAYSCRIPT
NAME="HODApplet"
CODE="com.ibm.eNetwork.HOD.cached.appletloader.CachedAppletLoader"
WIDTH="'+hod_Width+'" HEIGHT="'+hod_Height+'">');
document.write('<PARAM NAME="Cabinets"
VALUE="CachedAppletSupporter.cab">');
document.write('<PARAM NAME="CachedClient"                VALUE="true">');
document.write('<PARAM NAME="ParameterFile"
VALUE="HODData\\Example1\\params.txt">');
document.write('<PARAM NAME="JavaScriptAPI" VALUE="false">');

2 document.write('<PARAM NAME="EnableHTMLOverrides" VALUE="true">');

3 document.write('<PARAM NAME="TargetedSessionList"
    VALUE="3270 Display,5250 Display">');

4 document.write('<PARAM NAME="Luname" VALUE="3270

```

```
Display=<%=lunames.get(request.getRemoteAddr())%>">');
```

5

```
document.write('<PARAM NAME="ConfigBase"
VALUE="http://host_name/hod/">');
```

6

```
document.write('<PARAM NAME="DebugCode"
VALUE="65535">');
```

```
writeAppletParameters();
document.write("</APPLET>");
</SCRIPT>
```

```
<P>
<SCRIPT LANGUAGE="JavaScript">
var hod_AppName='';
var hod_Preloadlist='HABASE;HODBASE;HODIMG;HACP;HAFNTIB;
HAFNTAP;HA3270;HODCFG;HAFTP;HA5250';
var hod_Debugcomponents='false';
var hod_Debugcachedclient='false';
var hod_Upgradepromptresponse='Prompt';
var hod_Upgradepercent='100';
var hod_Framewidth='550';
var hod_Frameheight='250';

function isBookmark(mySearch) {
  if (mySearch.length < 2) {
    return false;
  } else {
    return (mySearch.toLowerCase().indexOf('launch=') != -1);
  }
}

if (hod_AppName == '') {
  if (isBookmark(window.location.search.substring(1)))
    hod_AppName = 'com.ibm.eNetwork.HOD.SessionLauncher';
  else
    hod_AppName = 'com.ibm.eNetwork.HOD.HostOnDemand';
}

function getHODFrame() {
  return self;
}
```

```
document.write('<APPLET ARCHIVE="CachedAppletSupporter.jar" MAYSCRIPT
NAME="CachedAppletSupporter"
CODE="com.ibm.eNetwork.HOD.cached.appletsupport.CachedAppletSupportApplet"
WIDTH="2" HEIGHT="2">');
document.write('<PARAM NAME="Cabinets"
VALUE="CachedAppletSupporter.cab">');
```

```
document.write('<PARAM NAME="DebugComponents"
VALUE="'+hod_Debugcomponents+' ">');
document.write('<PARAM NAME="PreloadComponentList"
VALUE="'+hod_Preloadlist+' ">');
document.write('<PARAM NAME="DebugCachedClient"
VALUE="'+hod_Debugcachedclient+' ">');
document.write('<PARAM NAME="CachedClientSupportedApplet"
VALUE="'+hod_AppName+' ">');
document.write('<PARAM NAME="InstallerFrameWidth"
VALUE="'+hod_Framewidth+' ">');
document.write('<PARAM NAME="InstallerFrameHeight"
VALUE="'+hod_Frameheight+' ">');
document.write('<PARAM NAME="UpgradePromptResponse"
VALUE="'+hod_Upgradepromptresponse+' ">');
document.write('<PARAM NAME="UpgradePercent"
VALUE="'+hod_Upgradepercent+' ">');
document.write('</APPLET>');
</SCRIPT>

</CENTER>
</BODY>
</HTML>
```

16.5 Scenario 2: Specifying the host name using an HTML form

A company decides to display a simple form that prompts its Host On-Demand client users to enter the host name. The administrator uses HTML overrides so that the host name entered by the user overrides the host name that was configured in the 3270 host session. In this case, the HTML form posts to a JSP program, which stores and uses the form data to override the host name.

In the following section, we provide the steps that this administrator takes to modify session properties dynamically, including editing her Deployment Wizard HTML file, creating a simple HTML file that users can enter their host names, and deploying the JSP. Because she has already followed the steps in Scenario 1 on page 589, she does not need to recreate a Web module using WebSphere Application Server. This means she can simply save the JSP file to the same directory where she deployed her JSP file in Scenario 1.

Once we show you the steps this administrator takes to modify session properties dynamically, we show you an example of the completed JSP code and offer troubleshooting advice.

Note: The following instructions are specific for this company's environment, which includes Windows 2000 operating system and IBM WebSphere Application Server Advanced Edition Version 4.0. Other environments may require additional steps.

16.5.1 Steps to modify the session properties

Follow these steps in order to display a simple form that allows users to enter in their host name and override the host name specified in the session properties.

1. Use the Deployment Wizard to set up your initial HTML file. On the Additional Options window, click Advanced Options and go to the Other tab. Type the relative path `/hod/` in the Codebase field (Figure 16-1). Save the HTML file to the default Host On-Demand publish directory `hostondemand/HOD`. Your HTML file is now in the same directory with Host On-Demand's archive files.

Important: Codebase refers to the installed Host On-Demand publish directory, not the directory where you publish your Deployment Wizard files. Although you can enter a fully qualified URL in the Codebase field, we strongly recommend that you enter the relative path `/hod/` for the default publish directory when modifying session properties dynamically. If you enter a fully qualified URL, any users who specify the host name in a different manner than you specified as the Codebase will not be able to access the files, even if the DNS entries resolve to the same IP address.

For more information about Codebase and which files are created by the Deployment Wizard, refer to Chapter 14., "Deployment Wizard" on page 529.

Take note of the following points:

- The Deployment Wizard generates some of the HTML using JavaScript, and the HTML parameters are specified within a JavaScript array or by means of JavaScript `document.write` statements.
 - The format of the HTML code generated by the Deployment Wizard depends on whether you select Java 1, Java 2, or Auto Detect for the Client Java Type and whether or not you select to cache the Host On-Demand applet on your users' machines. Both of these options are located on the Additional Options window of the Deployment Wizard. (This scenario uses a non-cached Auto Detect HTML file.)
2. Use the following code to create a simple HTML form called `HODForm.html` that allows users to input their host names. The form posts to a JSP program called `example2.jsp`. Save this file to the default Host On-Demand publish directory `hostondemand/HOD`.

```
<form method="POST" action="context_root/example2.jsp">  
Hostname <input name="form.hostname"><br>  
<input type="submit">  
</form>
```

Figure 16-12 shows what the HTML file looks like in a browser:

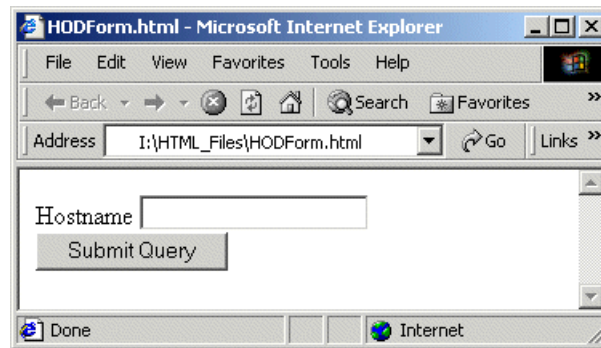


Figure 16-12 Browser view of HODForm.htm

3. The following steps require you to edit your Deployment Wizard HTML code:

- a. Get a session and store the host name that the user enters into the form by including the following code in your Deployment Wizard file.

```
HttpSession session = request.getSession(true);
String hostname = request.getParameter("form.hostname");

if (hostname!=null) {
    session.putValue("session.hostname", hostname);
}
```

Refer to [1](#) on page 614 to see where this code is located in the JSP file.

When using forms, the form data needs to be retained across requests to the program. This is because Host On-Demand HTML files reload themselves for Java detection and for bookmarking support. If Java 1 is selected and bookmarking support is disabled if using the configuration server-based model, the page will not need to reload and there is no need to retain the form data.

- b. In the following line, change `z_example2.html` to `z_example2.jsp`, where `example2.html` is the name of your Deployment Wizard file:

```
var hod_FinalFile = 'z_example2.jsp';
```

Refer to [2](#) on page 615 to see where this line is located in the JSP code.

- c. If you want a JavaScript alert window to appear when you first access the JSP file that tells you if your program generated the proper syntax, change `false` to `true` in the following line:

```
var hod_DebugOn = true;
```

Refer to **3** on page 615 to see where this line is located in the JSP code.

- d. Enable HTML overrides by including the EnableHTMLOverrides parameter in your HTML file and setting it to a value of true, as in the following example.

```
hod_AppletParams[4] = '<PARAM NAME="EnableHTMLOverrides" VALUE="true">';
```

Refer to **4** on page 615 to see where this line is located in the JSP code.

- e. List the sessions to be overridden by including the TargetedSessionList parameter and setting the value as the exact names of the sessions that you are modifying dynamically. Since you may have multiple sessions associated with your HTML file, you need to specify which sessions you are modifying.

In the following example, the administrator set the value as the list of session names—3270 Display and 5250 Display—and separated them with commas.

```
hod_AppletParams[5] = '<PARAM NAME="TargetedSessionList"
VALUE="3270 Display,5250 Display">';
```

Refer to **5** on page 615 to see where these lines are located in the JSP code.

- f. Change the host or destination address session parameter for the session named 3270 Display. In the following example, the host is set to the value saved in the JSP session from the HTML form.

```
hod_AppletParams[6] = '<PARAM NAME="Host" VALUE="3270
Display=<%=session.getValue("session.hostname")%>">';
```

Refer to **6** on page 615 to see where these lines are located in the JSP code.

Tip: When you are initially testing your changes, you may want to use a constant value to verify that the syntax is correct before you insert a calculated value.

- g. Add a ConfigBase parameter to the JSP file.

```
hod_AppletParams[7] = '<PARAM NAME="ConfigBase"
Value="http://host_name/hod/">';
```

Important: Similar to defining `/hod/` as the Codebase in Step 1, the ConfigBase parameter is necessary because you will eventually deploy your JSP file to a location that is different than the default publish directory, and the Host On-Demand applet needs to know how to find the session configuration files located in the `hostondemand/HOD/HODData` directory. These files are created at the same time you save your Deployment Wizard HTML file to the publish directory.

Unlike Codebase, the ConfigBase parameter requires a fully qualified URL. ConfigBase is a term that is specific to Host On-Demand.

For more information, refer to Section 14.5, “Files created by the Deployment Wizard” on page 562.

Refer to **7** on page 615 to see where this line is located in the JSP code.

- h. Find the `z_example2.html` file, where `example2.html` is the name of your Deployment Wizard HTML file that you created in Step 1, and change the file extension to `.jsp`. This file is located in the `hostondemand/HOD` directory. Open it and copy it to the same directory that you deployed the other JSP files that you created to override session properties. The reason you must manually copy this file is because the ConfigBase parameter does not affect this file.
- i. Compare your new HTML code to make sure that it is identical to the code in the file provided in Section 16.5.2, “Troubleshooting Scenario 2” on page 613. If not, make the necessary changes. If so, change the file extension from HTML to JSP and save it to the same directory where you keep the other JSP files that you have created to override session properties.

Important: This scenario assumes that you have already created a Web module using WebSphere Application Server. If you have not already created a Web module, repeat steps 9-19 in Section 16.4.1, “Steps to modify the session properties” on page 589.

4. Finally, point your browser to the form at `http://your_host/context_root/HODForm.html`, where `your_host` is the name of your local host and `context_root` is the name of the context root. Once the form appears and you enter and submit your host name, the browser redirects to the JSP file, and the session icon appears. Right click on the session icon and select Properties. In the Connection tab (Figure 16-11), check to make sure that the host name you specified in the form is located in the Destination Address field.

16.5.2 Troubleshooting Scenario 2

If your HTML override is not successful, take the following steps to help determine the cause:

1. Use a static value in your HTML file to see if your HTML overrides mechanism is working. For example, instead of coding the HTML file so that the host name comes from the HODForm.html (Step 2), replace the form lookup with a static host name value, as in the following example:

```
hod_AppletParams[4] = '<PARAM NAME="Host" VALUE="3270  
    Display=Static_host_name">');
```

Refer to [4](#) on page 615 to see where these lines are located in the HTML file.

Once you complete the steps to override the HTML parameter, open the session properties and make sure that the static host name is in the Destination Address field of the Connection tab.

2. If you want a JavaScript alert window to appear when you first access the JSP file that tells you if your program generated the proper syntax, change false to true in the following line:

```
var hod_DebugOn = true;
```

Refer to [3](#) on page 615 to see where this line is located in the HTML code.

3. Add the debug parameter DebugCode to your HTML file generated by the Deployment Wizard and set it to a value of 65535:

```
document.write('<PARAM NAME="DebugCode" VALUE="65535">');
```

This parameter sends useful debugging information to the Java Console, such as whether or not your Codebase, Configbase, and TargetedSessionList parameters and session properties were read correctly.

Refer to [3](#) on page 615 to see where this line is located in the HTML file.

4. Check your access log file for messages that contain the code '404'. This is the standard code for "Page Not Found." This access log file is generated by your Web server and is located in your Web server directory.

16.5.3 Completed HTML file after edits

The following code is the original HTML code generated from the Deployment Wizard plus the additions the administrator made to modify session properties dynamically. The lines added to the original file are displayed in bold. The callout boxes in the left margin show you the corresponding step from the previous section.

1

```
<%
HttpSession session = request.getSession(true);
String hostname = request.getParameter("form.hostname");

if (hostname!=null) {
    session.putValue("session.hostname", hostname);
}
%>
<HTML>
<!-- HOD WIZARD HTML -->
<HEAD>
<META content="text/html; charset=UTF-8">
<TITLE>example2</TITLE>
<!-- SUMMARY Begin -->
<!--
Configuration Model
    What configuration model would you like to use?
    -HTML-based model
Sessions created
    -3270 Display
    -5250 Display
Additional Options
    -Allow users to save session changes? = True
    -Cached = False
    -Java Type = detect
Disable Functions
Preload Options
    -5250 Sessions = True
    -Change Session Properties = True
    -3270 Sessions = True
Server Connection Options
Cache Options
Advanced Options
    Display
        -Standard Host On-Demand Client
        -Applet size = Autosize to browser
        -Maximum sessions = 26
    Other
        -Locale = Use the system Locale
        -Debug = False
        -HTML Template = Default
```

```

        Additional Parameters
        -None
    -->
<!-- SUMMARY End -->
</HEAD>
<SCRIPT LANGUAGE="JAVASCRIPT" SRC="CommonJars.js"></SCRIPT>
<SCRIPT LANGUAGE="JAVASCRIPT" SRC="HODJavaDetect.js"></SCRIPT>
<SCRIPT LANGUAGE="JAVASCRIPT" SRC="CommonParms.js"></SCRIPT>
<SCRIPT LANGUAGE="JAVASCRIPT">

//---- Start JavaScript variable declarations ----//
var hod_Locale = '';
var hod_AppName = '';
var hod_AppHgt = '80%';
var hod_AppWid = '80%';
var hod_CodeBase = '/hod/';
2   var hod_FinalFile = 'z_example2.jsp';
var hod_JavaType = 'detect';
var hod_Obplet = '';
var hod_jars =
'habasen.jar,hodbasen.jar,hodimg.jar,hacp.jar,hodsignn.jar,
    ha3270n.jar,hodcfgn.jar,ha5250n.jar';

var hod_URL = new String(window.location);
3   var hod_DebugOn = true;
var hod_SearchArg = window.location.search.substring(1);

var hod_AppletParams = new Array;
hod_AppletParams[0] = '<PARAM NAME="ParameterFile"

hod_AppletParams[0] = '<PARAM NAME="ParameterFile"
    VALUE="HODData\\example2\\params.txt">';
hod_AppletParams[1] = '<PARAM NAME="ShowDocument" VALUE="_parent">';
hod_AppletParams[2] = '<PARAM NAME="JavaScriptAPI" VALUE="false">';
hod_AppletParams[3] = '<PARAM NAME="PreloadComponentList"
    VALUE="HABASE;HODBASE;HODIMG;HACP;HAFNTIB;HAFNTAP;HA3270;HODCFG;HA5250">';

4   hod_AppletParams[4] = '<PARAM NAME="EnableHTMLOverrides" VALUE="true">';
5   hod_AppletParams[5] = '<PARAM NAME="TargetedSessionList"
    VALUE="3270 Display,5250 Display">';
6   hod_AppletParams[6] = '<PARAM NAME="Host" VALUE="3270
    Display=<%=session.getValue("session.hostname")%>">';
7   hod_AppletParams[7] = '<PARAM NAME="ConfigBase"
    Value="http://host_name/hod/">';
8   hod_AppletParams[8] = '<PARAM NAME="DebugCode" VALUE="65535">';

```

```
var lang = detectLanguage(hod_Locale);

function getHODMsg(msgNum) {
    return HODFrame.hodMsgs[msgNum];
}
//---- End JavaScript variable declarations ----//

function getHODFrame() {
    return HODFrame;
}

document.writeln('<FRAMESET cols="*,10" border=0 FRAMEBORDER="0">');
document.writeln('<FRAME    src="hoddetect_' + lang + '.html"');
name="HODFrame">');
document.writeln('</FRAMESET>');
</SCRIPT>
</HEAD>
</HTML>
```



Host On-Demand Portlets

In this chapter we cover the support added in Host On-Demand for IBM WebSphere Portal V4.1. This support allows Host On-Demand to run as a portlet within the Portal Server component of Websphere Portal.

17.1 Introduction to Portal Servers

The IBM redbook Websphere Portal 4.1.2 for Linux by IBM defines a portal like this:

What makes a portal a portal

A portal is an application or device that provides a personalized and adaptive interface for people to discover, track and interact with other relevant people, applications and content. I discovered a portal is very different from that simple HTML page with framesets, because it possesses these distinguishing features:

1. Personalization for end user's is the most critical feature. A portal must deliver a personal or community desktop to users by establishing unique looks, content, and application interfaces and operatively rendering them based on the user's role in their community or by actively tracking the user's individual usage, interests and behaviors.
2. Organization of the user's desktop to eliminate the information glut. Users want consolidated access to their important contacts, applications and content. The concept of stovepipe applications are a thing of the past. Organizations want easier control to design their desktop in a layout that suits them.
3. Resource division determines who sees what. Portals must have a membership services layer for user authentication, single-logon and credential mapping. Users demand the highest level of security, but the least amount of annoyance.
4. Tracking of activity provides users with a payback for using the portal. The more users use the portal, the more it becomes tailored to specific interests and affinities the user may develop. While this may sound threatening at first, users will have the ultimate control over what gets tracked.
5. Access and display of aggregated multiple heterogeneous data stores, including relational databases, multidimensional databases, document management systems, e-mail systems, Web servers, news feeds, and various file systems/servers (e.g., audio, video, image, and so on). It's extremely helpful for users to see their e-mail, next to news feeds, beside a list of online users who can help understand information while maintaining a single context.
6. Location of important people and things. A portal is based on the basic desire of users to easily find information and people by searching or navigation. There must be a means of passively or actively discovering the experts, communities and content in a relevant context. If developers succeed in incorporating all these capabilities into a single application they have built a

basic portal design that can be targeted at all types of audiences and applied against a broad range of content and tool types.

17.1.1 What is a portlet

As described in the redbook “IBM Websphere Portal 4.1.1 in a Linux Environment”:

Portlets are reusable components that provide access to enterprise applications, Web-based content, and other resources. For example, Web pages, Web services, legacy host applications can be access through portlets

Any particular portlet is developed, deployed, managed, and displayed independent of other portlets. Administrators and end users create personalized portal pages by choosing and arranging portlets, resulting in web pages.

17.1.2 Information on IBM Portal Servers

IBM provided a tremendous amount of documentation on configuring IBM Portal servers.

Search the IBM redbook site for Portal.

- ▶ <http://redbooks.ibm.com>
- ▶ Information on Portal server 2.1
- ▶ <http://www.redbooks.ibm.com/redpapers/pdfs/redp0191.pdf>

Search the IBM web site for Portal Products:

- ▶ <http://www.ibm.com/software/webservers/portal/>

17.2 How HOD portlet support works with Websphere Portal Server 4.1

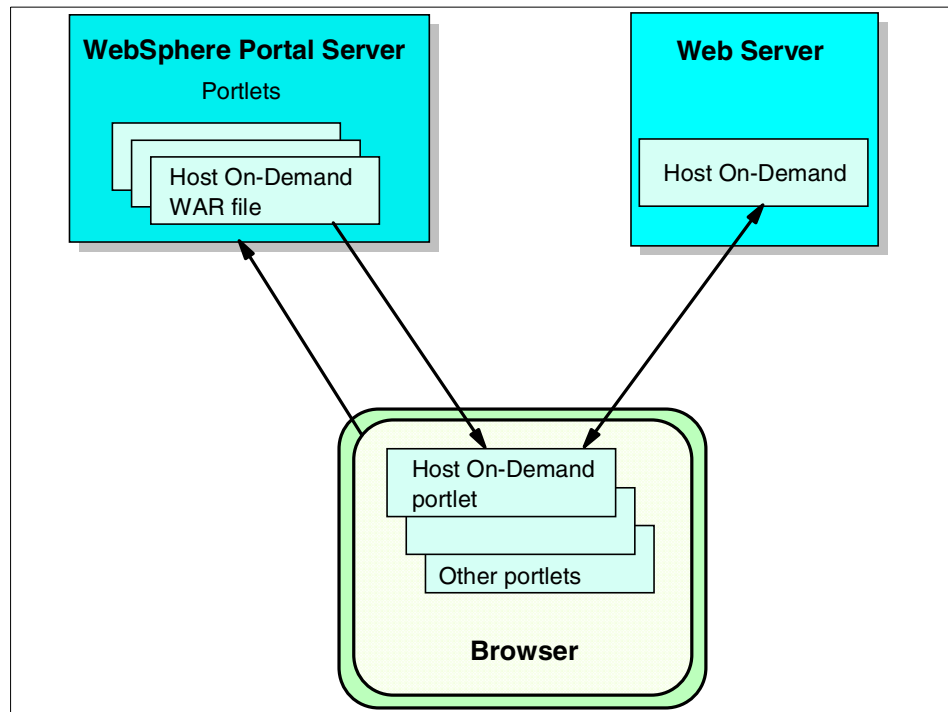


Figure 17-1 HOD with Portal Server

- ▶ A user logs into the portal through a browser and is authenticated by a user ID and password.
- ▶ If the user has configured a Host On-Demand portlet, Host On-Demand starts. This gives the user full Host On-Demand functionality within the portlet window, including being able to start sessions and perform other Host On-Demand tasks.
- ▶ A user logs into the portal through a browser and is authenticated by a user ID and password. The user may also have to provide a userid and password to login to HOD. It is possible to configure HOD to bypass or not require a HOD userid.

All the features of HOD Deployment wizard are supported. See Chapter 14, “Deployment Wizard” on page 529.

1. The sessions launch in the same window by selecting the Advanced tab in Session Properties and setting Start in a Separate Window to No.

2. **Starting the session automatically.** By default, Host On-Demand sessions will not start until the user selects the icon to start. If you wish to have the session start automatically, select the Advanced tab in Session Properties and set Start Automatically to Yes.
3. **Setting the portlet's access control in Portal Server.** The Host On-Demand portlet does not have any fields that a user can edit using the portlet interface. Therefore, when you import the portlet into Portal Server, you should set the access control to be viewable, but not editable.

17.3 Scenarios for When and where to use HOD portlet

- ▶ for B2B
 - The portal server and Host On-Demand provide a mechanism for implementing direct access to your host applications on your B2B portal
- ▶ for B2E
 - Allowing your employees and others in your internet or intranet access to existing legacy host applications
- ▶ for B2C
 - A portal provides one easy to use, attractive interface to your corporate business applications from any location in the world. The Host On-Demand portlet can provide a eye opening first impression of your company to your customers.

Beginning with HOD V7 the generation of Portlet files becomes simple using the additional functions added to the Deployment Wizard. The basic use of the Wizard is explained in Chapter 14, “Deployment Wizard” on page 529. The specific options to generate a Portlet file occur on the last panel of the Wizard when you are presented the options for the type of file the Deployment Wizard should generate, Figure 17-2 on page 622.

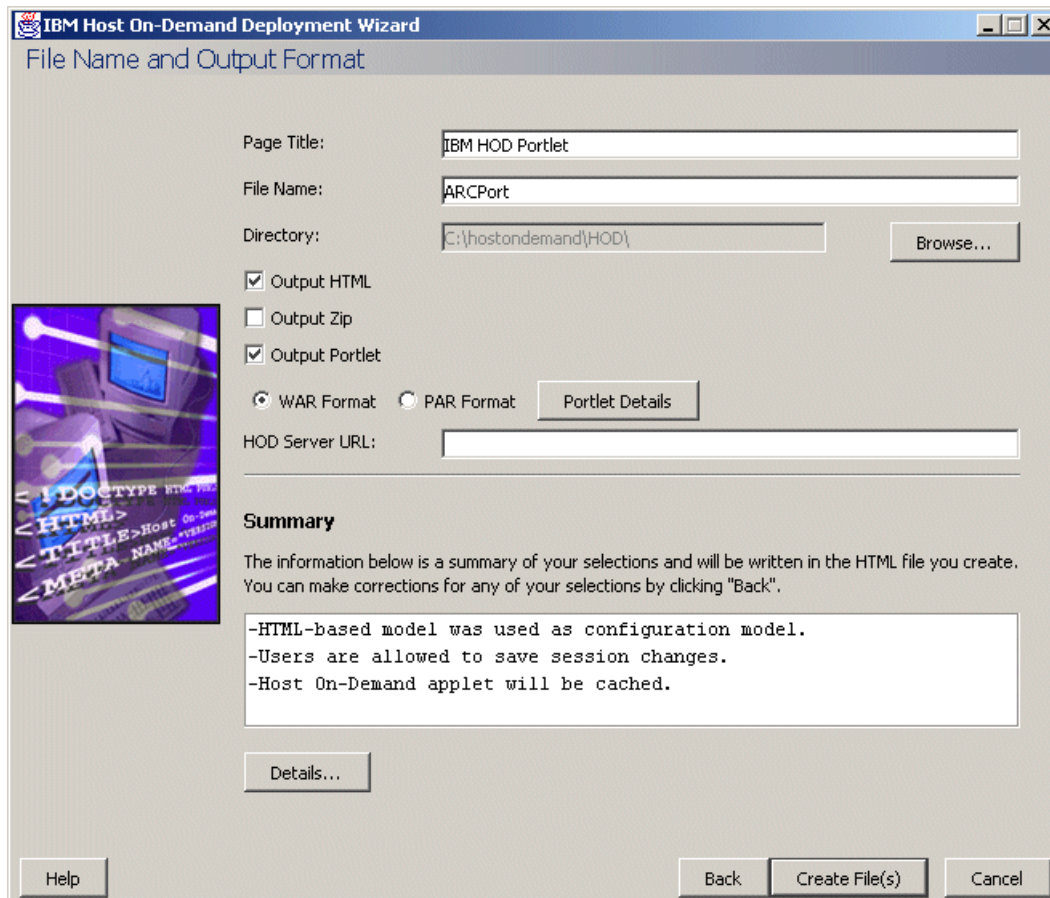


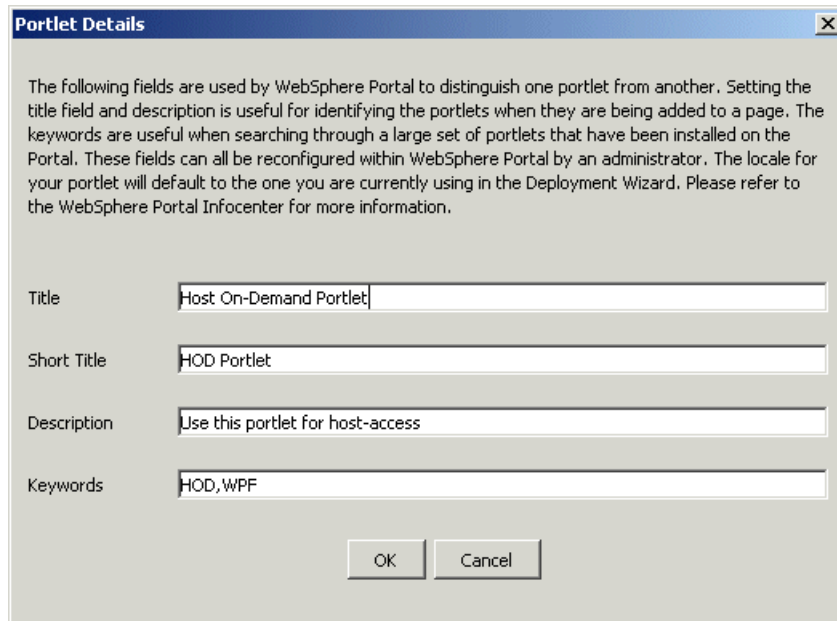
Figure 17-2 Deployment Wizard Portlet Definition

Select the type of portlet file you require:

- | | |
|-----|--|
| WAR | Web Archive file for Websphere Portlet Server 4.1 and later. |
| PAR | Portlet Archive File for Websphere Portlet Server 2.1 or earlier |

Note: The type of portlet file you generate is determined by the level of WebSphere Portal Server you are running. The different file types are NOT interchangeable

Choose the Portlet details button in order to set description fields for the portlet.



Portlet Details

The following fields are used by WebSphere Portal to distinguish one portlet from another. Setting the title field and description is useful for identifying the portlets when they are being added to a page. The keywords are useful when searching through a large set of portlets that have been installed on the Portal. These fields can all be reconfigured within WebSphere Portal by an administrator. The locale for your portlet will default to the one you are currently using in the Deployment Wizard. Please refer to the WebSphere Portal Infocenter for more information.

Title: Host On-Demand Portlet

Short Title: HOD Portlet

Description: Use this portlet for host-access

Keywords: HOD,WPF

OK Cancel

Figure 17-3 Portlet Definition Fields

17.4 Installing HOD portlet in Websphere Portal Server 4.1

Once you have generated the appropriate portal file, you are ready to install this file in portal server.

- Navigate to the web page you have setup to manage the portal server and logon as the Portal Administrator

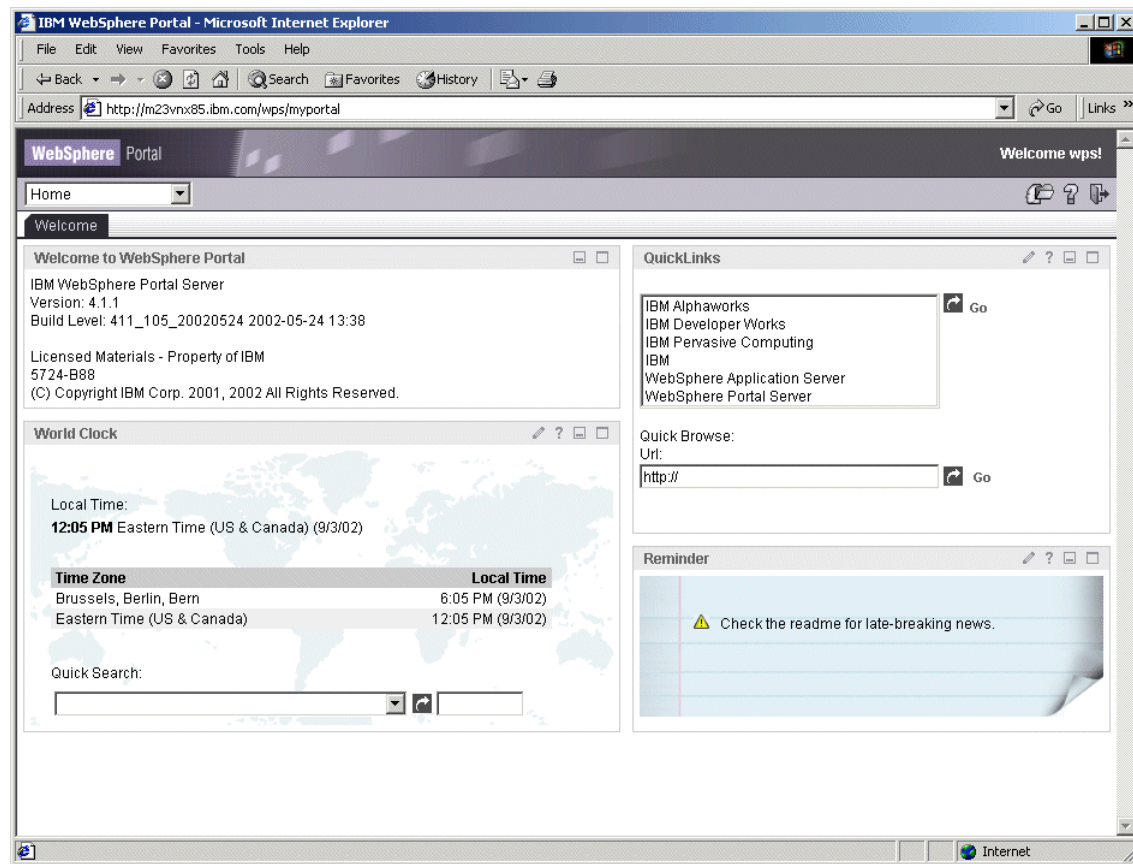


Figure 17-4 Portal Welcome screen after login

- ▶ Once you log in, select **Install Portlets** in the page pull-down on left hand side of page
- ▶ Browse to the location of the HOD portlet file

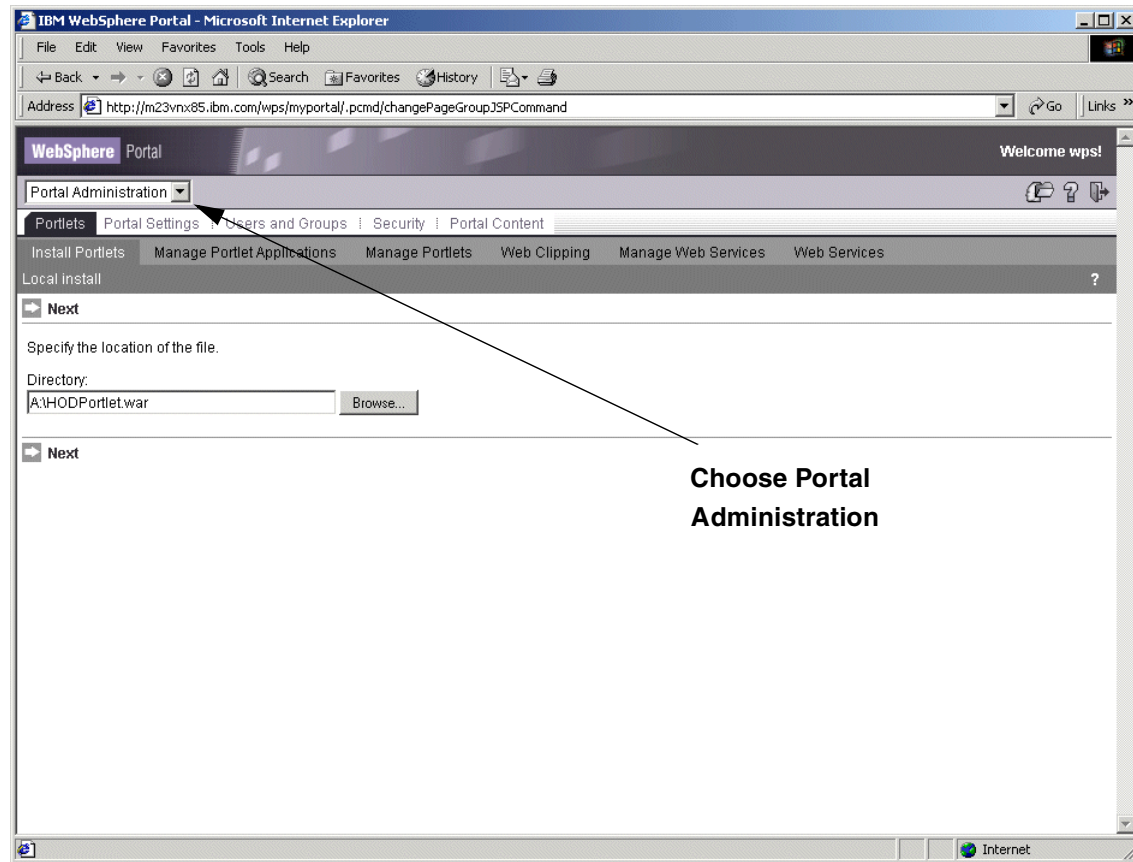


Figure 17-5 Specify Location of HOD Portal file

- Install the Portlet file, see Figure 17-6 on page 626

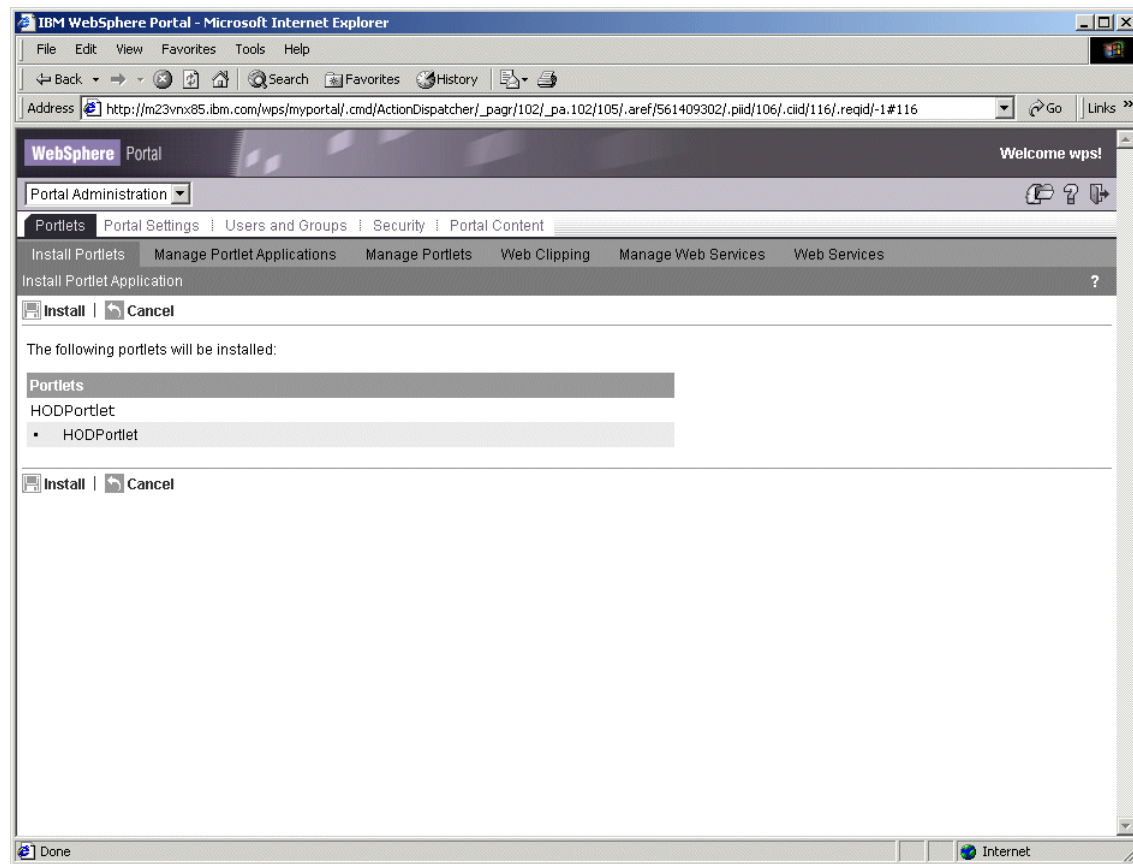


Figure 17-6 List of Portlets to be Installed

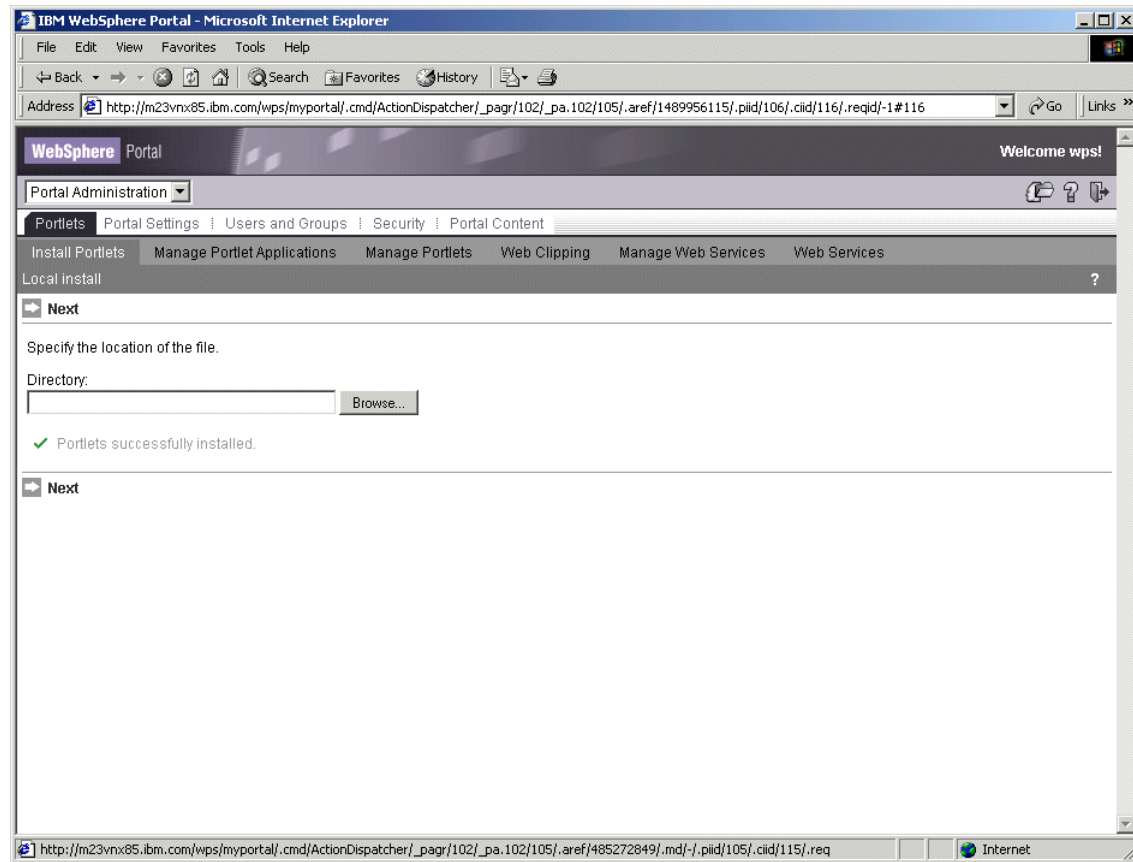


Figure 17-7 Portlet Install OK

- ▶ Once the portlet is installed you must check that it is active and ready to be deployed for use within the Portal server.
 - Access the Portal Administration panel and selecting Manage Portlets, see Figure 17-8 on page 628
 - Verify in the displayed list of portlets that the HOD portlet you installed is active and if not, activate it

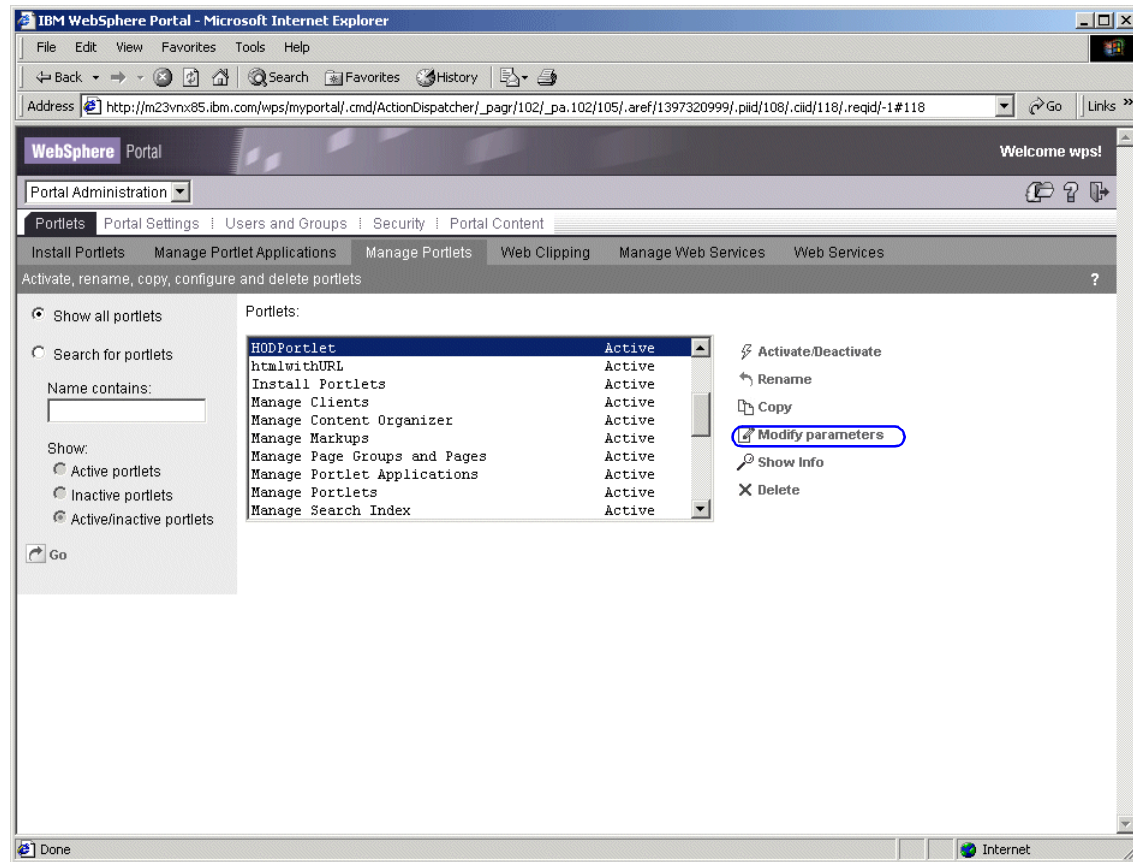


Figure 17-8 Manage Portlet Parameters

Using the same Manage Portlet page you can select Modify Parameters to update any parms originally set in the HOD portlet. This can be useful when the address for the HOD server changes, and you must modify the value for `hodCodeBase` as shown in Figure 17-9 on page 629

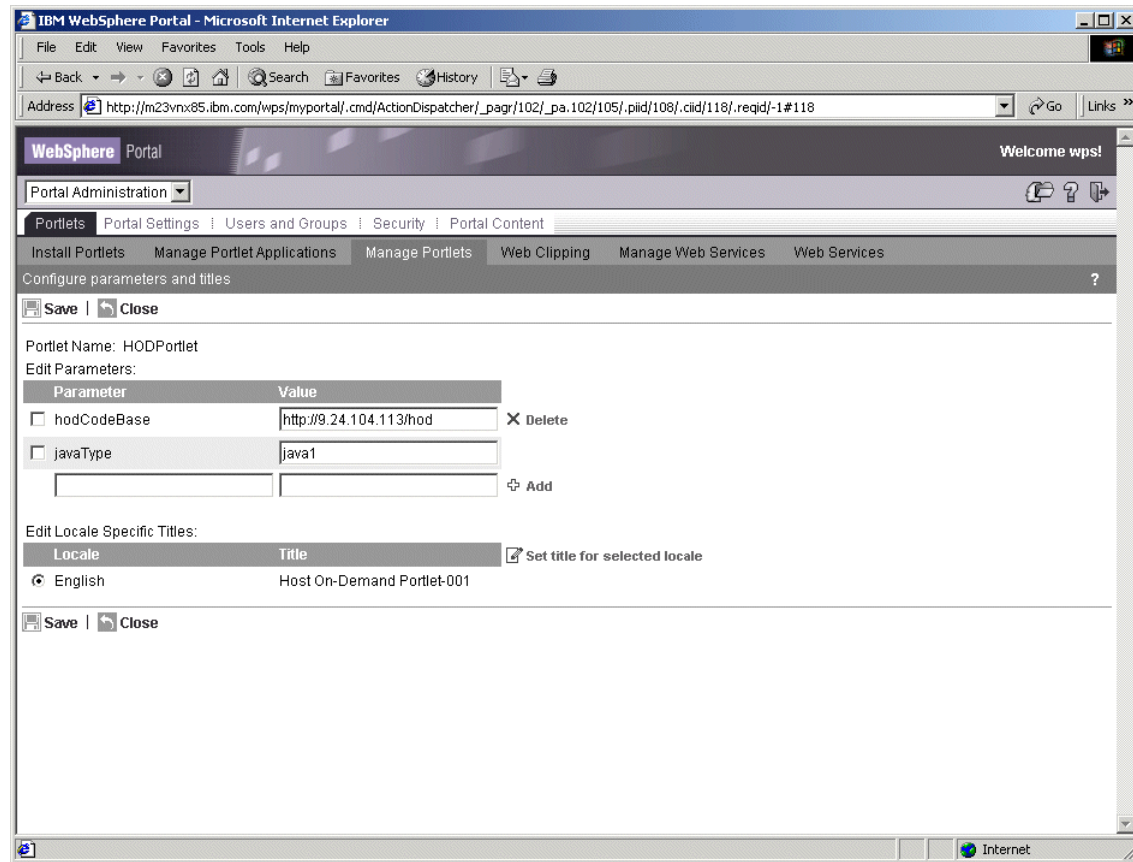


Figure 17-9 Manage Portlet Properties with Portal Server

17.4.1 Deploying HOD portlet to Portal page

Once you have installed the HOD portlet and made it active, you will need to setup individual users pages in the Portal server to display the HOD portlet. Go back to the Portal server administrator console and select *Work with Pages*

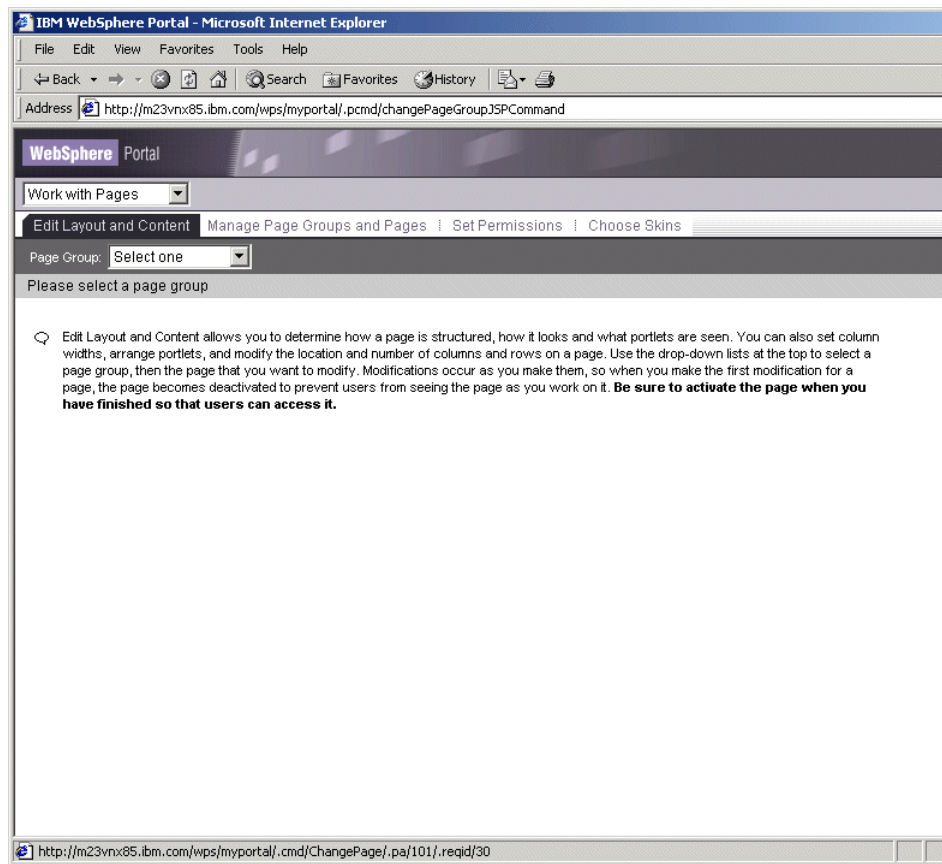


Figure 17-10 Portal Server, working with Portal Pages

Once you are here, you select Manage Page PortalManage Groups and Pages to add the HOD portlet to users portal page

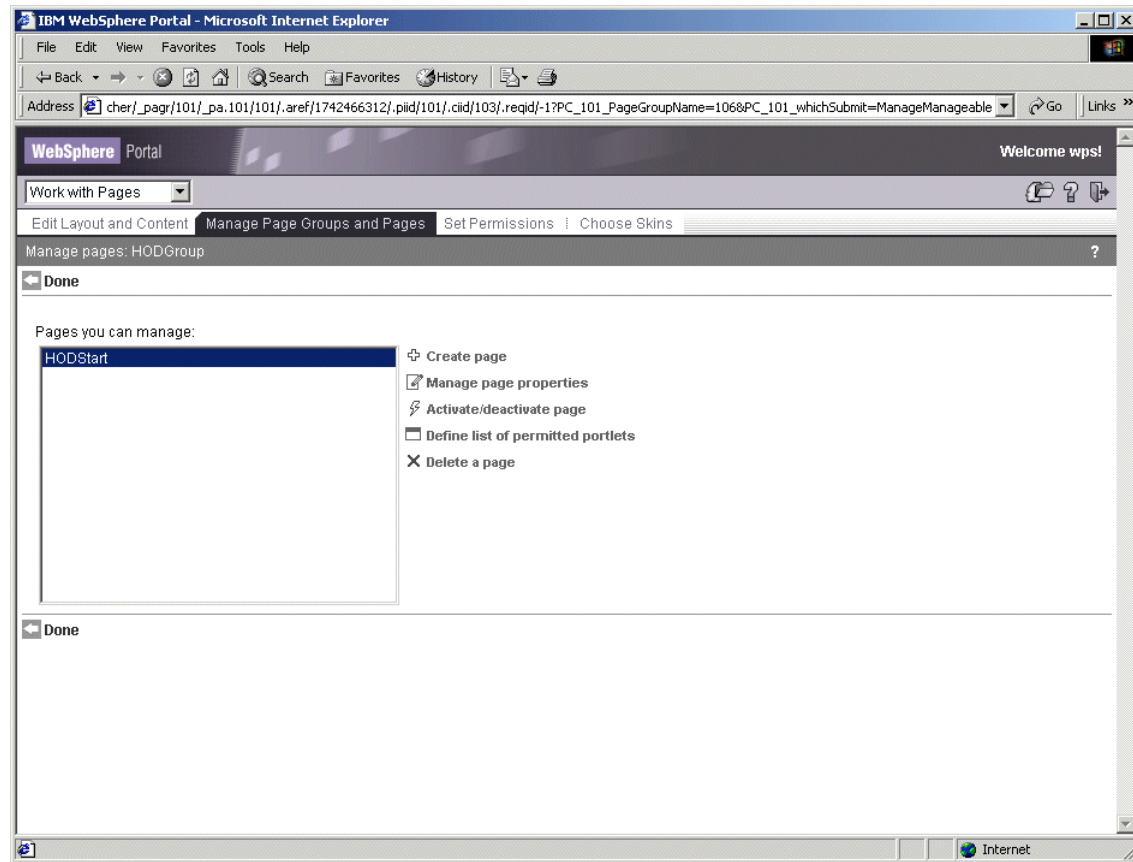


Figure 17-11 Manage Portal pages

Once the portal is deployed on a page the users may see something that looks like this:

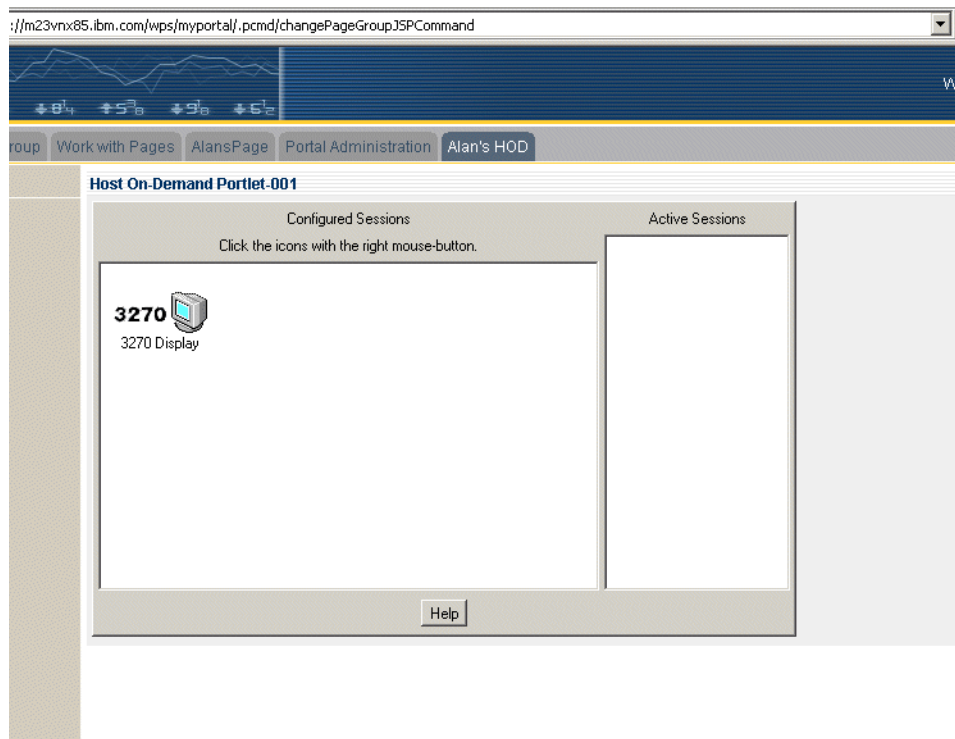


Figure 17-12 Initial HOD Portlet Screen when configured as HTML Model in Portal Server page

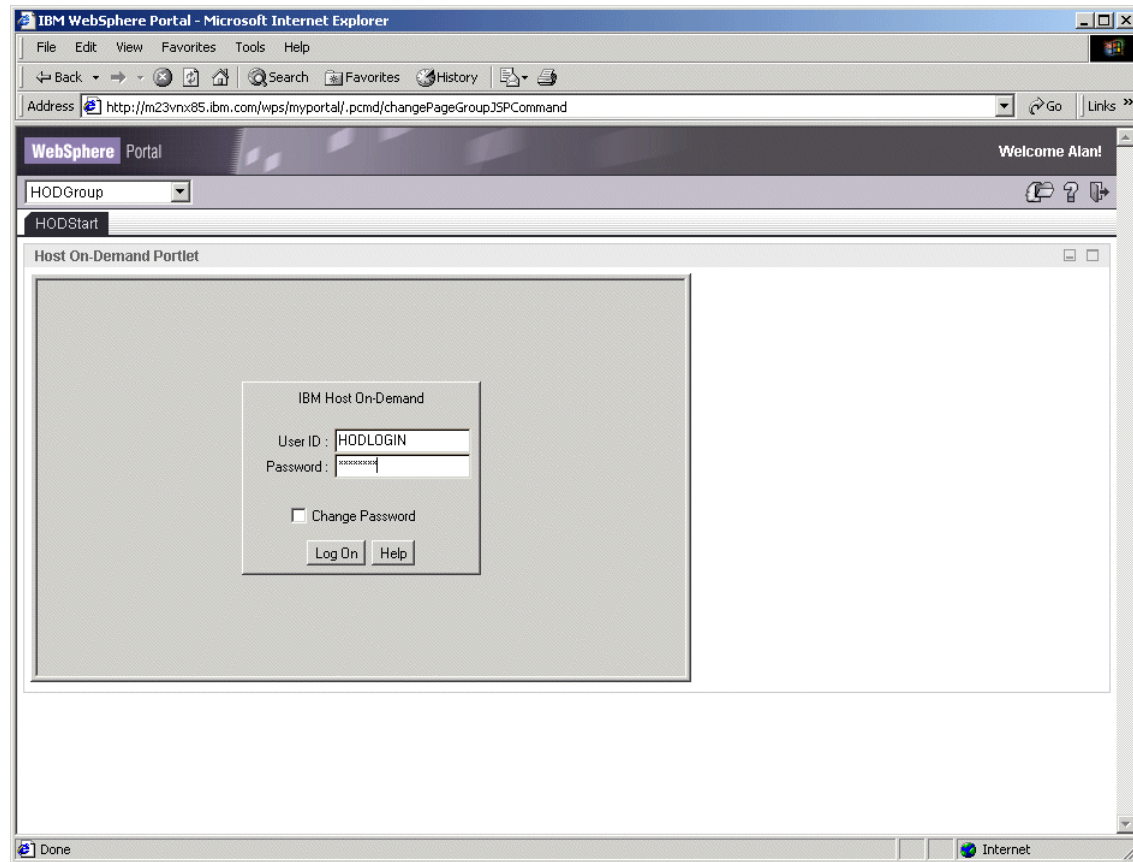


Figure 17-13 Initial HOD Login screen when Portlet configured to use Config Server

17.4.2 Special Considerations when using HOD portlet

When using Host On-Demand with Portal Server, The HOD administrator may want to consider the following issues:

1. **Setting the Host On-Demand applet size for the client.** If you would like an applet size that is different from the available options in the Deployment Wizard, you can modify the portlet to specify pixel width and height. To do this, you will first need to extract the portlet and locate the file called `WpsHODFinal.jsp`. In this file, locate the two lines beginning with `var hod_AppHgt` and `var hod_AppWid`. These are JavaScript variables defining the applet dimensions. Edit the quantities assigned to each of these variables with the dimensions you desire. Save the file, repackage the portlet, and install the portlet in your portal.

- Here is an example of modifying the portlet size.
 - First Open the WAR file with a zip utility or the JAR utility packaged with the JRE. See Chapter 17-14, “Unpack Host On-Demand WAR file” on page 634
 - Edit the file WpsHODFinal.jsp and change the values for these 2 variables to fit your environment, see Chapter 17-15, “Modify WpsHODFinal.jsp” on page 635. The value of these settings is


```

          hod_AppHgt = '340';
          hod_AppWid = '550';
          
```

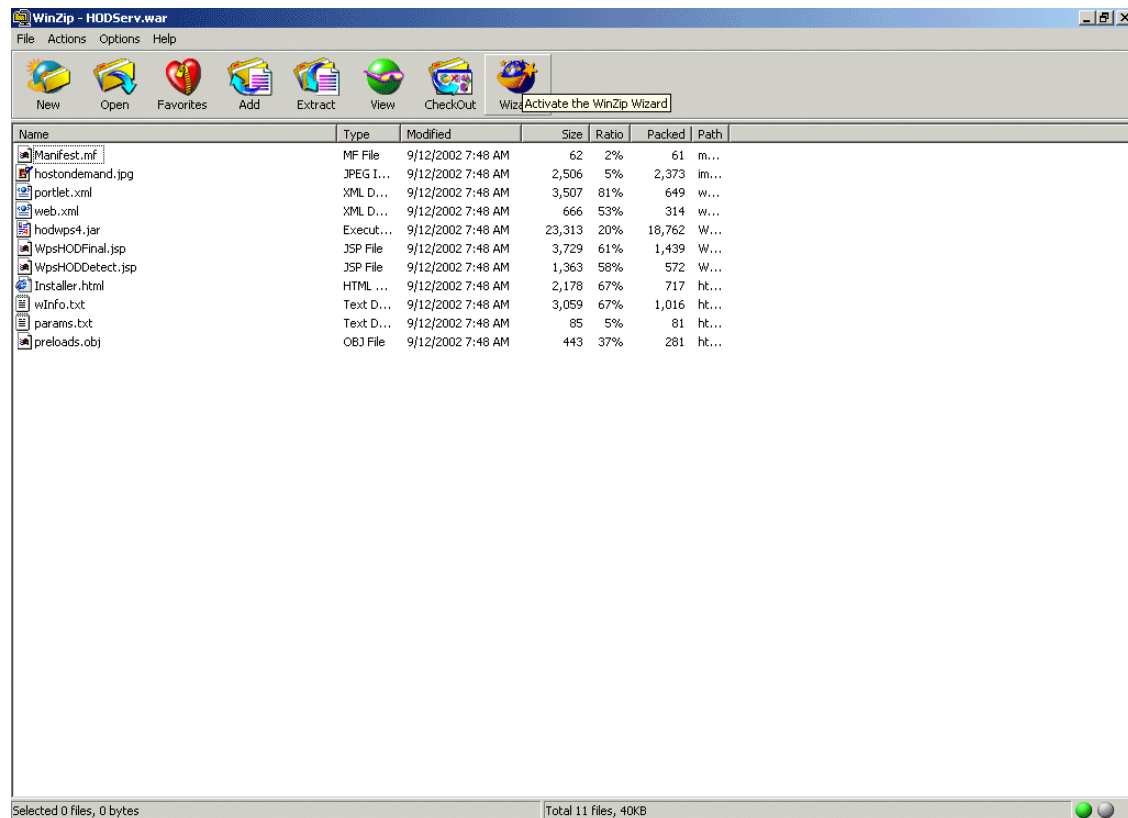
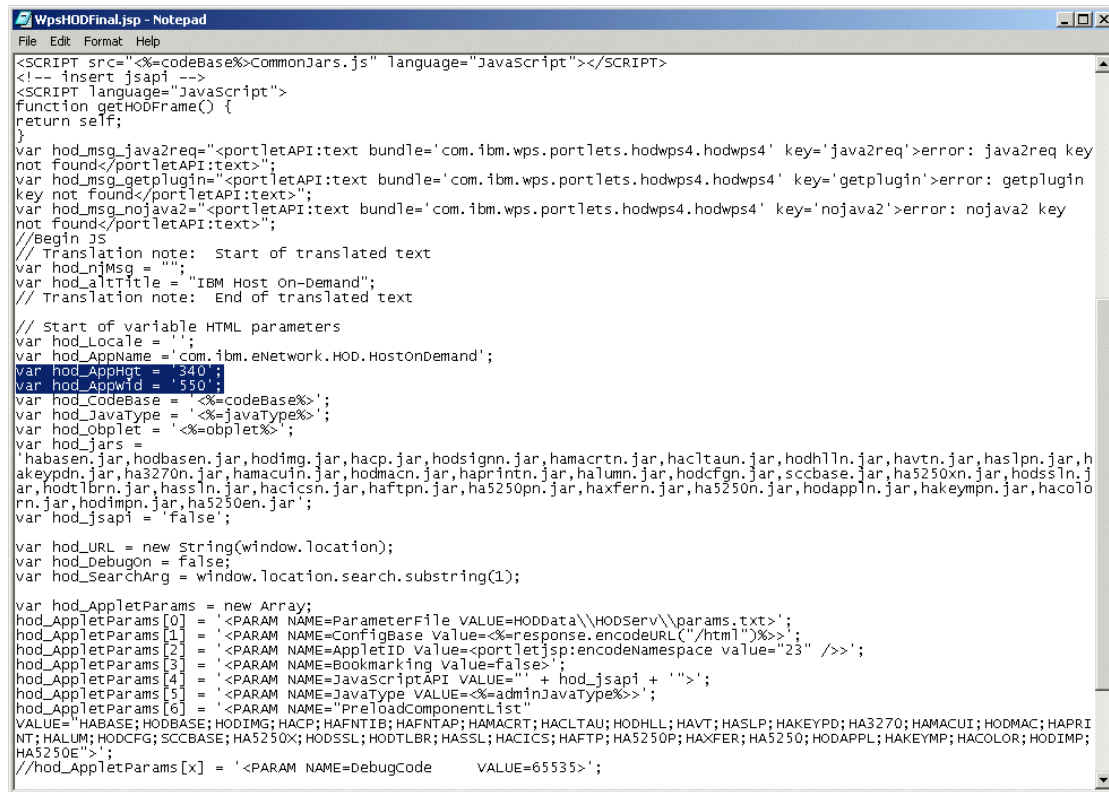


Figure 17-14 Unpack Host On-Demand WAR file



```

WpsHODFinal.jsp - Notepad
File Edit Format Help
<SCRIPT src="%codeBase%CommonJars.js" language="JavaScript"></SCRIPT>
<!-- insert jsapi -->
<SCRIPT language="JavaScript">
function getHODFrame() {
return self;
}
var hod_msg_java2req="<portletAPI:text bundle='com.ibm.wps.portlets.hodwps4.hodwps4' key='java2req'>error: java2req key
not found</portletAPI:text>";
var hod_msg_getplugin="<portletAPI:text bundle='com.ibm.wps.portlets.hodwps4.hodwps4' key='getplugin'>error: getplugin
key not found</portletAPI:text>";
var hod_msg_nojava2="<portletAPI:text bundle='com.ibm.wps.portlets.hodwps4.hodwps4' key='nojava2'>error: nojava2 key
not found</portletAPI:text>";
//Begin js
// Translation note: Start of translated text
var hod_njMsg = "";
var hod_altTitle = "IBM Host On-Demand";
// Translation note: End of translated text
// Start of variable HTML parameters
var hod_Locale = '';
var hod_AppName = 'com.ibm.eNetwork.HOD.HostOnDemand';
var hod_AppMgn = 540;
var hod_AppWid = 550;
var hod_CodeBase = '%codeBase%';
var hod_JavaType = '%javaType%';
var hod_Objlet = '%objlet%';
var hod_Jars =
'habasen.jar,hodbasen.jar,hodimg.jar,hacp.jar,hodsignn.jar,hamacrtn.jar,hac1taun.jar,hodh1ln.jar,havtn.jar,has1pn.jar,h
akeypdn.jar,ha3270n.jar,hamacuin.jar,hodmacn.jar,haprintn.jar,halumn.jar,hodcfgn.jar,scbase.jar,ha5250xn.jar,hodssl.n.j
ar,hodt1brn.jar,hass1n.jar,hac1csn.jar,haftpn.jar,ha5250pn.jar,haxfern.jar,ha5250n.jar,hodappln.jar,hakeympn.jar,hacolo
rn.jar,hodimpn.jar,ha5250en.jar';
var hod_jsapi = 'false';

var hod_URL = new String(window.location);
var hod_Debugon = false;
var hod_SearchArg = window.location.search.substring(1);

var hod_AppletParams = new Array;
hod_AppletParams[0] = '<PARAM NAME=ParameterFile VALUE=HODdata\\HODserv\\params.txt>';
hod_AppletParams[1] = '<PARAM NAME=ConfigBase Value=%response.encodeURL("/html")%>';
hod_AppletParams[2] = '<PARAM NAME=AppletID Value=<portlet.jsp:encodeNamespace value="23" />>';
hod_AppletParams[3] = '<PARAM NAME=Bookmarking Value=false>';
hod_AppletParams[4] = '<PARAM NAME=JavaScriptAPI VALUE=" + hod_jsapi + ">';
hod_AppletParams[5] = '<PARAM NAME=JavaType VALUE=%adminJavaType%>';
hod_AppletParams[6] = '<PARAM NAME=PreloadComponentList
VALUE=HABASE;HODBASE;HODIMG;HACP;HAFNTIB;HAFNTAP;HAMACRT;HAC1TAU;HODHLL;HAVT;HASLP;HAKEYPD;HA3270;HAMACUI;HODMAC;HAPRI
NT;HALUM;HODCFG;SCCBASE;HA5250X;HODSSL;HODT1BR;HASSL;HAC1CS;HAFTP;HA5250P;HAXFER;HA5250;HODAPPL;HAKEYMP;HACOLOR;HODIMP;
HA5250E>';
//hod_AppletParams[x] = '<PARAM NAME=DebugCode VALUE=65535>';

```

Figure 17-15 Modify WpsHODFinal.jsp

2. **Host On-Demand sessions when the user logs out of Portal Server-Forcing Session inactivity timeout.** Host On-Demand runs as an applet on the user's machine and therefore does not know when the user logs out of Portal Server. If the session is running in a separate window (default), the Host On-Demand session will continue until the user either closes the session or closes the browser. If the Host On-Demand session is running embedded in the Portal Server window and the user logs out of Portal Server, the session may appear to have ended, although the connection will remain until the browser window is closed. We strongly recommend that users close their browser window at the time they log out of Portal Server. In addition, you may wish to configure a session inactivity timeout for your sessions. see Figure 17-16.

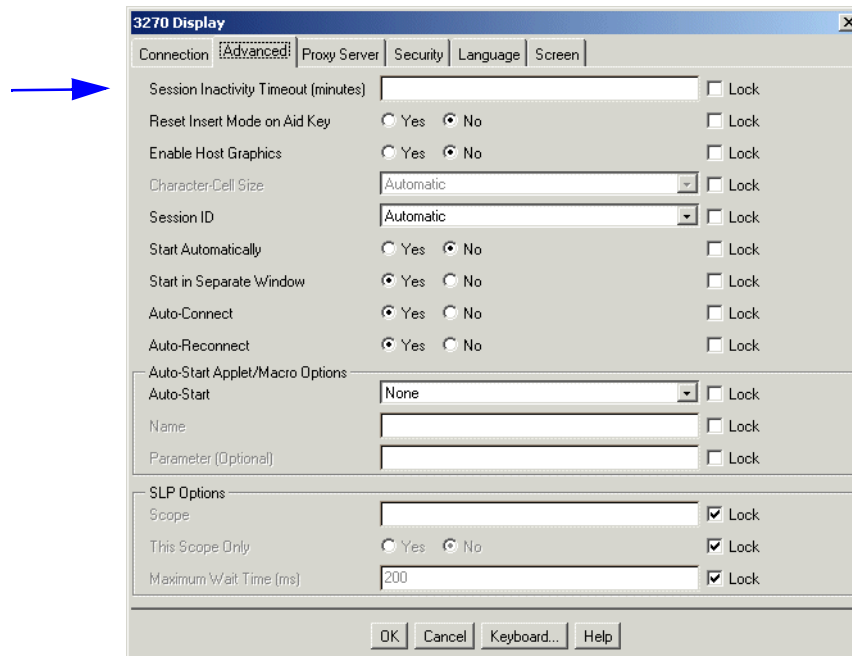


Figure 17-16 Setting Session Inactivity Timeout

3. **Installing Websphere Portal and Host On-Demand on different servers.** If you install Websphere Portal and Host On-Demand on different servers, certain browsers, such as Netscape 6, may give you a security violation when accessing the Host On-Demand portlet. The problem occurs because some aspects of Host On-Demand functionality rely heavily on the interaction between Java (from the Host On-Demand server) and JavaScript (from Websphere Portal), and some browsers will not allow the interaction simply because they come from different servers. One solution is to use proxying to make it appear to the browser that Websphere Portal and Host On-Demand are on the same server. The online Help describes how to setup proxying with the Apache/IBM HTTP server
4. **Caching vs. no-Caching.** The default setting in the Deployment Wizard is to cache Host On-Demand on each user's machine. Many customers like this option with Host On-Demand because it effectively installs all necessary code

on the user's machine and does not require network loads each time the user accesses the HTML file or portlet. However the caching behavior may not be familiar to many Portal Server users, and you may elect to reject the caching option. Caching also requires the user to restart the web Browser, forcing the user to disconnect from the Portal

5. **Choosing the Host ON-Demand Server model.** The model you choose for your portlet (Configuration-Server, HTML, or Combined) will reflect where your sessions are configured and will determine how user changes are stored. Although Host On-Demand treats portlets the same as HTML files, consider the following characteristics as you decide how to configure your portlet:
 - a. **HTML model:** This model has no dependency on the Host On-Demand Configuration Server. If users are allowed to make updates, their changes will be stored on their local machines. These user changes will not be available if the user roams to a different machine. See Chapter 13, "Deployment strategies" on page 513.
 - b. **Configuration-Server model:** This model requires user access to the Host On-Demand Configuration Server. It allows your users to roam from one machine to another and still see any session modifications they may have made. For more info on this topic refer to Chapter 13, "Deployment strategies" on page 513.
 - i. Using this model requires users to login to Host On-Demand with a userid and password. If you are considering forcing the use of a userid and password to restrict access to HOD, rather than to associate specific user preferences with a specific HOD userID, then It may be worth thinking if the userid and password capabilities of the Portal server can suffice, and this may allow you to consider using the HTML model.
 - c. **Combined model:** This model requires users to have access to the Host On-Demand Configuration Server in order to obtain the initial session configurations. Any user updates will be saved to the user's local machine and will not be available on a different machine if the user roams. See Chapter 13, "Deployment strategies" on page 513
 - d. **Defining embedded sessions.** By default, Host On-Demand sessions are configured to launch in a separate browser window. You can choose to have xxxxx

17.4.3 Migrating to HOD V7 portlet from previous HOD Versions

There are two actions that are necessary when migrating from a previous HOD portlet.

- ▶ The Portlet Administrator must install the new portlet and make it active for all users that require it.
- ▶ The HOD administrator needs to be aware of the general client upgrade concerns. See Chapter 5, “Clients” on page 163 for more information on this topic.

17.5 Potential Portlet problems

- ▶ When configuring the value for HOD SERVER URL in the deployment wizard portlet configuration screens, you should NOT put in an address that includes an HTML file.
 - Do not enter **//server/hod/hodmain.html** This is wrong.
This is correct **//server/hod** The trailing / is not required.
You will have errors if you code this parameter incorrectly. The errors will range from Scripting errors to errors indicating portlet not running. To fix this you need to modify the hodCodeBase parameter of the portlet

For General information on debugging Websphere Portlet server try these IBM web sites.

WAS 2.1 Trouble Shooting Information

InfoCenter Trouble Shooting Information:

<http://www.ibm.com/software/webservers/portal/library/InfoCenter/wps/trouble.html>

WAS 4.0 Trouble Shooting Information

- ▶ Trouble Shooting”
 - <http://publib.boulder.ibm.com/pvc/wp/current/ena/en/InfoCenter/wps/trouble.html>

- ▶ Release Notes:

http://publib.boulder.ibm.com/pvc/wp/current/exp/en/InfoCenter/wps/release_notes.html

- ▶ Portal Library Pages:

<http://www.ibm.com/software/webservers/portal/library.html>

17.5.1 Location of Portal log Files

Table 17-1 Location of Portal Server Log Files

WAS Version	Location of Log Files
WAS Portal 2.1	<Portal Home>/app/web/WEB-INF/log
WAS Portal 4.1	<Portal Home>/log



Session Manager APIs

The Host On-Demand Session Manager offers application programming interfaces (APIs) that you can use to embed host sessions in your company's Web infrastructure. Unlike the APIs included in the Host Access Toolkit, Session Manager APIs are JavaScript-based and do not require any Java programming.

Session Manager APIs are available in four different sets: Session Manager APIs, Presentation Space APIs, Host On-Demand Function APIs, and Error Reporting APIs. These APIs have many functions, including starting, stopping, and displaying either one or multiple sessions, starting macros, opening a session to the host, sending a string of text to the presentation space, retrieving text plane information, setting new string values, sending data stream function keys back to the host, and returning error messages. After an interaction is complete, the JavaScript code can switch to other tasks or simply close the session. The entire operation can be done without ever showing host screens.

In this chapter, we describe the four sets of Session Manager APIs in more detail, provide information about which components support the JavaScript API environment, and then present a real-life scenario with step-by-step instructions that show you how to embed host sessions in your own Web infrastructure. Finally, we introduce you to working demonstration code available on the redbooks Web site that you can use as an example of how to use JavaScript APIs.

For more in-depth information about each API, including the method names, parameters, and return values, refer to the *Session Manager API Reference* in the Host On-Demand InfoCenter, accessed by clicking **Start > Programs > IBM Host On-Demand > InfoCenter**.

18.1 The four types of JavaScript-based APIs

Session Manager APIs are divided into four sets: Session Manager APIs, Presentation Space APIs, Host On-Demand Function APIs, and Error Reporting APIs. This section describes each set in more detail.

18.1.1 Session Manager APIs

Session Manager APIs allow you to manipulate host sessions. Specifically, you can do the following:

- ▶ Start the specified session
- ▶ Stop the currently selected session, a specified session, or all active sessions
- ▶ Connect the currently selected or specified session
- ▶ Disconnect the currently selected or specified session
- ▶ Display the currently selected or specified session in a frame

In addition, get methods are available that return a string of all active sessions as well as the session ID of the last session started using the startSession API. Session Manager APIs allow you to manage multiple instances of a session as well as multiple sessions.

18.1.2 Presentation Space APIs

Presentation space APIs allow you to interact with host sessions within the presentation space. The presentation space is a virtual screen that contains all the characters and attributes that would be seen on a traditional emulator screen. Specifically, you can do the following:

- ▶ Send a string of text characters and keystrokes to the presentation space
- ▶ Send a given text string and keystrokes to the specified session either at the current or a specified cursor position.
- ▶ Send a string to the presentation space
- ▶ Send a string to the presentation space either at the current or a specified cursor position
- ▶ Retrieve text plane information from the presentation space

- ▶ Retrieve the data from the specified session at the beginning of the presentation space
- ▶ Retrieve the data from the specified session at the beginning of the specified position or coordinates until the specified number of plane positions have been copied
- ▶ Return the number of characters copied into the last getString call's returned String object
- ▶ Check whether the currently selected session is ready for interaction, such as sending keystrokes or calling other API methods
- ▶ Reset the currently selected or specified session's locked keyboard
- ▶ Check to see if the currently selected or specified session is ready to communicate with the host
- ▶ Provide screen recognition and OIA (Operator Information Area)-uninhibited functions

Presentation space APIs support multiple instances of a session as well as multiple sessions.

18.1.3 Host On-Demand Function APIs

Host On-Demand Function APIs include running a macro in either a currently selected or a specified session. They allow users to start a macro, such as one that automatically leads to a login screen. The macro must have been predefined in the session.

18.1.4 Error Reporting APIs

The Error Reporting API returns a saved error message from the last API call. If there was an exception thrown by Host On-Demand or Java as a result of one of the Session Manager, Presentation Space, or Host On-Demand Function APIs, then a getErrorMessage method returns the exception message.

18.2 Host On-Demand components that support Session Manager APIs

The Host On-Demand Deployment Wizard, cached client, and tracing function support the JavaScript API environment.

18.2.1 Deployment Wizard

Use the Deployment Wizard to embed Host On-Demand sessions in your Web environment. Although all three configuration models support Session Manager APIs, we recommend the HTML-based model for Web integration. If you use the configuration server-based model or the combined model, client users will have to log in using the login screen (unless the username and password are specified in the HTML), and they will have to log off using the logoff button on the Host On-Demand desktop. This is because the JavaScript APIs do not provide login and logoff functions.

On the Other tab of the Deployment Wizard's Advanced Options window, when you select **Enable Session Manager JavaScript API** as shown in Figure 18-5 on page 650, the following code is added to your HTML file:

```
<SCRIPT LANGUAGE="JavaScript" SRC="HODJSAPI.js"></SCRIPT>
```

HODJSAPI.js provides the JavaScript interface to all the Session Manager APIs and can be found in Host On-Demand's default publish directory `hostondemand/HOD`.

The Deployment Wizard now provides a user parameter called `HideHODDesktop` that hides the Host On-Demand desktop and session tabs once an embedded session starts. You can add this parameter on the Advanced Options' Additional Parameters tab in the Deployment Wizard. See Figure 18-6 on page 651 for additional details.

Session Manager APIs do not support bookmarking of sessions due to the embedded nature of the HTML files.

18.2.2 Cached Client

The Host On-Demand cached client also supports Session Manager APIs. In the Deployment Wizard, when you select both to cache the Host On-Demand applet and to enable Session Manager APIs, the cached client's loader applet loads the `com.ibm.eNetwork.HOD.JSHostOnDemand` applet instead of the `com.ibm.eNetwork.HOD.HostOnDemand` applet.

18.2.3 Tracing

The `JSHostOnDemand` applet provides parameters entry and exit level of tracing for all Session Manager APIs. It also traces all exceptions. In addition, you can use the ECLPS component of the Host Access Class Library to trace Presentation Space APIs. Existing Host On-Demand components can trace the Session Manager and Host On-Demand Functions APIs and debug problems.

A new HOD.JSSessionManager option_tag traces Session Manager APIs using the TraceOptions HTML parameter. You can use this tag by doing the following in your Deployment Wizard HTML file:

- ▶ select to include problem determination components on the Advanced Options > Other window, and
- ▶ add TraceOptions in the Name field and SaveLocation=Local, OutputFile=c:\HODTrace\trace.tlg, HOD.SessionManagerAPI=3 in the Value field on the Advanced Options > Additional Parameters window. (The output file can be any valid file name.)

18.3 Example customer scenario

A real estate company's financial consultants access customer data through Host On-Demand sessions. The company wants to be able to embed these host sessions into its already-existing Web infrastructure, allowing its consultants to interact with the sessions and use Host On-Demand's built-in functions within the company's own personalized Web interface.

The firm is very cost-conscious and does not want to outsource Java programmers to code Java-based APIs. Instead, they prefer to use JavaScript-based APIs, which can be maintained by their own in-house Web developers.

The firm decides to take advantage of Host On-Demand's Session Manager APIs, which are JavaScript-based and are separate from the Java-based APIs provided with the Host Access Toolkit. The company's Web development team designs some Web pages that allow the company's consultants to access host sessions while maintaining the company's own look and feel. Their overall goals include the following:

- ▶ to allow single sign-on to both the company's Web server as well as host applications
- ▶ to print the contents of host sessions
- ▶ to enter values into the host sessions

Figure 18-1 shows the Web page that the company's Web development team designs for the consultants to use as a home page when they access the company's personal HTML files as well as their Host On-Demand sessions. Although the Host On-Demand sessions are active in the Web site and the user can interact with them, the sessions are hidden in the user's browser.

**Jones Real Estate
Brokerage Firm**

Welcome Session A Login Print
Contact Info Session B Enter a Value GO

**Welcome to the Jones Real Estate
Brokerage Firm Homepage!**

Username
Password OK

Figure 18-1 Sample home page with initial login and welcome message

Once users access the home page, the site displays a welcome message and prompts them with a login screen. After they enter their user name and password and click **OK**, they are logged into the company's Web server and are able to access all the company's applications and Host On-Demand sessions.

18.3.1 Instructions for embedding host sessions

Take the following steps to embed your host sessions and implement Session Manager APIs in a Web page environment similar to this example shown in Figure 18-1:

1. Use the Deployment Wizard to create your Host On-Demand HTML file or files. (In the example home page in Figure 18-1, the company has created two individual sessions using the Deployment Wizard—Session A and Session B.)

In order to use Session Manager APIs successfully, you must perform the following steps:

- Select the HTML-based configuration model, as shown in Figure 18-2. Click **Next**.

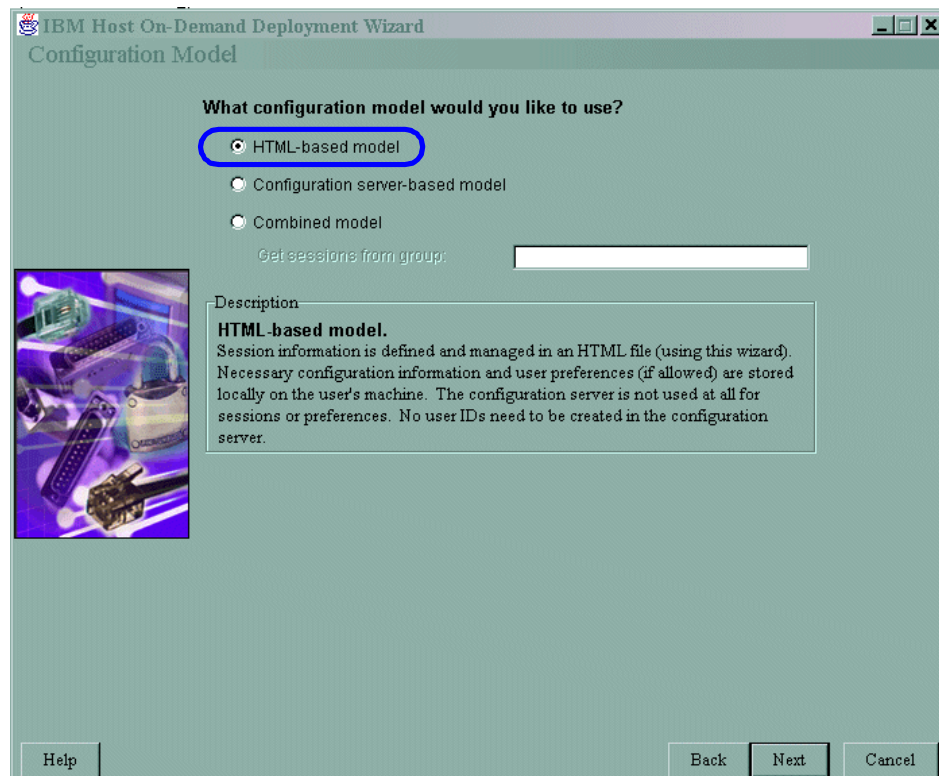


Figure 18-2 Select configuration model

- Click **Properties** on the Host Sessions window. On the Advanced tab (Figure 18-3), set Start Automatically to **Yes**. The default setting is No. Also, set Start in Separate Window to **No**. The default setting is Yes. Click **OK**.

3270 Display

Connection | **Advanced** | Proxy Server | Security | Language | Screen

Session Inactivity Timeout (minutes) ☐ Lock

Reset Insert Mode on Aid Key ☐ Yes ☒ No ☐ Lock

Enable Host Graphics ☐ Yes ☒ No ☐ Lock

Character-Cell Size ☐ Lock

Session ID ☐ Lock

Start Automatically ☒ Yes ☐ No ☐ Lock

Start in Separate Window ☐ Yes ☒ No ☐ Lock

Auto-Connect ☒ Yes ☐ No ☐ Lock

Auto-Reconnect ☒ Yes ☐ No ☐ Lock

Auto-Start Applet/Macro Options

Auto-Start ☐ Lock

Name ☐ Lock

Parameter (Optional) ☐ Lock

SLP Options

Scope ☒ Lock

This Scope Only ☐ Yes ☒ No ☒ Lock

Maximum Wait Time (ms) ☒ Lock

OK Cancel Keyboard... Help

Figure 18-3 Advanced Tab

- Click **Disable Functions** on the Host Sessions window, highlight Desktop, and disable Bookmark Sessions (Figure 18-4). Session Manager APIs do not support bookmarking due to the frame format. The default setting for Bookmark Sessions is Enabled. Click **OK** and then **Next**.

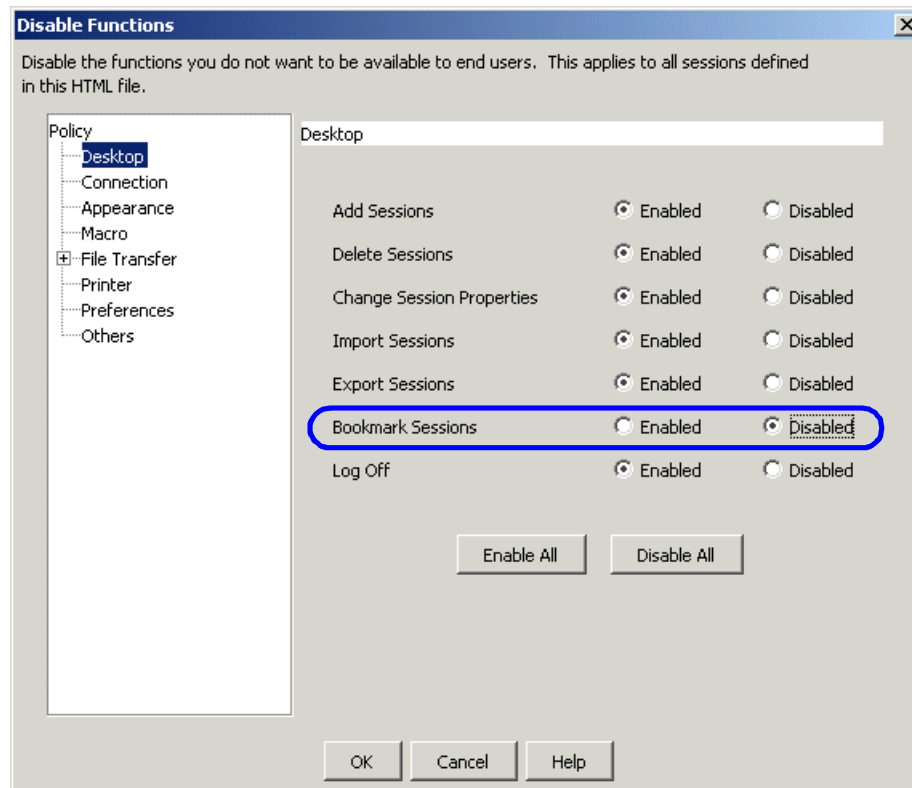


Figure 18-4 Disable Bookmark Sessions

- On the Additional Options window, click **Advanced Options** and select **Enable Session Manager JavaScript API** on the Other tab, as shown in Figure 18-5. This allows a file called HODJSAPI.js to provide the JavaScript interface to all the Session Manager APIs. HODJSAPI.js can be found in Host On-Demand's default publish directory hostondemand/HOD.

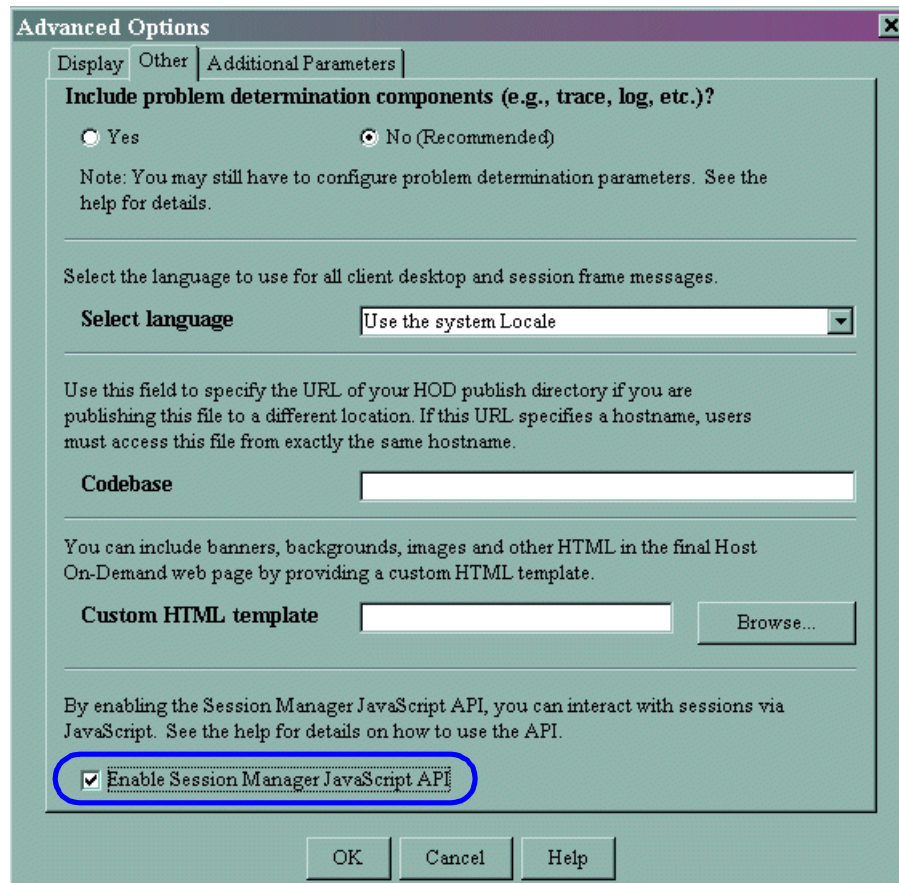


Figure 18-5 Advanced Options window

- On the Additional Parameters tab of the Advanced Options window (Figure 18-6), if you want to hide the Host On-Demand desktop and session tabs once an embedded session starts, you can type HideHODDesktop in the parameter field and true in the value field. Click **Set**, **OK**, and then **Next**.

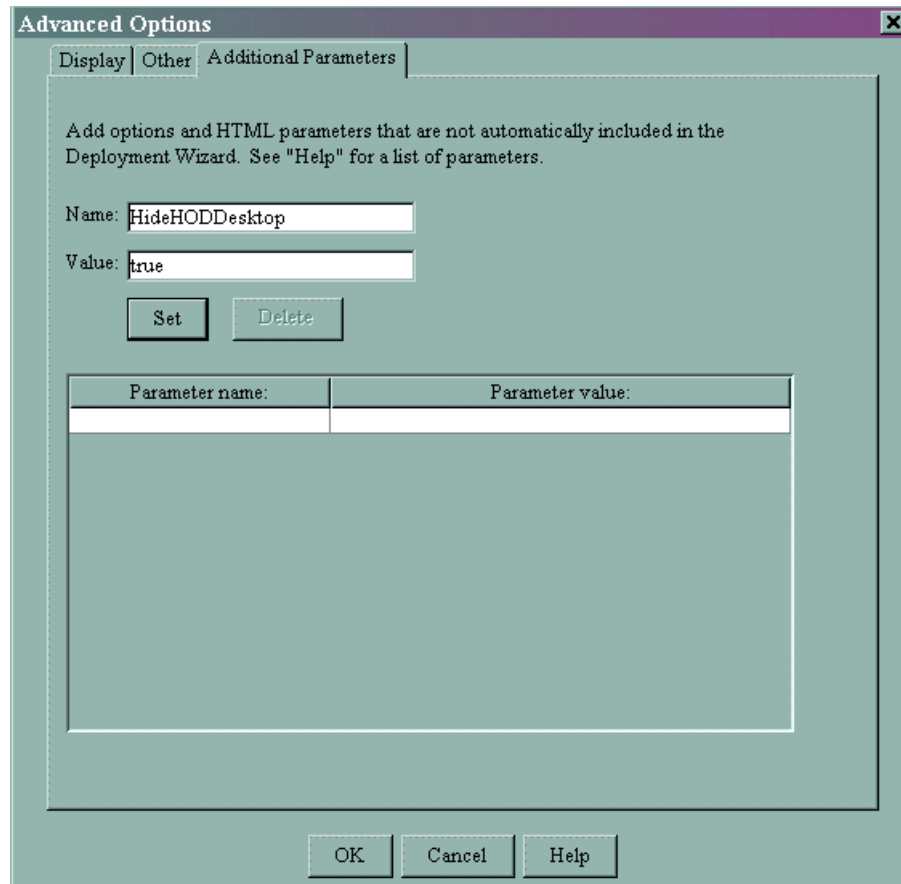


Figure 18-6 Additional Parameters window

- On the File Name and Output Format window, create your HTML files and save it to the default publish directory, which is /hostondemand/HOD. You are now finished using the Deployment Wizard.
2. Create the main Web site (Figure 18-7 on page 656) that your clients will use as a home page to access your company data. Use the following code as an example of how to divide your page into three separate frames. Notice that each frame loads a different HTML file (navigation.html, contents1.html, and Wizard_file.html).

We have allocated 36 percent of the browser's frame to the first frame, 64 percent to the second frame, and zero percent to the third frame. The reason that we did not allocate any space to the third frame is because we are hiding the Host On-Demand sessions from users. In other words, users will see only the first two frames in their browser.

Example 18-1 Pseudocode of main home page

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<HTML>
<HEAD>
<TITLE>Main Home Page</TITLE>
</HEAD>
<FRAMESET rows="36%, 64%,*" frameborder="NO" name="fs">
  <FRAME src="navigation.html" name="navigation">
  <FRAME src="contents1.html" name="company_files">
  <FRAME src="Wizard_file.html" name="contentsHOD">
</FRAMESET>
<BODY><P>To view this page, you need a browser that supports frames.</P></BODY>
</NOFRAMES>
</FRAMESET>
</HTML>
```

3. Create a file named navigation.html. The contents of this file are used in the first (top) frame of the home page that you created in the previous step. Use the following code as an example of how to do the following:
 - allow the bottom frame to swap out other HTML files based on the user action in the top frame
 - add JavaScript functions
 - add buttons

Example 18-2 Pseudocode of navigation.html

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head><!-- Navigation HTML file -->
<META http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
<META http-equiv="Content-Style-Type" content="text/css">
<title>Navigation page</title>

<script language="JavaScript">

<!-- show me
var printBuf = " ";
var savedData = "";

//provide the swapping mechanism
var current = 0;
var row_def = new Array("36%, 64%, *", "36%, *, 64%");
```

```

function swap(index) {
    parent.fs.rows = row_def[index];
    current = index;
}
//add the JavaScript functions
function displaySession(SessionName) {
    swap(1);
    parent.contentsHOD.getHODFrame().displaySession1(SessionName);
}
function sendLoginData() {
    parent.contentsHOD.getHODFrame().sendKeys1(savedData);
}
function sendKeys(data) {
    parent.contentsHOD.getHODFrame().sendKeys1(data);
}
function setloginData(username, password) {
    saveData = username + "[tab]" + password + "[enter]";
    window.status="Saved: " + saveData;
}
function showOtherApp(pg) {
    swap(0);
    parent.contents1.location = pg;
}
function printScreen() {
    printBuf = "";
    var buf = parent.contentsHOD.getHODFrame().getString();
    var row = 1;
    for (row = 0; row < 24; row++) {
        printBuf += buf.substring(row*80, (row+1)*80) + "<br>";
    }
    openPrintWindow();
}
function openPrintWindow() {
    var printWindow = window.open("JSDemoPrint.html", "PrintWindow",
    "menubar,height=500,width=650");
    if(printWindow.opener == null) printWindow.opener = self;
}
</SCRIPT> //end of JavaScript
</head>

<BODY>

//add the welcome message
<div align="center"><font size="5"><i><b>Jones Real Estate Brokerage
Firm</b></i></font></div>

//create a table, add the buttons, and map the onclick parameters to the
JavaScript functions defined above

```

```

<table>
<tr>
<td><div>

<td><div></div></td>

<td><div></div></td>

<td><div></div></td>
</tr>

<tr>
<td><div></div></td>

<td><div></div></td>

<td><div><FORM name="Go">Enter a Value
<INPUT type="text" name="SSValue" size="20">
<IMG src="GO.gif"
onclick="sendKeys(window.document.Go.SSValue.value+'[enter]');>
</FORM></div></td>

</tr>
</table>

</BODY>
</html>

```

-
4. Create your company welcome page with the initial login. In our example in Figure 18-7 on page 656, we have created a file named contents1.html. It displays the string 'Welcome to the Jones Real Estate Brokerage Firm Homepage!' and prompts users for a user name and password when they click the **Welcome** button in the top frame. Once they enter their user name and password and click **OK**, they are logged into the company's Web server and can access the company's files as well as their Host On-Demand sessions.

Example 18-3 Pseudocode of contents1.html

```

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<HTML>
<HEAD>
<META http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
<META http-equiv="Content-Style-Type" content="text/css">
<title>Welcome</title>
</HEAD>
<BODY>

```

```

<div align="center"><font size="5"><i><b>Welcome to the Jones Real Estate
Brokerage Firm Homepage!</b></i></font></div>

<FORM name="saveform"><BR>
<INPUT size="20" type="text" name="username">
<INPUT size="20" type="text" name="password">
<INPUT type="button" value="OK"
onclick="parent.navigation.setLoginData(window.document.saveform.username.value
, window.document.saveform.password.value);">
<BR>
</FORM>

</BODY>
</HTML>

```

5. Create your Contact Information page. This page will display in the bottom frame when users click **Contact Info** in the top frame. Our contact information file is named contents2.html.

Example 18-4 Pseudocode of contents2.html

```

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<HTML>
<HEAD>
<META http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
<META http-equiv="Content-Style-Type" content="text/css">
<title>Contact Information</title>
</HEAD>
<BODY>
//add your company contact information here
</BODY>

```

6. Place all HTML files in your Host On-Demand publish directory. The default directory is /hostondemand/HOD.

18.3.2 Explanation of Session Manager APIs in this scenario

Now that we have taken you through the steps to create the HTML files that are used in this customer scenario, we can explain more about how this company's site makes use of Session Manager APIs.

Looking at the company's main home page in Figure 18-7, recall that only two out of three frames are visible on the browser. The top frame displays the navigation.html file that you created in step 3. on page 652, and the bottom frame displays the contents1.html file that we discussed in step 4. on page 654.

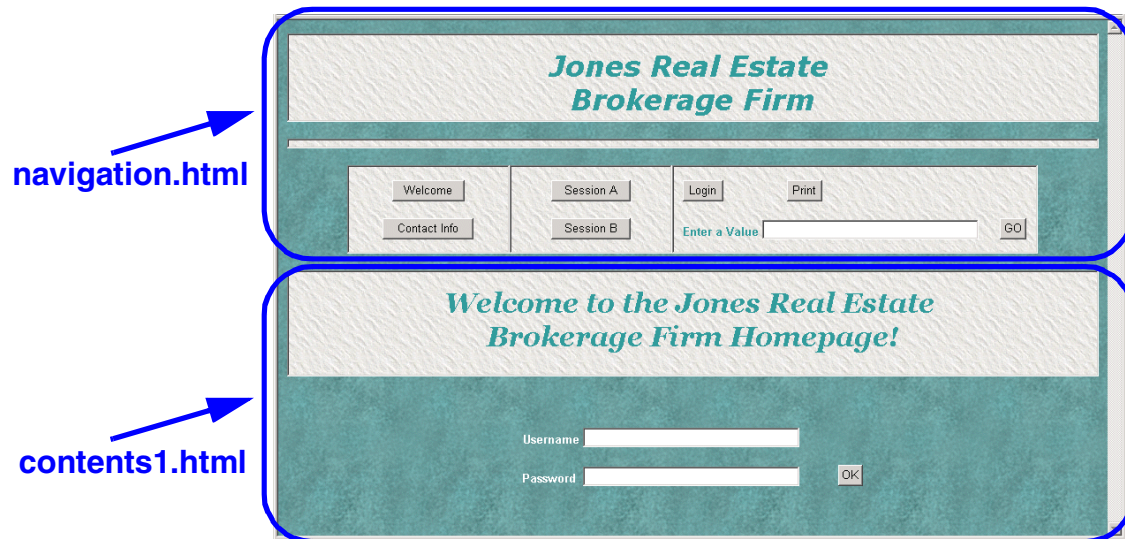


Figure 18-7 Main home page

Remember that the file you created using the Deployment Wizard (Wizard_file.html) in step 1. on page 647 is hidden from view because we did not allocate any space to it.

The contents of the bottom frame changes depending on what the user selects in the top frame. This bottom frame can display one of two HTML files (Welcome and Contact Info) or one of two Host On-Demand sessions (Session A and Session B). For example, Figure 18-8 shows the home page after the user clicks the Session A button.

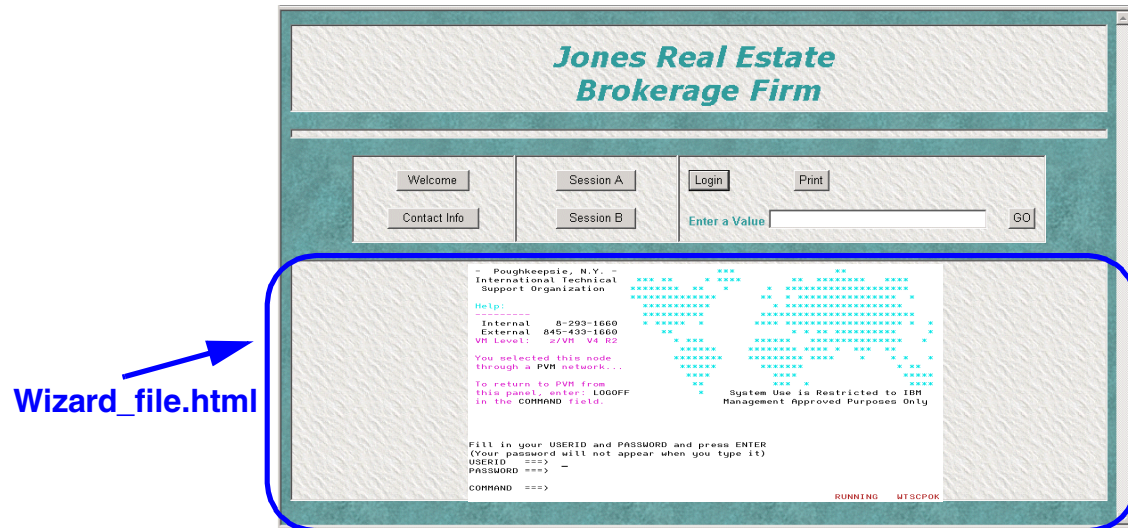


Figure 18-8 Sample home page with Session A displayed

The Session A, Session B, Print, and Login buttons as well as the Enter a value field represent JavaScript APIs. Now we describe these built-in functions in more detail.

- ▶ When users click Session A or Session B, the displaySession JavaScript API is called to display the Host On-Demand session in the bottom JavaScript frame. Each of the session buttons will swap in one of two sessions and display the session's 'green screen', as shown in Figure 18-8. Remember that the HideHODDesktop parameter was added in the Deployment Wizard file to hide the Host On-Demand desktop (See Figure 18-6 on page 651).
- ▶ When users click Print, the getString JavaScript API is called to retrieve the visible characters of the presentation space (called the text plane) using the TEXT_PLANE parameter. Once the API retrieves the information, the information is displayed in a pop-up window.

The data is returned starting from the beginning of the presentation space and continuing until the JSHostOnDemand applet's buffer is full, or the entire text plane has been copied. It removes duplicate DBCS characters before the text plane data is returned.

You can use the length variable of String object to get the number of characters copied to the returned string, or you can call the getStringLength method immediately after a getString call to get the number of characters copied to the returned string.

- ▶ When users click Login, the sendKeys JavaScript API is called to send the user ID and password to the current cursor position on the presentation space.

The string consists of keystrokes that can contain text characters such as the Enter key, the Tab key, or the Page Up key. These special keys are represented by keywords that are delimited by square brackets and called mnemonics, such as [enter], [tab], and [pageup].

For example, in this scenario, when users first access the company's home page, the site prompts them with a login screen to access the company Web server, as shown in Figure 18-7 on page 656. Once they type the user name and password and click **OK**, the data is saved in the JavaScript code. Once they click either Session A or Session B and then Login, the string `userID[tab]password[enter]` is used to send the user's ID to the session screen, tab to the next field, send the user's password to the session screen, and finally execute the command. The user is now logged into the host application and is presented with the next screen.

For a list of mnemonic keywords for the SendKeys method and the type of session or sessions in which the mnemonic is supported, see the *Session Manager API Reference* in the Host On-Demand InfoCenter.

- ▶ The Enter a value function calls the sendKeys1 JavaScript function. Similar to the Login function, the Enter a value function allows users to type a string of keys in the field and send this string to the current cursor position on the presentation space.

For example, the consultants from the brokerage firm in this scenario need to enter customer information into one of the host sessions. Instead of having to navigate on the session screen and type information, the consultants can simply enter strings of keystrokes and mnemonics into the Enter a value field and click Go.

18.4 Description of working demonstration

Refer to Appendix D, "Additional material" on page 1035 for procedures you can follow to download the HTML code for a complete, working example using the Host On-Demand Session Manager APIs. Instructions on where to find the code, how to download it, and file names are included in the appendix. The example code must be placed in your Host On-Demand publish directory (hostondemand/HOD).

Similar to the real-life scenario above, the demonstration shows you working code of how to embed Host On-Demand sessions in a Web infrastructure that contains a main home page with three frames, two of which are visible in the browser. In this example, the Web developer has created three Host On-Demand sessions using the Deployment Wizard as well as two personal HTML files. Using a swapping mechanism, the bottom frame displays either one of the three host sessions or one of the two personal HTML files at a time. The developer has also added JavaScript functions that allow the user to display, start, and stop host sessions, print the contents of the bottom frame, save data and send it to the presentation space, and play a macro.



Host printing

With the host printing sessions, you can print host-application files on a printer that is directly attached to your workstation or to a network printer.

19.1 Overview

Host On-Demand provides both a 3270 and a 5250 printer session, which run through a browser and use a Java interface in the same way as a display session. Depending on the session type and platform, Host On-Demand provides three printing modes:

- ▶ Java file interface mode (3270 and 5250, all supported platforms):

This is the printing mode used by all non-Windows platforms, and that has been supported since Host On-Demand Version 3.

In this mode, printer sessions run through a browser and use the Java File I/O API. All print data is converted into each printer's control language based on the definitions in a printer definition table (PDT) and sent to the system's printer port. As a result, printer sessions cannot use the drivers provided by the workstation's native operating system when using this mode.

For 3270 printer sessions the print data can be printed to file as a portable document format as used by an Acrobat reader

- ▶ Windows spooler interface mode (3270 and 5250, Windows platforms only):

This new printing mode is supported by Host On-Demand Version 6 and later on Windows platforms only. Like the Java file interface mode, this mode also uses a PDT to format the data, but instead of using the Java file interface, it uses the Windows native spooler interface to send the text and printer commands to the printer.

This mode is useful when the attached printer cannot be accessed by the Java file interface, for example:

- A USB or serial port attached printer
- Novell NetWare network printers from Novell Windows 9x clients

- ▶ Windows native printer interface mode (3270 mode, and Windows platforms only):

This new mode formats the data more like a Windows application, using a Windows printer driver and Windows font. This mode does not require a PDT to be specified, since all formatting of the data stream is managed internally by the native printer interface.

This mode is useful when printing to a Postscript printer, or when a job should be printed with a Windows font.

When selecting a Windows font via the Page Setup tab (see Figure 19-1), the font style (regular/italic/bold) and font size are governed by Host On-Demand.

- For font style, Host On-Demand always uses “regular” except for the intensified characters on LU3 sessions. Those intensified characters are printed with the bold font.

- For font size, Host On-Demand calculates from the CPI on single-byte character set sessions. For double-byte character sets, Host On-Demand will use 10 point if the character spacing is wide enough. Otherwise Host On-Demand calculates an appropriate point size based on the LPI and CPI values.

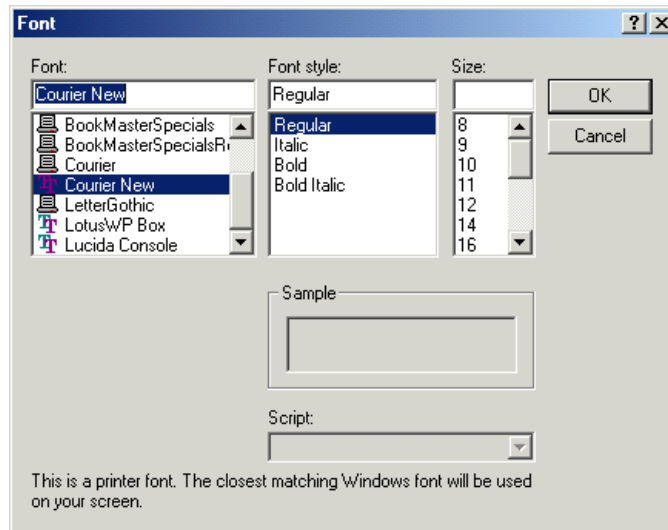


Figure 19-1 Selecting a Windows font

Generally, the printing modes that use a PDT (Java file interface or Windows spooler modes) will provide better performance, while the Windows native printing mode will utilize more Windows system capabilities. On non-Windows platforms, the use of a PDT is mandatory. PDTs provide great flexibility because you can tailor them to produce the desired printed output without having to modify the host application.

If you are using a Host On-Demand client downloaded from a server, the PDT needed for a printer session is stored on the server and downloaded with the client. A locally installed client stores the PDT on the client workstation.

Several PDTs and their equivalent Printer Definition Files (PDFs) are provided, plus you can create your own.

The following single-byte character sets (SBCS) are provided:

- Basic ASCII text mode
- HP PCL Level 3 (Laser Printers)
- IBM PPDS Level 2
- IBM PPDS Level 1 (Proprinter XL, X24, XL24)

Selections made on the Printer tab (see Figure 19-2) will determine what print mode will be used.

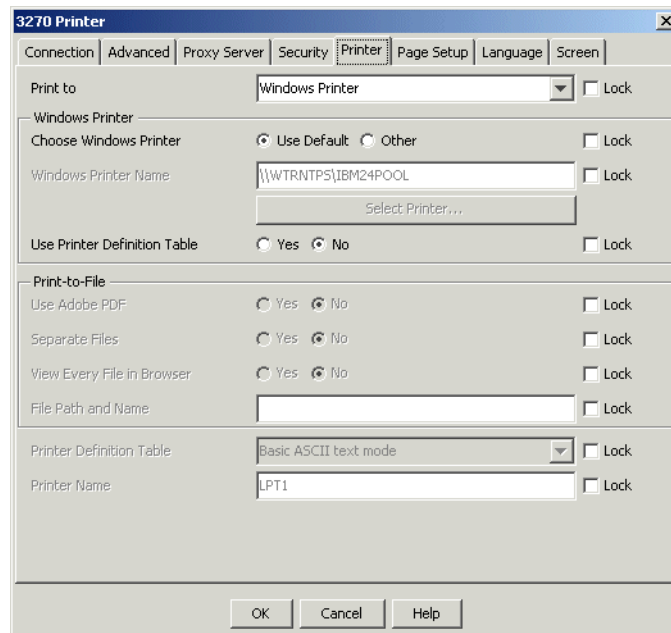


Figure 19-2 Printer tab on the 3270 printer session window

If the Print parameter is set to:

- ▶ **Other**, Java interface, printing directly to printer port or to file via printer definition file or as portable document format
- ▶ **Windows Printer**, and Use Printer Definition Table is set to **No**, the Windows native printing mode is used.
- ▶ **Windows Printer** and Use Printer Definition Table is set to **Yes**, the Windows spooler interface mode is used.

19.1.1 Types of printer LU

The following information will help you decide how to configure the LU or pool name for a 3270 printer session and the Associated Printer for a 3270 display session. It also provides an explanation of the two types of printer sessions supported by Host On-Demand.

LUs are defined in different ways by various Telnet servers. To decide how to configure the Host On-Demand LU or pool name, talk to your administrator of the system programmer responsible for the server.

Explicit or implicit LU

On most servers, an LU can be defined as specific or as a member of a pool (often referred to as explicit or implicit, respectively).

- Explicit Printer LU
When a specific LU is configured in the Telnet server, you must enter the name of that LU in your session configuration. This ensures that the session always connects to the same LU, which is helpful for system management purposes.
- Implicit Printer LU
When an implicit LU is configured on the Telnet server, you can either leave the LU or pool name blank or enter the name of a pool. If you leave the name blank, the session connects to the first available LU that is defined at the server as implicit or pooled. See “LU pools” on page 666.

The LU name on the server may differ from the name of that LU at the host. Example 19-1 shows a configuration managed by a Communications Server for OS/2 server.

Example 19-1 Example configuration on Communications Server for OS/2

[<warpserver>-C:\]cmtn3270

Local			Host		Host		Idle		LU	mins
LU name	Class	Assoc	LU	IP address	Primary	Second	Status	LU		
@LUA0001	IW			9.24.106.179	LU-LU		ADPAVM4	X1F80202	0	
@LUA0002	IW				INACTIVE			0		
@LUA0003	IW				INACTIVE			0		
@LUA0004	IW				INACTIVE			0		
@LUA0005	IP			9.24.106.179	LU-LU		ADNAVEN8	X1F80206	0	
@LUA0006	IW				INACTIVE			0		
@LUA0007	IW				INACTIVE			0		
@LUA0008	IW				INACTIVE			0		
@LUA0009	IW				INACTIVE			0		
@LUA000A	IP				INACTIVE			0		

As you can see, the LU @LUA0001 in the Communication Server gateway is connected to the host LU X1F80202 and @LUA0005 to X1F80206. Both LU names are important for printer sessions:

- The LU name in the TN3270E gateway has to be configured in the HOD printer session configuration if explicit LUs are used. It also appears in the title bar of the session window.

- ▶ Your host application has to send the print data to the LU name known at the host, for example, to X1F80206 (Class = IP means this is a printer session).

Therefore, we recommend that you give the printer LUs in the server the same names as they have at the host, so that the host LU name appears in the title bar of the printer session window and you can see easily to which LU a print job has to be sent.

LU pools

If a group of LUs is configured in a pool at the server, the session connects to the first available LU in the pool you name. Servers handle pools in different ways, so you must configure Host On-Demand as follows:

- ▶ IBM Communications Server for Windows NT
You can enter the name of a pool, or you can leave the LU or pool name blank if the LU desired is included in the default pool.
- ▶ IBM Communications Server for OS/2
This does not support pools. If you want to connect to the first available LU (implicit), leave the LU or pool name blank.
- ▶ Microsoft SNA Server
Leave the field blank.
- ▶ IBM Communications Server for AIX
Leave the field blank, and the session will connect to the first available LU, or enter the name of a pool.

19.2 3270 printer session

A Host On-Demand 3270 printer session emulates an IBM 3287 printer in either LU Type 1 or LU Type 3 mode. The LU-type is configured at the host system, and Host On-Demand detects it automatically when the session is established. The configured LU type is significant, since some configuration parameters apply to one type or the other, and may also necessitate customization of any printer definition tables (PDT) you are using.

- LU Type 1** This has an SNA Character String (SCS) data stream, which contains a series of characters, formatting commands, and attributes that can be translated from EBCDIC to ASCII and sent immediately to the printer.

Note: Only LU1 is the type of data stream which can be used with printer definition files. The imbedded formatting commands can be translated by printer definition files to the appropriate escape sequence for the printer

LU Type 3 This has a data stream that is very similar to that of a display. It is formatted by the host in a buffer, then sent to the printer

Host On-Demand provides the following three printing modes for the 3270 printer sessions:

- ▶ Java file interface mode
- ▶ Windows spooler interface mode (Windows platforms only)
- ▶ Windows native printer interface mode (Windows platforms only)

19.2.1 Configuring a 3270 printer session

The 3270 session definition consists of 8 tabs:

- ▶ Connection
- ▶ Advanced
- ▶ Proxy
- ▶ Screen
- ▶ Security
- ▶ Printer
- ▶ Page setup
- ▶ Language

Most of these tabs and their parameters are documented in “3270 and 5250 display sessions” on page 286. So the following paragraphs will only discuss the parameters which are specific to printing

3270 printer Connection tab

.This tab, shown in Figure 19-3, defines the server to which the printer will connect. The only printer-specific parameter on this page is Print-Buffer Size. Select the printer buffer size that this printer session is expected to have.

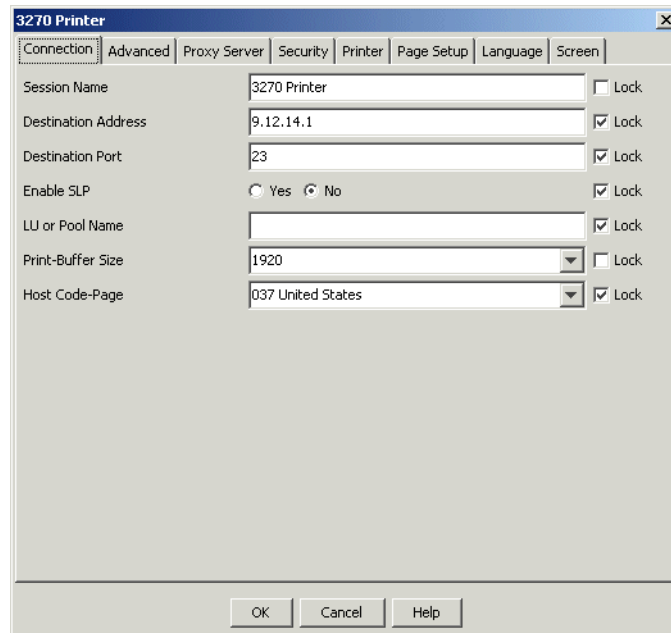


Figure 19-3 3270 printer Connection tab

3270 printer Advanced tab

The printer advanced tab contains the same parameters as for the display sessions explained in "3270/5250 Advanced tab" on page 294.

3270 printer Proxy tab

This printer tab contains the same parameters as for the 3270/5250 display sessions. explained in “3270/5250 Proxy tab” on page 301

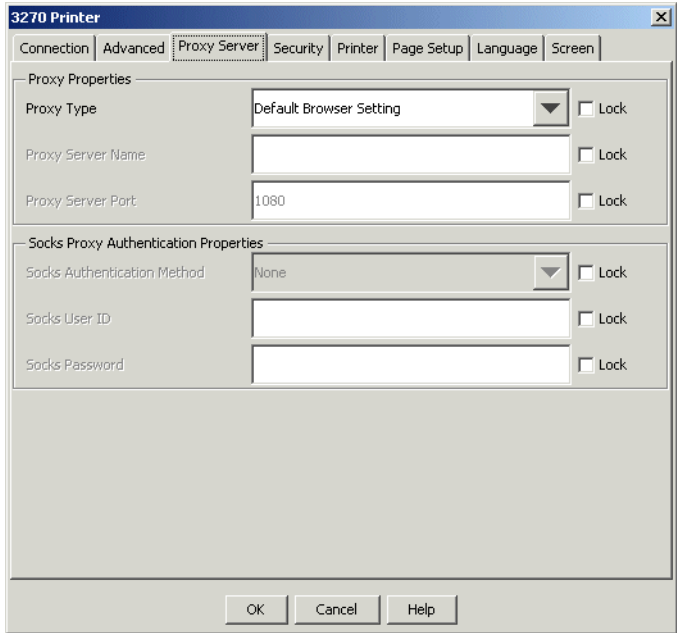


Figure 19-4 3270 Printer Proxy tab

3270 printer Security tab

This printer tab contains the same parameters as for the 3270/5250 display sessions explained in “3270/5250 Security tab” on page 304

3270 printer Printer tab

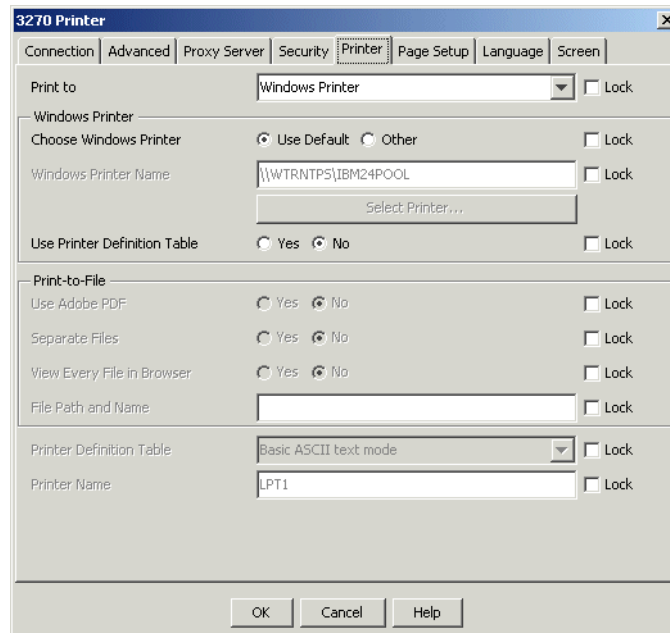


Figure 19-5 3270 Printer tab

Some of the parameters are generic for printer sessions and are used in the 5250 printer session as well.

► **Print To (3270/5250)**

Select to print to a local Windows printer (Windows platforms only), to another type of printer (for example, LPT1), or to a file. For 5250 printer sessions, printing to another type of printer requires a printer name. For 3270 printer sessions, printing to another type of printer requires a printer name and a printer definition table (PDT).

On Windows platforms, the default is Windows Printer. On non-Windows platforms, the default is Printer.

► **Windows Printer (3270/5250)**

This group box lists the options that are available only on Windows platforms.

– **Chose Windows Printer (3270/5250)**

Select Use Default to use the default Windows printer. If you select Other use the Select Printer button. Clicking that button opens the Windows Printer Setup dialog from which you can select from the printers defined on your system and as well their settings.

- Windows Printer Name (3270/5250)
- Displays the currently-selected Windows printer name. On Emulator clients (for example, HOD.html), this field is read-only. Click Select Printer to change the printer selection. On Administration clients, (for example, HODAdmin.html), you can type any printer name in this field. Make sure the specified printer name is available on the client machines.

The default value is Windows Default Printer.

- Select Printer (3270/5250)

Click this button to see the Print Setup Windows common dialog window where you can specify various settings for printing, including the printer to be used.

- Use Printer Definition Table (3270)

Choose whether a PDT is used or not.

If you select No, the Windows graphical device interface (GDI) is used, and you can specify a printer font on the Page Setup tab.

If you select Yes, the Windows spooler API is used for printing with a PDT, and you are required to specify the PDT. This selection provides better print performance over the GDI in many cases.

The default is No.

► Print-to-File

This group box lists the options that are used for printing to a file instead of a printer.

- Use Adobe PDF (3270)

Select Yes to generate an Adobe PDF (portable document format) file. This is an option only if you select to print to a file.

The default value is No.

If printing Adobe PDF files please make sure you have configured in the Page Setup tab the Advanced Options for PDF according to your requirements. See Figure 19-6 on page 672.

Some limitations for PDF printing apply (e.g. limited paper size selection etc.). See as well Chapter 19.7, “Adobe PDF printing” on page 694

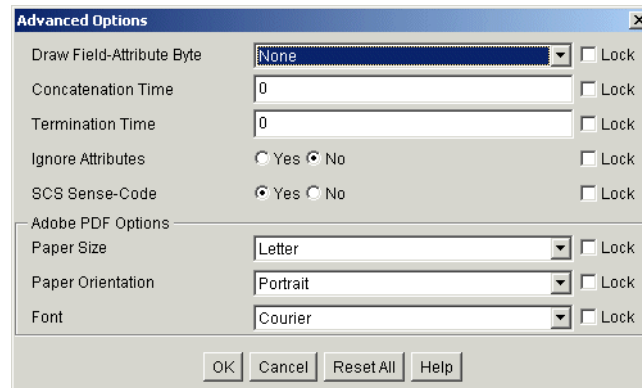


Figure 19-6 Advanced Options window with PDF parameters

- Paper Size

Select the paper size to be used in the generated PDF files. The default value with US English locale is “Letter.” See National Language Support for the default values when other locales are being used.

- Paper Orientation

Select either Portrait or Landscape paper orientation. The default value is Portrait.

- Font

Select a font from the drop-down list of predefined fonts to use in the generated PDF file. The default font with US English locale is Courier. See National Language Support for the default values when other locales are used.

The list of fonts is also different depending on the host code page that is currently selected. Note that the PDF file generated with the LucidaConsole font or the CourierNewPSMT font might not be displayed correctly on non-Windows or older Windows platforms. This display problem does not occur on newer Windows platforms that support OpenType fonts by default (Windows 98 Second Edition, Windows ME, Windows 2000, and Windows XP).

- Separate Files

When the print destination is a file, you can choose whether you want to save each print job to a unique file or to have jobs appended to each other in one file. When the Use Adobe PDF option is set to Yes, this option is not available and each print job is saved to a unique file.

- View Every File in Browser

Select Yes to view files in a browser after they are created. You can then view or print the file from your browser. If you want to view Adobe PDF files, you need to have Adobe Acrobat Reader plug-in (or equivalent) installed in your browser environment.

Note that thumbnail images of Adobe PDF files generated by Host On-Demand do not always appear.

– File Path and Name

When the print destination is a file, type the path and name of the file. If the file path and name already exist on the client, Host On-Demand will print the file to that destination and will overwrite any files that already exist there. If the file path and name do not exist on the client, they are automatically created and the files will be printed to that destination. You can then view or print the file using the appropriate viewer on the client.

Note: If you do not type the path of the file, Host On-Demand will write the file to your browser's default directory. Your browser's default directory depends on your operating system. Refer to the Host Printing Reference for more information.

- If you choose Separate = Yes in the Separate Files field, you have a choice:

You can specify a unique name for each file.

Put an asterisk in the file name. The file name is numerically incremented for each print job. For example, if you name the file prt*.file and the Use Adobe PDF option is set to No, the first file will be named prt000.file, the next will be named prt001.file, and so on.

When the Use Adobe PDF option is set to Yes, the file extension will always set to be equal to “.pdf” and one asterisk is converted into an eight-figure counter. For example, if you name the file prt*.file and the Use Adobe PDF option is set to Yes, the first file will be named prt00000001.file.pdf, the next will be named prt00000002.file.pdf, and so on.

You can let Host On-Demand generate the name.

Do not use the asterisk in the file name. For example, type the name as prt.file. As long as the Use Adobe PDF option is set to No, Host On-Demand appends numbers to the file name, starting at prt.file.000, prt.file.001, and so on.

When the Use Adobe PDF option is set to Yes, Host On-Demand generates a file name by adding an eight-figure counter value and “.pdf” file extension. For example, when you type the name as prt.file and the Use Adobe PDF option is set to Yes, the first file will be named prt.file00000001.pdf, the next will be named prt.file00000002.pdf, and so on.

- If you choose Separate = No and the Use Adobe PDF option is set to No, a single file is created and each job is appended to this file. A system-generated print-job name is added to the start of each job so that jobs can be identified. If the file already exists, the system will continue to append to it.

You can also specify an external command to run after host print jobs using this field. Refer to Running external commands after host print jobs in the Online Help.

Refer to the Host Printing Reference for more information about Adobe PDF files, file paths, and file names.

► Printer Definition Table

A printer definition table (PDT) formats print data sent by the host application so it can be printed on a workstation printer.

The PDT you select must be suitable for the printer and for the printer-emulation mode that the printer will use (PCL, PPDS etc; note that PostScript is not supported). You can create your own PDTs, which are automatically added to the pull-down list.

Select a name from the pull-down list.

If you are not sure which printer emulation modes are supported by your printer, you must refer to the printer's technical documentation, which usually lists the supported modes.

In some cases, it may be necessary to change the settings on the printer itself so that they match the mode intended for the PDT that you want to use. Some printers can switch between modes automatically or supply software that enables you to change the mode. It is important to refer to the printer documentation to decide which PDT to use and how to set the correct mode on the printer.

You might find it useful to go to the printer manufacturer's Web site for information.

Most laser printers can use HP PCL Level 3. Level 3 commands are understood by later levels.

Basic ASCII text mode may work if your printer does not support one of the other modes supported by Host On-Demand; however, if you use this mode, the commands that are unique to your printer will not be available.

Host On-Demand does not support PostScript mode with a PDT. If you are using Host On-Demand on a Windows platform, you can use your PostScript printers as a Windows printer without a PDT.

VT sessions do not use a PDT when non-Bidi code page is being selected. Printer data from the VT application is sent as-is to the printer device. You must insure that your VT application supports the printer you want to use.

► **Printer Name**

Type the name of the port for the printer you want to use. On Windows workstations, you can also type the UNC (Universal Naming Convention) name of a network printer in either of two formats:

```
\\server_name\printer name  
\\server's_host_name_or_IP_address\printer name
```

For example, if you are configuring a printer on Windows 95 or NT, you can type a port name such as LPT1, or a network printer name such as \\myhost\printer. If you are configuring a printer on UNIX, type a device name such as /dev/lp0.

For further details and 5250 specific parameters please refer to the online help of that tab.

3270 Page Setup tab

The Page Setup tab window as shown in Figure 19-7 on page 676, lets you choose several options used in 3270 host print. When a PDT is being used, some values set on this panel will temporarily override the values set in the PDT. The changes are effective only for sessions started from this configuration; they do not alter the PDT. Change these options only if you are familiar with VTAM and with LU Type 1 and LU Type 3 protocols.

The values you set remain in effect for this configuration, even if your administrator later modifies and recompiles the PDT. The host SCS commands take precedence over the following when the Bestfit options was not set to Yes on the Advanced Options window:

- Characters per inch
- Lines per inch
- Maximum lines per page
- Maximum characters per line

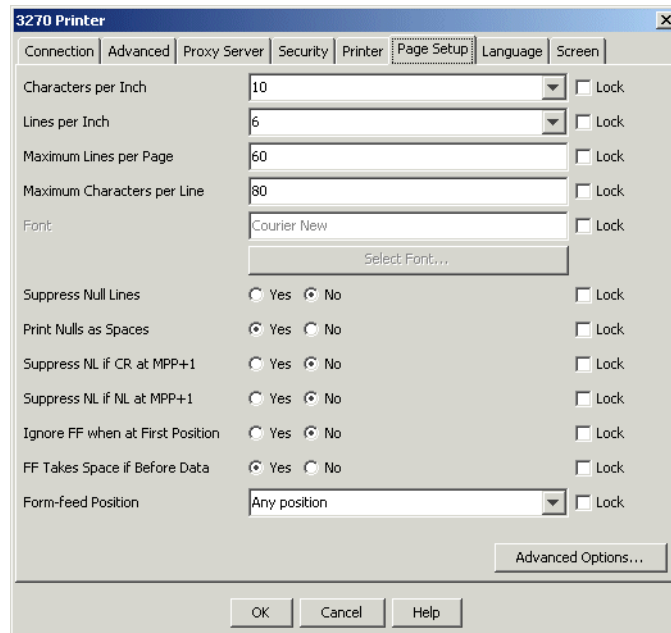


Figure 19-7 printer Page Setup tab

3270 printer Language tab

This printer tab contains the same parameters as for the 3270/5250 display sessions explained in Chapter , “3270/5250 Language tab” on page 308

3270 printer Screen tab

The Screen tab for printers controls what options the user has available on the session window representing the printer session. The controls are shown in Figure 19-8 on page 677.

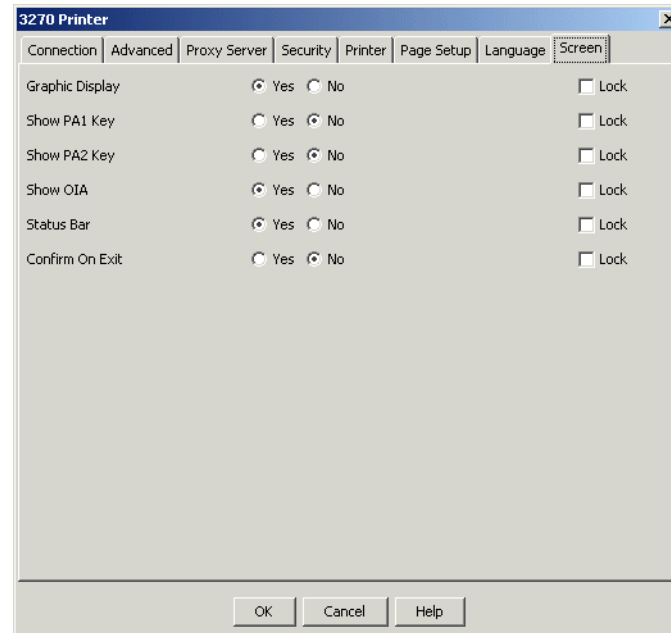


Figure 19-8 3270 printer Screen window

The following describes the options:

- ▶ **Graphic Display**
If you turn on Graphic Display, the information window for this session will show the printer, workstation and host system as icons. Therefore the window will be bigger than when just displaying the raw information.
- ▶ **Show PA1 Key**
Specifies whether the Program Attention 1 key should be displayed on the window as a button.
- ▶ **Show PA2 Key**
Specifies whether the Program Attention 2 key should be displayed on the window as a button.
- ▶ **Show OIA**
Determines whether the operator information area (OIA) is visible on the window (recommended).
- ▶ **Status Bar**
Determines whether the status bar is visible at the bottom of the window when the session starts (recommended).

► Confirm on Exit

Select Yes if you want a warning message to appear when a user attempts to close a session. If users select File > Exit, close a session window, exit from the toolbar, or right-click the left corner of the session window, a window appears asking if they really want to exit. If the user clicks OK, the session ends. If the user clicks Cancel or closes the window, the session remains open and unchanged. If the user closes the browser window, no exit warnings appear. If the user closes both a session and its associated printer session, the exit warning appears only once.

The default is No.

19.2.2 Using a 3270 printer session

When you start a printer session, you should see a window similar to the one shown in Figure 19-9.

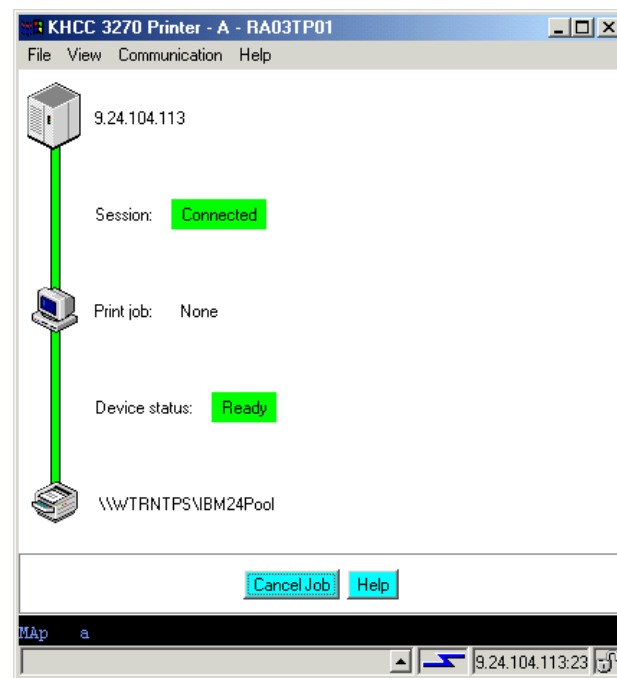


Figure 19-9 Connected 3270 printer session

To send a print job, issue the necessary command in the host application to send print output to the LU as it is named at the host, not as it is named in the communications server.

The session window

This section is a description of what you may do in the session window:

Title Bar

The title of the printer session window includes parameters that you have configured and perhaps an ID that is generated by the system. They appear in the following order, delimited by '-':

- ▶ The session name (for example, KHCC 3270 Printer)
- ▶ The session ID or short name
- ▶ The printer LU name (for example, RA03TP01)
- ▶ The LU name of the associated display session, if any (for example, RA03TN01)

Menu Bar options

The Communication and Help items are much the same as in a display session, but others are different.

- ▶ File

This allows you to print a test page, or eject a page. You can also select another printer or modify the page properties without restarting the session. Configurable items that have been locked by the administrator will appear greyed out, and cannot be modified.

Printing a test page causes a page to be printed that contains some standard text that should be formatted correctly. It also serves as a test of the connection between the workstation and the printer; it does not test the link to the server or host.

- ▶ View

From this menu, you can toggle the display of the status bar, and the graphic display. Switching off the graphic display removes the pictures of the devices and the links between them. Once you are used to the meanings of the status indicators, you may wish to remove the graphics.

Main emulator window

The main emulator window shows the following information:

- ▶ IP address or host name of the communications server
- ▶ Link status

The lines between the server, the workstation and the printer show green or red, depending on whether they are connected or not.

- ▶ Session status
 - Disconnected

Indicates that there is no session between Host On-Demand and the Telnet server. If the session is connecting to the host system through a communications server/gateway, this indicates the status of the connection to the server/gateway, not to the host.

- COMM nnn

Indicates a communications problem. Note the specific number (nnn), then use the “?” button to get more information. If the status bar is displayed, you can also select the message in the status bar history, and click “?” for help.

- PROG nnn

Indicates an error in the print data stream from the host system. Note the specific number (nnn), then click the ? button to get more information.

- Connected

Indicates that Host On-Demand and the Telnet server are connected. If the session is connecting to the host system through a communications server or gateway, this state indicates the status of the connection to the server or gateway, not to the host. However, column 3 of the OIA indicates the status of the host connection, as follows:

An asterisk (*) indicates that the session is connected to an application program (LU-LU connection)

A p indicates that the session is connected to a host but not to an application (SSCP-LU connection).

- Print Job:

This field will contain one of the following to describe the status of the print job:

- None

Indicates that there are currently no print jobs being sent from the Telnet server.

- Waiting

Indicates that Host On-Demand is trying to send a print job to the printer or file. However, it cannot begin or continue the job because of the status of the printer or file.

- Page nnn

Indicates that page number nnn of the print job is currently being printed. This is the page number as viewed by the host; the printer may not be printing that page (3270 only).

- Canceling

Indicates that the user has clicked **Cancel Job**. This state will be maintained until the host application has sent the rest of the print job because Cancel Job cannot stop the host application. Host On-Demand must process the rest of the print job, but does not print it. This can take a while for a large print job.

► **Device status**

– **Ready**

Indicates that the printer or file is active and ready to receive a print job.

– **Busy**

Indicates that the printer or file is active but is not available to receive a print job because it is currently being used by another application or session.

– **Error**

Indicates that the printer or file cannot be used at this time.

– **Printing**

Indicates that the printer or file is currently in use by this session.

► **Port**

Printer or file to which this session prints.

Cancel Job button

There is no architectural way to tell the host application to cancel a print job, so when you click **Cancel Job**, Host On-Demand continues to receive and process each page but does not print it. This may take some time for large print jobs.

PA1 and PA2 buttons

These appear only if they were switched on when the session was configured. There is no point in having them in the window unless one or more of your host applications have been written to respond to them.

Operator information area

Depending on the configuration, the OIA may be visible at the bottom of the window.

Logical configuration

Figure 19-10 shows a 3270 printer session and how it works from a logical (as opposed to the physical) perspective.

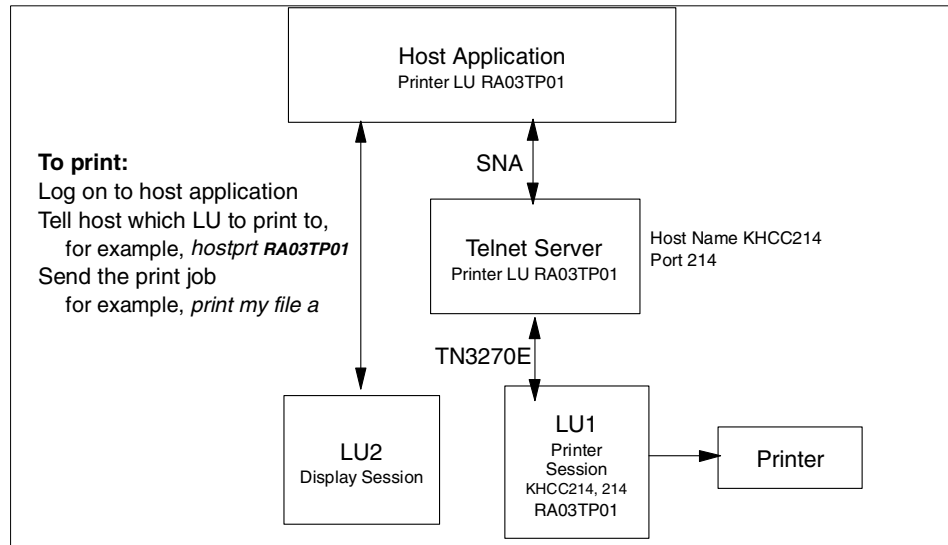


Figure 19-10 Logical configuration of 3270 host printing

19.3 3270 associated printer sessions

Telnet servers, such as Communications Server for Windows or Communications Server for OS/390, can be configured to support the association of a display LU with an associated printer LU. The purpose of this is convenience. You know that when you start the display session, a specific printer session will start; therefore if you direct your host printing to that session, it will appear on the correct printer. It is also easier to configure an associated printer session, because you do not have to enter the destination address or LU name.

The association between a display and a printer session must be made in the TN3270E communications server. In the Host On-Demand configuration notebook, you can associate the same printer session with more than one display session.

19.3.1 Configuring associated printer sessions

To configure a display session with an associated printer session:

1. Configure an empty printer session

Configure the session as usual but leave the Destination Address blank, as shown in Figure 19-11.

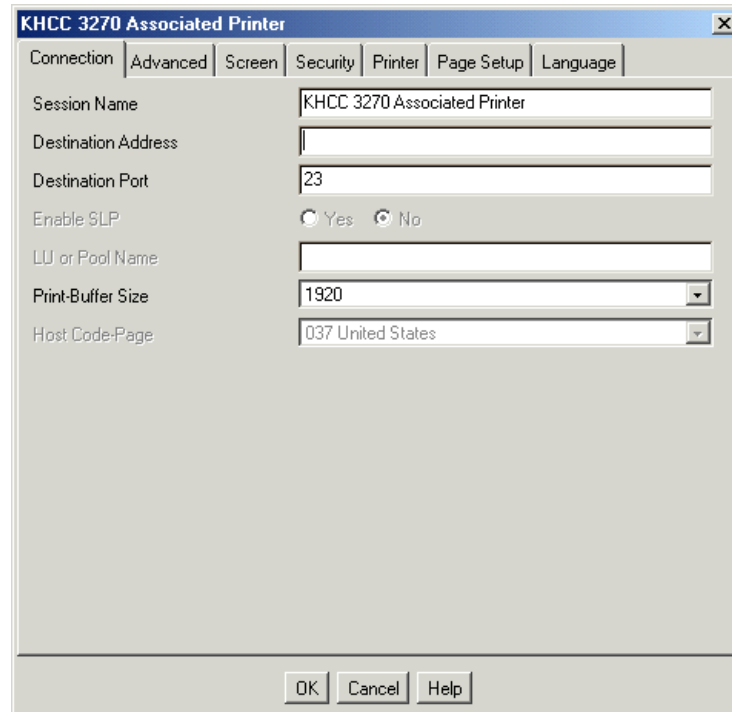


Figure 19-11 Configuring an associated printer

2. Create or modify a display session

The only difference from a normal display session configuration is that you select the printer session created in step 1 from the Associated Printer Session drop-down list, which lists the names of all your pre-configured printer sessions defined for that display type. As you can see in Figure 19-12, we chose the printer session we configured in step 1.

With the Close Printer With Session parameter, you can force the printer session to be closed when the 3270 session is closed. We recommend this, especially if you are selecting displays and printer sessions from a pool. Doing so ensures that display/printer pairs are always available.

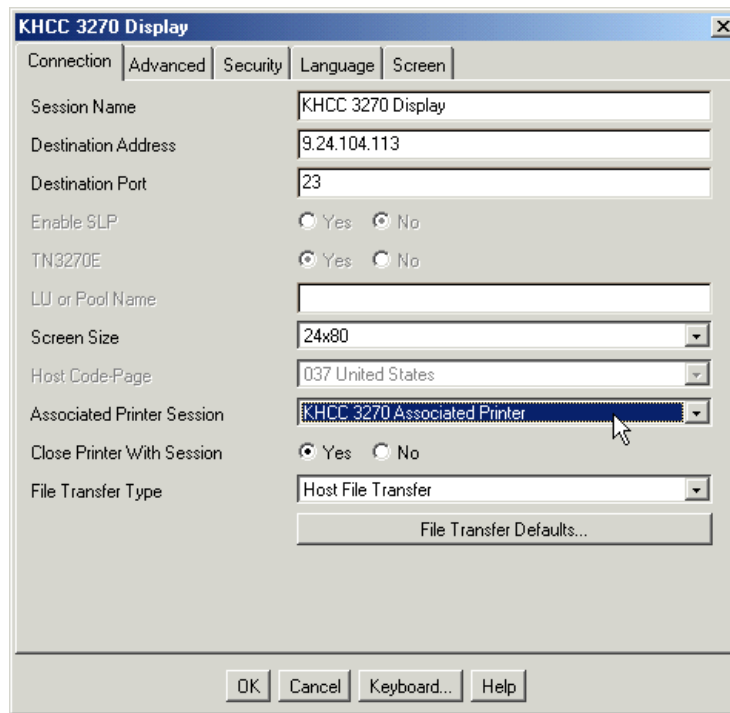


Figure 19-12 Associating a printer session with a 3270 display

When you start the display session, the associated printer session starts automatically (see Figure 19-13).

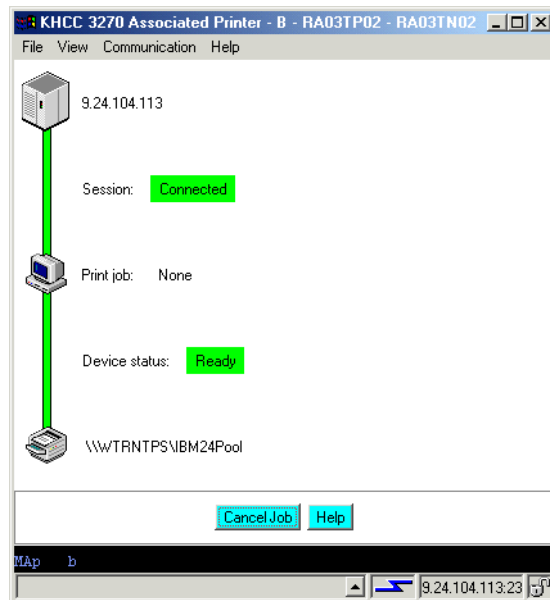


Figure 19-13 Associated printer session

The title of the printer session window includes parameters that you have configured and perhaps an ID that is generated by the system. They appear in the following order, delimited by '-':

- ▶ The session name (for example, 1 KHCC 3270 Associated Printer)
- ▶ The session ID or short name
- ▶ The printer LU name (for example, RA03TP02)
- ▶ The LU name of the associated display session, if any (for example, RA03TN02)

19.3.2 How an associated printer session works

Figure 19-14 on page 686 shows an associated printer session, and the logical connections among them.

Display LU RA03TN02 is associated with printer RA03TP02. Remember that the association itself is made in the Telnet server, not by the host application or in your emulator configuration. When you configure the display session, you just select, from the Associated Printer Session drop-down list, a printer emulator session that is going to send its output to the correct physical printer. When the display session starts, the Host On-Demand client will automatically issue a request (Telnet INIT) for a printer session providing

the LU name of the display session. The Communications Server will use the display LU name to locate the associated printer LU name. If a match is found the LU will be assigned and a session established if the LU is not currently in session. If no match is found, no printer session will be established.

The association does not mean that the host application automatically directs output to the correct LU unless it has been written to do so, because it knows nothing about the association. When you are ready to print, you must tell the host application which printer LU to which to direct the output. Although you don't necessarily know the name of the printer LU until its session starts, it is available from the printer session's title bar.

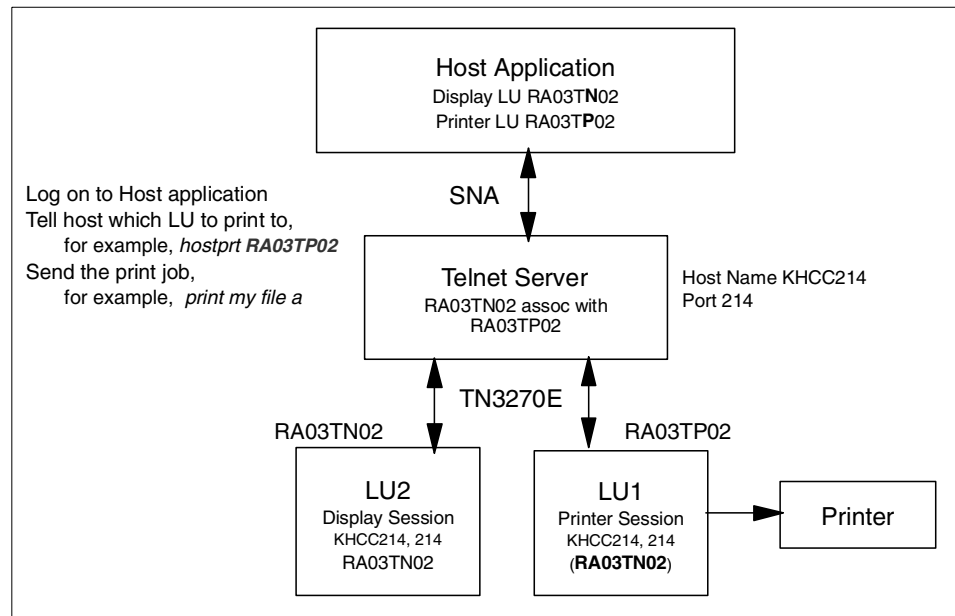


Figure 19-14 How associated display and printer sessions work

19.4 5250 printer session

A Host On-Demand 5250 printer session emulates a 3812 printer attached to an iSeries Server. Host On-Demand 5250 printing is performed using Host Print Transform (HPT).

19.4.1 Host Print Transform

Host Print Transform is an OS/400 function that converts an SNA character string (SCS) or Advanced Function Printer (AFP) data stream into an ASCII data stream. The ASCII data stream is then formatted and sent to an ASCII printer through a Host On-Demand 5250 printer session. The conversion is done on the iSeries, which provides these advantages:

- ▶ Consistent output for most ASCII printers

HPT is capable of supporting many different types of ASCII data streams. For example, it supports the Hewlett-Packard printer control language (PCL), the IBM personal printer data stream (PPDS), and the Epson FX and LQ data streams.

- ▶ 3812 SCS printer emulation

If HPT is used, all of the ASCII printers connected to an iSeries system can perform a 3812 SCS level of function.

Your printer may not support all functions. For example, you cannot print in 180-degree orientation if your printer supports only 0 and 90-degree orientations.

- ▶ Support for many different ASCII printers

Many printers, including IBM printers, support HPT.

- ▶ Customized printer support

You can add or change characteristics for a particular printer using workstation-customizing objects that come with HPT. Also, if a workstation-customizing object for a particular printer does not exist, you can create one.

- ▶ Support for the conversion of a double-byte SCS or AFP data stream into an ASCII data stream

For the AFP-to-ASCII data stream conversion, there are additional advantages such as support for AFP font, text, image and bar code commands. The following types of printers support this function:

- IBM 4019, 4029 and 4039 laser printers
- HP laser and ink jet printers
- IBM PAGES printers (DBCS)

On other printers, images or bar codes may not be supported by the AFP-to-ASCII transform function, and the text may not be positioned correctly.

How Host Print Transform works

The 5250 Host Print Transform (HPT) converts the iSeries print data stream just before it is sent from the iSeries to the printer spool file. Because the iSeries does the conversion, the host does most of the print processing instead of the workstation.

Many printers, including IBM printers, support the ASCII print-data stream. The ASCII data stream uses iSeries system objects that describe the characteristics of a particular ASCII printer. When you configure a printer session, you select the printer from the list provided.

By default, Host On-Demand uses the SCS-to-ASCII transform, but you can configure the iSeries to do an AFP-to-ASCII transform, which Host On-Demand also supports. The ASCII data stream is passed through the emulator using the SCS ASCII Transparency (ATRN) command. Host On-Demand deletes the ASCII Transparency command and passes the ASCII data stream to the workstation printer.

For more information about the Host Print Transform, refer to the iSeries Printer Device Programming documentation.

19.4.2 Configuring a 5250 printer session

There are six tabs to the 5250 printer definitions:

- ▶ Connection
- ▶ Security
- ▶ Advanced
- ▶ Proxy Server
- ▶ Printer
- ▶ Screen

Except of the Printer tab those tabs are similar to the 5250 display session tab or only a subset of the parameters of a display session. Therefore please refer to chapters “3270 and 5250 display sessions” on page 286

5250 printer Printer tab

The printer tab show in the upper portion of the window the generic printer parameters as used as well for 3270 printer session. We had pointed them out in the 3270 printer session section “3270 printer Printer tab” on page 670. Please refer to that for this part of the tab.

The Printer tab is also used to define the output device for the print data, and because 5250 printer support uses Host Print Transform (see “How Host Print Transform works” on page 688) we specify characteristics of the target printer.

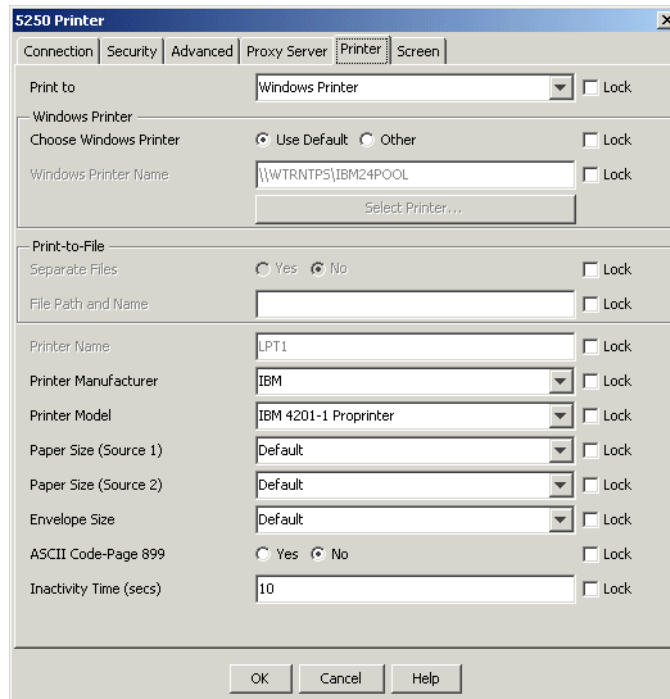


Figure 19-15 Printer tab on the 5250 printer session window

- ▶ **Printer Manufacturer (5250)**
The manufacturer of the printer that will be used for this session.
- ▶ **Printer Model (5250)**
The model of the printer that will be used for this session.
- ▶ **Paper Size (source 1) (5250)**
Specifies the size of the paper in Source 1.
- ▶ **Paper Size (source 2) (5250)**
Specifies the size of the paper in Source 2.
- ▶ **Envelope Size (5250)**
Specifies the size of the paper in the envelope feeder.
- ▶ **ASCII Code Page 899 (5250)**
Click Yes if your printer supports ASCII code-page 899. This is not resident on most printers.
- ▶ **Inactivity Time (secs) (5250)**

Specifies the amount of time to wait for printing to start. If printing does not start within the time set, an Intervention Required message pops up. The valid values are between 10 and 255 seconds. A value of 0 disables the timer and a message never appears.

The default is 10.

19.4.3 Using the 5250 printer session

Using the 5250 printer session is very similar to the 3270 printer session discussed in 19.2.2, “Using a 3270 printer session” on page 678, with the exception of the following.

Selecting **File -> JumpNext** allows the user to jump to the next session. Selecting **File -> Printer** will present the window shown in Figure 19-16 to temporarily modify printer settings. Printing of a test page, or ejecting a page from the printer is not available for 5250 printer sessions.

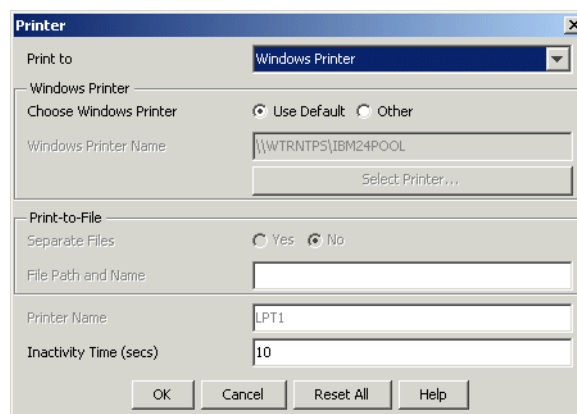


Figure 19-16 Printer settings available via the menu bar

The Print job: field on the main session window will state Printing when a print job is currently being printed. This is in contrast to the message Page nnn, which is presented for 3270 printer sessions.

19.5 3270 Host printing for DBCS

If you are the user of a DBCS language, it may be necessary to correct some files in order to print and display them with Host On-Demand. DBCS has user-defined characters (UDC), which are unique to the languages supported by Host On-Demand. This section tells you how to print and display DBCS, especially UDC.

Note: Comprehensive documentation and help is available with regard to host printing information. Please refer to the online help.

3270 Host Printing for DBCS is almost the same as for SBCS (single-byte character set). The following sections provide information about the differences between DBCS and SBCS and the points that are unique to DBCS.

When you configure a printer session in a DBCS language, you must be careful about the host code page and the Printer Definition table.

19.5.1 Host code page

Generally, the default host code page, which belongs to the default language of Host On-Demand, has already been selected by the system. In some DBCS languages, there are several code pages in the list, so you should change to the code page supported by the host system if necessary. Use the code page drop down box in the host code page tab of your printer session.

19.5.2 Printer definition tables

Host On-Demand and Personal Communications Version 5.6 provide PDTs and PDFs for each language. DBCS printers are different from SBCS and support the following modes:

- ▶ HP PCL Level 3
- ▶ IBM PPDS
- ▶ ESC/P

The list of available PDTs is determined by the host code page; only PDTs that are valid for the selected code page appear in the pull-down list. You should choose the PDT that is correct for the printer you will use.

The PDT lists for DBCS languages are:

- 930 Japan (Katakana Extended) and 939 Japan (Latin Extended)
 - ASCII text mode
 - Printers based on ESC/P 24-J84

- IBM 5577-B02, F02, G02, H02
- IBM 5585-H01 Printer
- IBM 5587 G01, H01 (without advanced function)
- Lips3a4 Printer
- Lips3b4 Printer

933 Korea (Extended)

- Korea IBM 5577 Printer
- Ks_jo Printer
- Ks_wan Printer
- Kssm_jo Printer
- Kssm_wan Printer

935 PRC (Simplified Chinese Extended)

- Simplified Chinese ESC/P Printer

937 ROC (Traditional Chinese Extended)

- Traditional Chinese ESC/P Printer (5550)
- Traditional Chinese ESC/P Printer (Big-5)
- Traditional Chinese ESC/P Printer (cns)
- Traditional Chinese ESC/P Printer (tca)
- Traditional Chinese IBM 5577 Printer
- Traditional Chinese IBM 5585 Printer

If you want to use a PDT other than one in the list, for example a SBCS PDT, you must put the appropriate PDF in the directory \pdfpd\usrpdf with a different name, then compile it. This creates a unique PDT, which will appear in the PDT pull-down list on the configuration window.

19.5.3 Font image file

When you print user-defined characters (UDC), a UDC font image file is needed; you must create it and copy it to the \ondemand\Hod\fonts directory on a server, and to the \ondemand\lib\fonts directory on a locally installed client.

For host printing support of User-Defined Characters (UDCs), in either Java file interface mode or Windows spooler interface mode, you must prepare a UDC font-image file. On a server, this file must be located in the \hostondemand\hod\fonts\ directory so that it is accessible to clients. In Windows native printing mode, you do not have to do this as long as UDCs are defined on your Windows system.

UDC support is applicable to double-byte languages only.

On Windows, you must run the *w32udcnv.exe* utility to find and convert Windows user-defined fonts into a usable font-image file. The utility is provided in the \udc directory on the Host On-Demand CD. It is not copied to the server during installation. To use the utility:

- ▶ Run **w32udcnv.exe**.
- ▶ Click **Convert** to start the conversion and generate a font-image file.

After conversion, the font-image file is saved in the C: drive's root and is named according to the language of the operating system you are running, as shown in Table 19-1.

Table 19-1 Font-image file names

Platform	Font-image filename
Japanese Windows	jpn24.fnt
Korean Windows	kor24.fnt
Simplified Chinese Windows	chs24.fnt
Traditional Chinese Windows	cht24.fnt

Copy the font-image file to the \hostondemand\hod\fonts directory.

On an OS/2 server, copy the OS/2 font-image file, \$SYS1Z24.FNT, to the \hostondemand\hod\fonts directory and rename it according to Table 19-1.

Limitations

In a Traditional Chinese Windows environment, there are 13 more UDCs than IBM's Big-5 UDCs. Therefore, if the last 13 UDCs are defined in the range of 0xC8F2-0xC8FE, they are ignored by the utility and cannot be used.

In a Korean Windows environment, only PC code page 949 is supported. You can define and print 188 UDCs in the ranges of 0xC9A1-0xC0FE and 0xFEAF-0xFEFE.

19.6 VT Host printing

Host On-Demand provides VT host printing in much the same manner as a VT 420 terminal would provide this service to a VT host. You can print host application files on a printer that is directly attached to your workstation or to a network printer.

In a traditional VT host print session, the host determines when a print operation is desired (typically triggered by a user's action). In a VT print session, the same host-terminal connection used for screen-based information is used to pass information to the printer. Once the host determines that a print job is to start, a terminal sequence is sent by the host to the terminal to initiate the desired print operation. The host sends all desired print information and then ends the process by sending a print termination sequence to the terminal. In some cases, both the screen and the printer may be the simultaneous destination of the information sent by the host.

Host On-Demand supports all of the VT 420 defined print sequences; however, due to limitations in the Java environment, not all printer status commands are supported. The Windows spooler interface mode and the Windows native printing modes are not supported on VT host print sessions.

More information on VT Host Printing is contained in the *DEC VT220 Programmer Reference Manual*.

19.7 Adobe PDF printing

The new option of HOD 7 to print 3270 print jobs as Adobe PDF files to disk has the following features and limitations:

- ▶ Features:
 - No need to have Adobe Acrobat installed for creating the PDF files
 - Can be browsed and printed with Adobe Acrobat Reader or Acrobat Reader plug-in for browser
 - Unlike a plain text file, Adobe PDF file keeps all formatting information (CPI, LPI, etc.)
- ▶ Limitations
 - 3270 Printer Session Only (no 5250, no Print Screen support)
 - Limited font support
 - For U.S. English, “Courier”, “Courier New” and “Lucid Console” are supported
 - Fixed top/bottom/left/right margins
 - 1/4 inch for North American forms (Letter, etc.)
 - 0.5 cm for ISO forms
 - Types of form are also predefined
- ▶ NLS considerations

- Non-Latin 1 SBCS languages (including Bidi, Thai)
Embed font file in PDF
Makes file size bigger (100-200K), but runs on all platforms
- Or Use Windows font
File size is smaller, but generated file can be browsed only on Windows
- DBCS languages
Use DBCS fonts provided by Adobe
<http://www.adobe.com/products/acrobat/acrrasianfontpack.html>
DBCS characters might be browsed without those fonts, though

19.8 Host Print Java Beans

Host on Demand Version 7 has added two new beans for host printing:

- ▶ HostPrintSession
- ▶ HostPrintTerminal
- ▶ Both beans are available for 3270 and 5250 only. It does not support VT. Both support Java1 (JDK 1.1.8) and Java2 (JDK 1.3.0)

Relationship between Session, HostPrintSession, HostPrintTerminal:

- ▶ HostPrintSession is an extension of the session bean.
- ▶ HostPrintSession is a session
- ▶ HostPrintTerminal is a visual component that contains the HostPrintSession and thus the session.
- ▶ HostPrintTerminal has a HostPrintSession that again is a session

How to access HOD beans:

There are several jar files that come with the toolkit. The following jar files contain all available HOD beans:

- ▶ habeans.jar and habeansnlv.jar for Java1
- ▶ habeans2.jar and habeansnlv2.jar for Java2

There are two ways to utilize beans within these jar files:

- ▶ via a visual builder tool that comes with any modern IDE like VisualAge for Java, JBuilder, Visual Cafe, etc.
- ▶ writing the code manually



Macro support and enhancements

Previous releases of Host On-Demand supported the creation of Macros to perform certain automated or repetitive tasks during the life of a host session. In this chapter we cover basic Macro support and the Macro enhancements added in version 7 of Host On-Demand.

20.1 Macros

The Macro feature of Host On-Demand allows you to build scripts (Macros) consisting of command sequences that perform actions on the host. If you regularly do the same task when you work with a host system, you can record your keystrokes and the host's reactions, save them, and then play the Macro whenever you need to perform the same task. This is an ideal way to store frequently used actions for repeated use.

Macros are stored using XML script and are stored with the icon that launched the session. Multiple different Macros may be recorded and saved for a Host On-Demand session. However, remember that each Macro is downloaded when you start the session to which it applies.

20.2 Creating and Editing Macros

Macros may be recorded in two ways (see Figure 20-1):

1. Using the **Actions-> Record Macro...** pull-down on the menu bar.
2. Using the **Record macro** button on the Macro Manager toolbar. The Macro Manager gives you advanced Macro editing capability including:
 - Smart waits that cause a macro to wait during playback until it recognizes a screen according to set conditions.
 - Prompts that allow you to type in information that varies or which you do not want displayed during playback.
 - Data extraction that will retrieve data from a host application and put it into an applet or graphical user interface by using the MacroExtractEvent method of HACL. A Macro with data extraction in it is particularly useful for application developers who want to retrieve data from the host without knowing the structure. The data extraction function has no significance with regard to playback through a Host On-Demand emulator.

20.2.1 Recording a simple Macro

Let's record a Macro that simply logs onto your host VM session and opens a list of files.

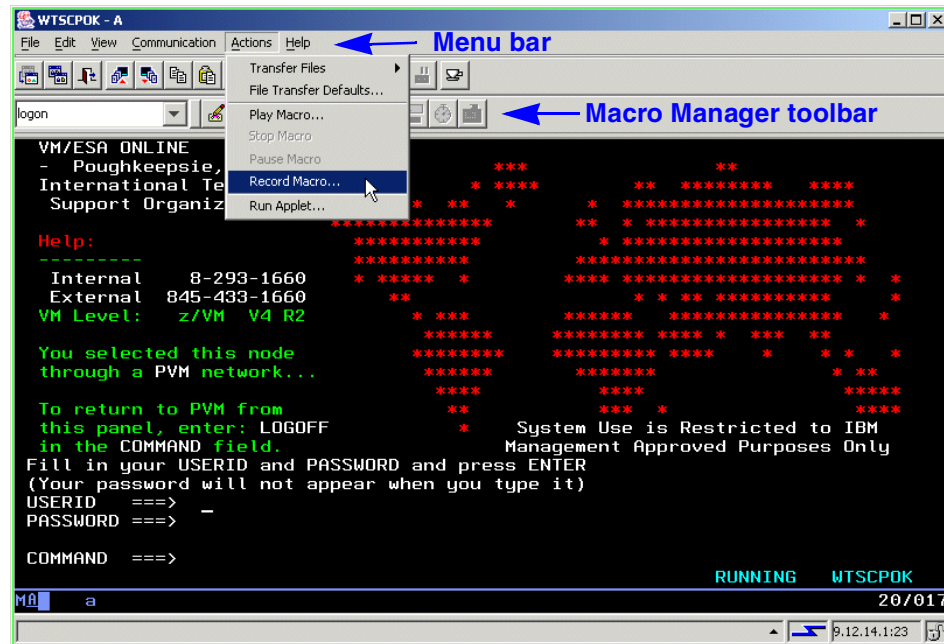


Figure 20-1 Begin recording a Macro

- Click **Actions > Record Macro**. See Figure 20-1. The following window will be displayed.

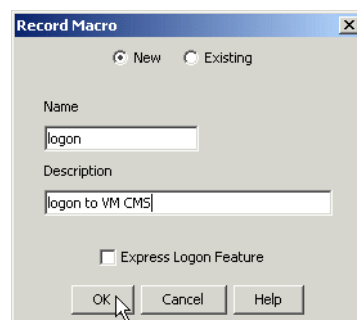


Figure 20-2 Record macro window

- Select **New** and enter a name to identify the Macro for later use. Optionally, type a description. This is useful when you have more than one Macro; it can help to remind you what the Macro is used for. Select Express Logon Feature if you are recording a Macro to use the Express Logon feature. To use this feature, the session must be an SSL session and using client authentication. The Express Logon option allows you to use the client certificate for obtaining

the user ID and password. It requires additional configuration on the telnet servers. Refer to 11.8, “Express Logon Feature” on page 455 for more details. Click **OK**.

- ▶ You can insert one or more prompts in a Macro so that during playback, you can type information that varies or that you do not want to have displayed. During playback, a window opens so that you can enter the information which is then sent to the host application. By default, you are asked to respond to all the prompts in a single window when playback starts, but a check box in the Edit panel lets you ask to have the prompts appear at the appropriate places in sequence. We will add a prompt to the Macro requesting a VM userid.

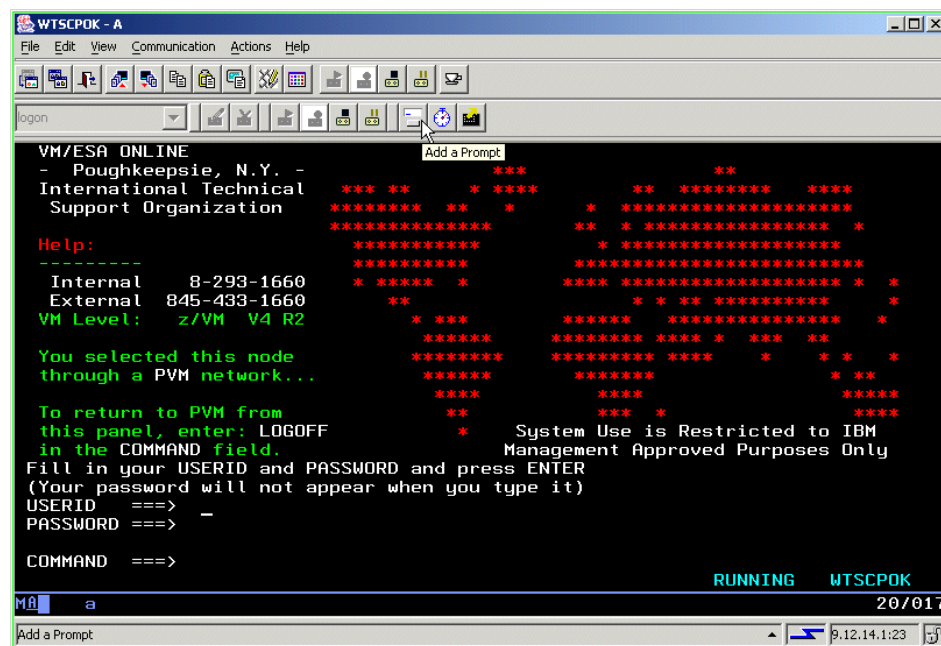


Figure 20-3 Adding a prompt to a Macro

- ▶ Click the **Add a Prompt** button on the Macro Manager toolbar to add a prompt. See Figure 20-3.
- ▶ The prompt panel shown on the left side of Figure 20-4 will be displayed. There are several points worth mentioning about the prompt panels shown in Figure 20-4.
 - The Row and Column fields indicate the position of the cursor when you clicked the Prompt button during recording. Your response to the prompt during playback will be entered at that position unless you change it. Unless there is a good reason, you should not do so because the host application will probably fail.

- If your response to the prompt will often be the same, you might want to enter a Default Value; you will still be prompted during playback and will be able to change the value, but you will often need only to click **OK**.
- If you check the **Is it a Password?** box, your response to the prompt will be displayed as asterisks and will be encrypted when the Macro is saved. Of course, this is not restricted to passwords.
- Some host applications put data into fields automatically, which means that a field that you are expecting to fill in from a prompt may already have something in it. If such a field is not cleared, invalid characters might be added to those that you enter. If you check the **Clear Host Field** box, anything already in the field is removed before you are prompted.

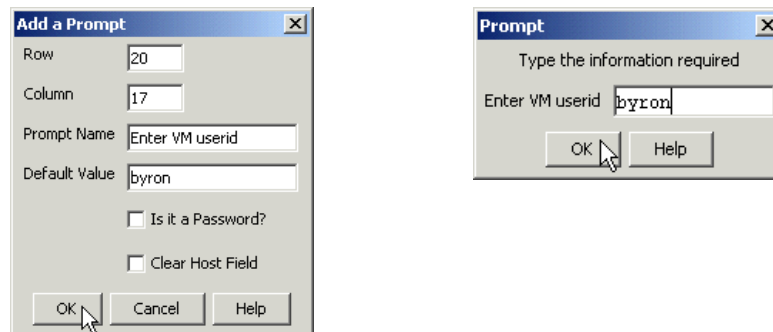


Figure 20-4 Adding a prompt to a Macro

- ▶ The prompt panel shown on the right side of Figure 20-4 will be displayed when the Macro is run.
- ▶ Continue in the host session performing the tasks you want to record. Every key you press is recorded as part of the Macro. To press keys you don't want to be included in the Macro, click **Pause**. When you have finished, click **Pause** again to continue. If you enter the wrong data while recording a Macro, you cannot go back to make corrections. You can, however, record over the existing Macro or edit the Macro code to make changes.
- ▶ When your task is complete, click **Stop**. Recording stops and the Macro is saved. Macros are recorded using XML script (beginning in Version 4 of Host On-Demand). To make changes to the Macro use the Macro Manager (see 20.2.2, "Adding advanced functions to the simple Macro" on page 703). You can edit previous versions of Host On-Demand Macros using the Macro Editor. However, once you open a V3 Macro into the Macro Manager or Macro Editor, it is converted to the XML format. It cannot be converted back to the V3 format.

When we are finished the, the Macro looks like this:

Example 20-1 Simple Host On-Demand Macro

```

<HAScript name="logon" description="logon to VM CMS" timeout="60000" pausetime="300"
promptall="true" author="" creationdate="" suppressclearevents="false" usevars="false" >

<comment>
    Definition of the first screen. This is the initial logon screen.
</comment>
    <screen name="Screen1" entryscreen="true" exitsscreen="false" transient="false">
        <description>
            <oa status="NOTINHIBITED" optional="false" invertmatch="false" />
        </description>
        <actions>
            <prompt name="Enter VM userid" description="" row="20" col="17" len="8"
default="byron" clearfield="false" encrypted="false" movecursor="true" xlatehostkeys="true"
assigntovar="" varupdateonly="false" />
            <input value="[tab]jnglrot[enter]" row="0" col="0" movecursor="true"
xlatehostkeys="true" encrypted="false" />
        </actions>
        <nextscreens timeout="0" >
            <nextscreen name="Screen2" />
        </nextscreens>
    </screen>

    <screen name="Screen2" entryscreen="false" exitsscreen="false" transient="false">
        <description>
            <oa status="NOTINHIBITED" optional="false" invertmatch="false" />
            <numfields number="9" optional="false" invertmatch="false" />
            <numinputfields number="1" optional="false" invertmatch="false" />
        </description>
        <actions>
            <input value="[clear]" row="0" col="0" movecursor="true" xlatehostkeys="true"
encrypted="false" />
        </actions>
        <nextscreens timeout="0" >
            <nextscreen name="Screen3" />
        </nextscreens>
    </screen>

    <screen name="Screen3" entryscreen="false" exitsscreen="false" transient="false">
        <description>
            <oa status="NOTINHIBITED" optional="false" invertmatch="false" />
            <numfields number="5" optional="false" invertmatch="false" />
            <numinputfields number="1" optional="false" invertmatch="false" />
        </description>
        <actions>
            <input value="[clear]" row="0" col="0" movecursor="true" xlatehostkeys="true"
encrypted="false" />
        </actions>

```

```

    <nextscreens timeout="0" >
      <nextscreen name="Screen4" />
    </nextscreens>
  </screen>

  <comment>
    Definition of the last screen. It executes the fulist CMS command.
  </comment>
  <screen name="Screen4" entryscreen="false" exitsscreen="true" transient="false">
    <description>
      <oia status="NOTINHIBITED" optional="false" invertmatch="false" />
      <numfields number="5" optional="false" invertmatch="false" />
      <numinputfields number="1" optional="false" invertmatch="false" />
    </description>
    <actions>
      <input value="fulist[enter]" row="0" col="0" movecursor="true" xlatehostkeys="true"
encrypted="false" />
    </actions>
    <nextscreens timeout="0" >
      </nextscreens>
    </screen>

  </HAScript>

```

20.2.2 Adding advanced functions to the simple Macro

The Macro Editor allows you to expand a simple Macro to include advanced functions available in Host On-Demand V7.0 such as:

- ▶ Variables
- ▶ Conditional if-else logic
- ▶ Macro Chaining
- ▶ Run programs from a Macro

All of these features of Macros are documented in the Host Access Beans documentation installed with the Host On-Demand Toolkit. *Appendix A. Macro Script Syntax* contains specifics on all the Macro commands and extensive examples of sophisticated Macros. The InfoCenter contains a tutorial on using the Macro Manager. The Host On-Demand InfoCenter is located at:

<http://www-3.ibm.com/software/webservers/hostondemand/library/infocentergafinal/hod/en/help/2tabcontents.html>

Important: To enable the advanced Macro features, you must check the **Use Variables and Arithmetic Expressions in Macro** box on the main Macro Editor panel. See Figure 20-5.

Updating the Macro

We now return to the Host On-Demand session used to record the simple Macro and open the Macro Editor. The simple Macro recorded earlier will be enhanced in the following ways:

- ▶ Define two variables to be used in the Macro processing.
- ▶ Assign the userid to one of the variables.
- ▶ Execute an external program (Notepad) passing one of the variables to it. See Figure 20-8 on page 708.
- ▶ Read data from screen and display a message based on a conditional test of its value. See Figure 20-10 on page 710.

We invoked the Macro editor from the Macro Manager toolbar to make these updates to the Macro. See Figure 20-5

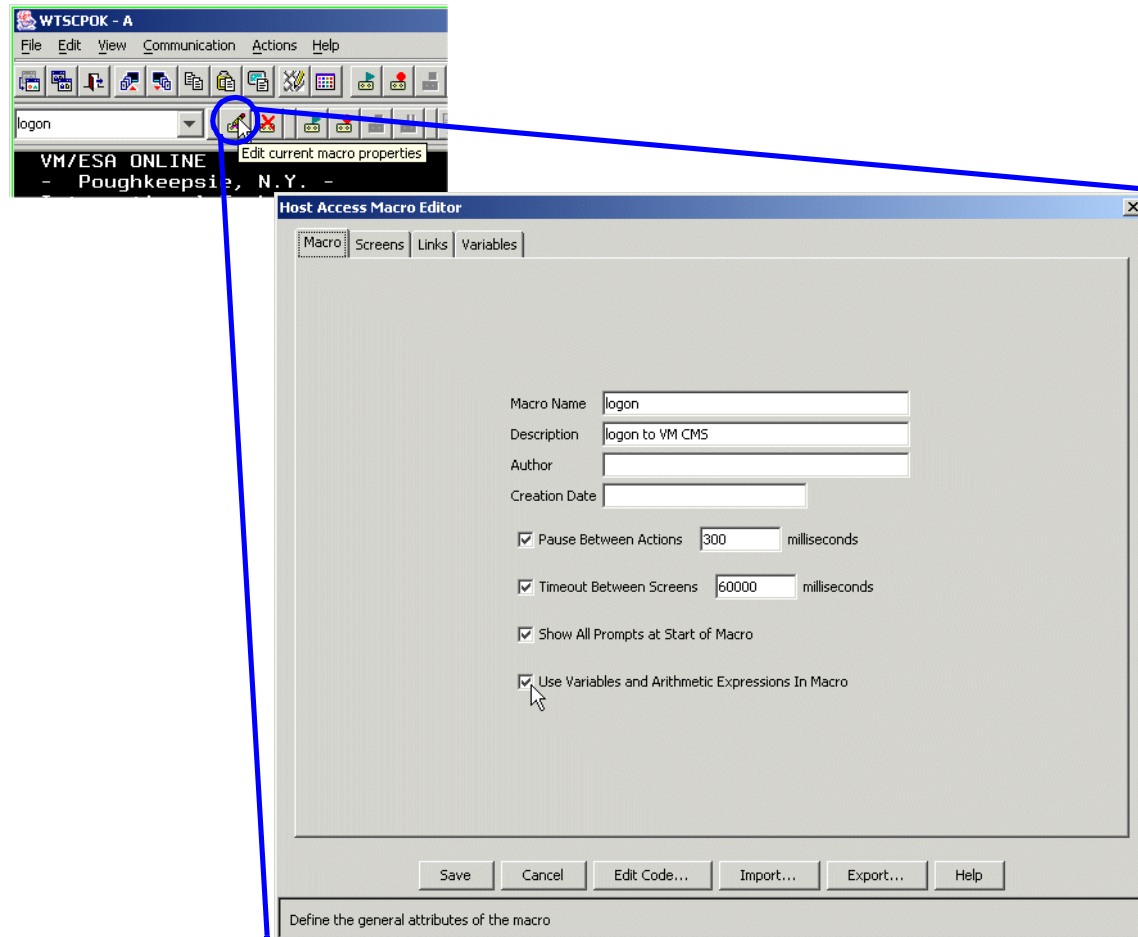


Figure 20-5 Invoking the Macro Manager

Our first task is to define two variables that we will use in the Macro. Selecting the Variables tab shown in Figure 20-5 displays the window shown in Figure 20-6.

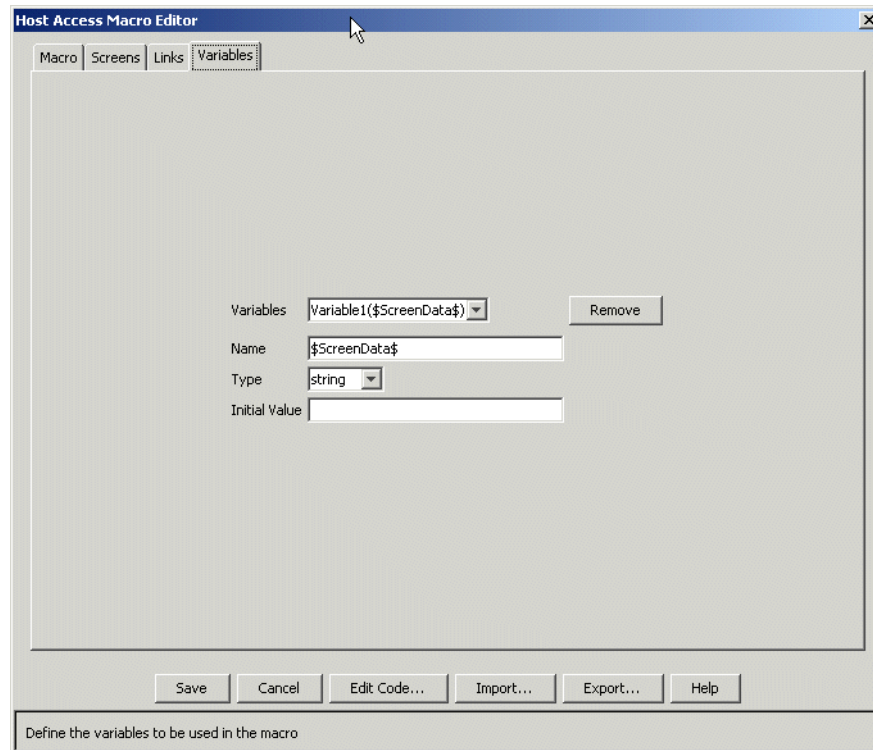


Figure 20-6 Defining Macro variables

We used this window to define two variables (\$ScreenData\$ and \$userid\$) that will be used later on in the Macro.

The remaining additional functions were added to the simple Macro by accessing the Actions panel on the appropriate screen.

First, we assign the userid entered in the Macro prompt (see Figure 20-4 on page 701) to the variable \$userid\$. Note from the Macro listing in Example 20-1 on page 702 that the Macro prompt action is on Screen1. Select the Screen tab on the Macro Editor panel to see the screen selections, then select Screen1.

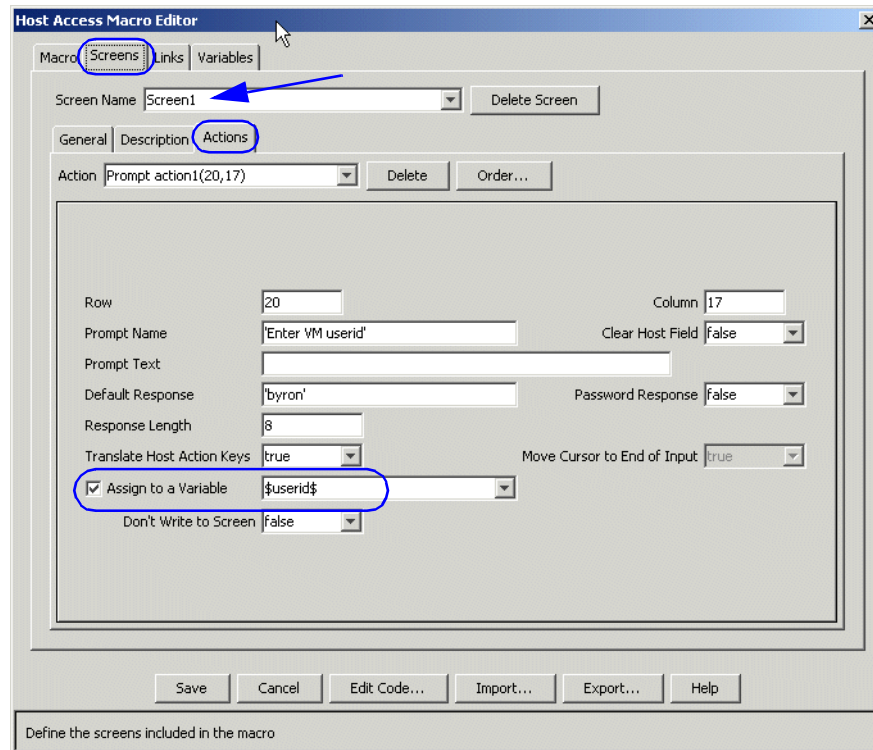


Figure 20-7 Macro screens tab displaying Screen1

There are several items to note on Figure 20-7. The Macro screen to be manipulated is selected in the **Screen Name** field. We want to modify the actions taken on this screen, so we must select the **Actions** tab. Note the **Prompt Name** and **Default Response** fields. These fields were filled in when we initially recorded our Macro. To assign the value entered for VM userid in the prompt to variable \$userid\$, all we must do is check the **Assign to a Variable** box and type in the variable name.

Next, we want to execute an external program using the variable we just captured. For our example, we will execute the Windows Notepad program passing it a filename of "\$userid\$.txt" to be edited. We chose to perform this action from Screen3, however it could have been performed from any of the screens in the Macro. Select Screen3 in the **Screen Name** field.

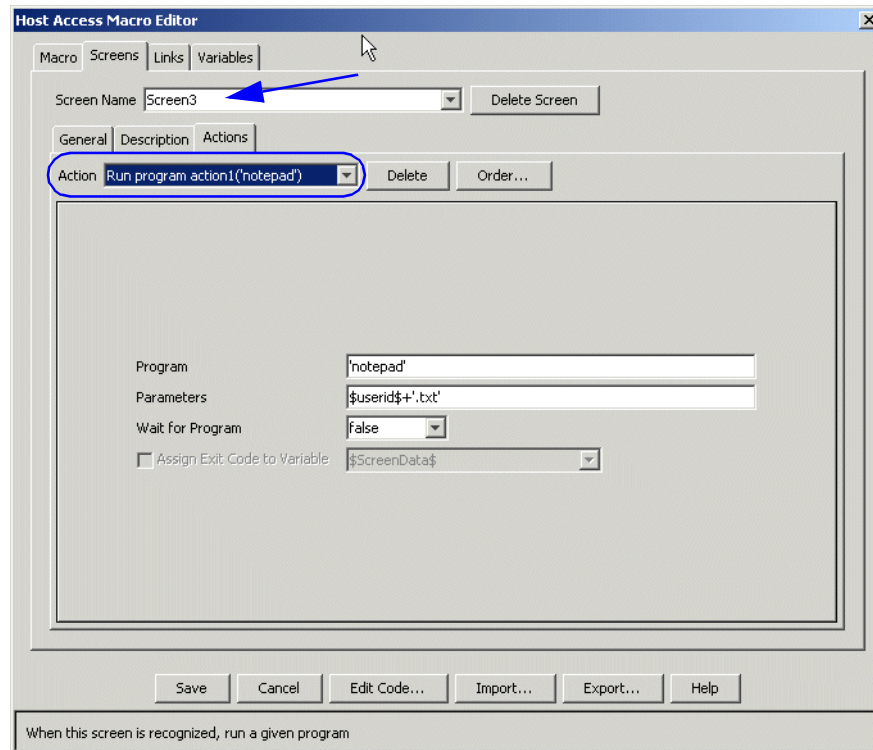


Figure 20-8 Execute Notepad from a Macro

In Figure 20-8, note that we have selected Screen3 to modify. In addition, we have selected “Run program action” from the Action pulldown list. When the “Run program action” is selected, the fields on the lower half of the panel are automatically displayed. Note that we selected “Notepad” as the program to be executed and “\$userid\$.txt” as the parameters to be passed to the program. When the Macro reaches Screen3 in its execution, a Notepad window will open up while the Macro continues execution.

Our last addition to the Macro is to read some data from the screen and display a conditional message based on the data read from the screen. We choose to perform these actions based on data read from Screen4. Note that we must perform two actions on Screen4:

1. Read (extract) data from the screen and assign it to a variable (\$ScreenData\$).
2. Display a message based on what data was read from the screen.

Select Screen4 in the **Screen Name** field.

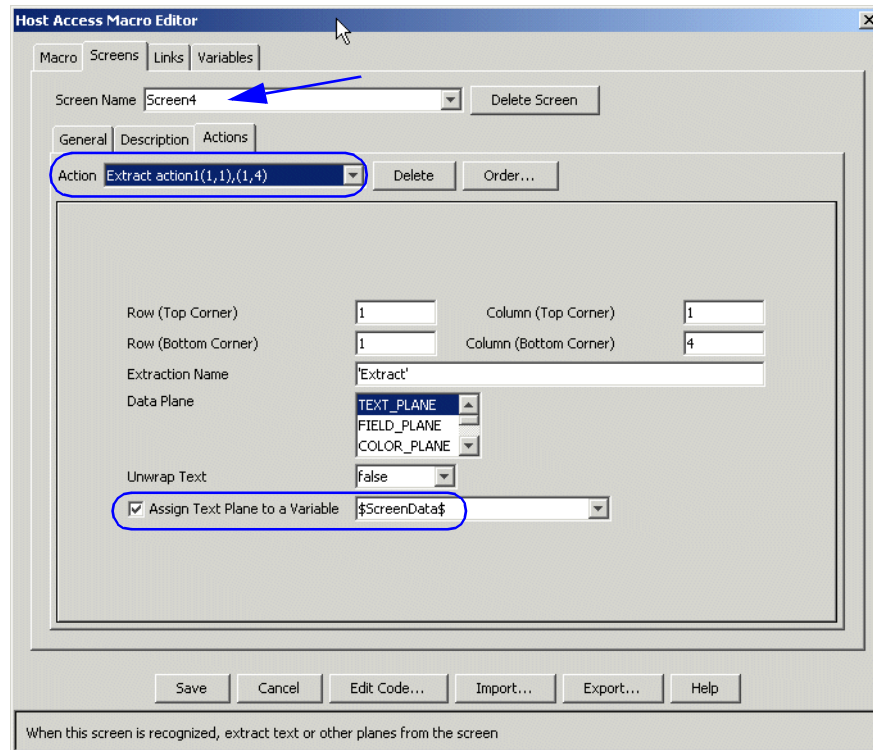


Figure 20-9 Extracting data from Screen4

The first action we perform on Screen4 is to extract a string of data from the screen and assign it to variable `$ScreenData$`. This is all easily performed with one Extract action as shown in Figure 20-9.

Next, we perform a conditional checking action on the data to decide which message to display.

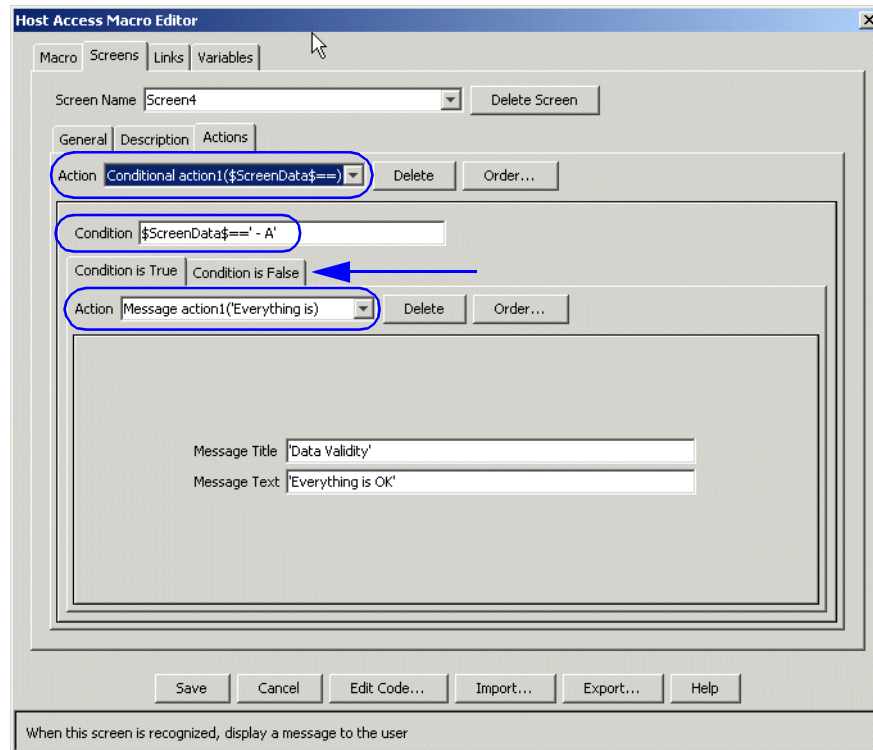


Figure 20-10 Conditional checking and actions in the Macro

Everything required for the conditional checking and action to be taken is defined on this Conditional action window. Note that the **Action** is “Conditional action” and the condition to be checked is defined in the **Condition** field. Two conditional results must be defined (“True” and “False”), and an action to be taken for each result must be defined (second **Action** field on the panel). The fields on the lower half of the panel change based on what **Action** and **Condition** are selected.

Here is a listing for the updated Macro. Compare it with the Macro listed on Example 20-1 on page 702.

Example 20-2 Simple HOD Macro enhanced

```
<HAScript name="logon" description="logon to VM CMS" timeout="60000" pausetime="300"
promptall="true" author="" creationdate="" suppressclearevents="false" usevars="true" >
```

```
<vars>
  <create name="$ScreenData$" type="string" value="" />
  <create name="$userid$" type="string" value="" />
</vars>
```

<comment>

Definition of the first screen. This is the initial logon screen.

</comment>

```
<screen name="Screen1" entryscreen="true" exitsscreen="false" transient="false">
  <description>
    <oia status="NOTINHIBITED" optional="false" invertmatch="false" />
  </description>
  <actions>
    <prompt name="&apos;Enter VM userid&apos;" description="" row="20" col="17" len="8"
default="&apos;jnglrot&apos;" clearfield="false" encrypted="false" movecursor="true"
xlatehostkeys="true" assigntovar="$userid$" varupdateonly="false" />
    <input value="&apos;[tab]by07on[enter]&apos;" row="0" col="0" movecursor="true"
xlatehostkeys="true" encrypted="false" />
  </actions>
  <nextscreens timeout="0" >
    <nextscreen name="Screen2" />
  </nextscreens>
</screen>
```

```
<screen name="Screen2" entryscreen="false" exitsscreen="false" transient="false">
  <description>
    <oia status="NOTINHIBITED" optional="false" invertmatch="false" />
    <numfields number="9" optional="false" invertmatch="false" />
    <numinputfields number="1" optional="false" invertmatch="false" />
  </description>
  <actions>
    <pause value="500" />
    <input value="&apos;[clear]&apos;" row="0" col="0" movecursor="true"
xlatehostkeys="true" encrypted="false" />
  </actions>
  <nextscreens timeout="0" >
    <nextscreen name="Screen3" />
  </nextscreens>
</screen>
```

```
<screen name="Screen3" entryscreen="false" exitsscreen="false" transient="false">
  <description>
    <oia status="NOTINHIBITED" optional="false" invertmatch="false" />
    <numfields number="5" optional="false" invertmatch="false" />
    <numinputfields number="1" optional="false" invertmatch="false" />
  </description>
  <actions>
    <pause value="500" />
    <input value="&apos;[clear]&apos;" row="0" col="0" movecursor="true"
xlatehostkeys="true" encrypted="false" />
    <runprogram exe="&apos;notepad&apos;" param="$userid$&apos;.txt&apos;"
wait="false" assignexitvalue="" />
  </actions>
  <nextscreens timeout="0" >
```

```

        <nextscreen name="Screen4" />
    </nextscreens>
</screen>

<comment>
    Definition of the last screen. It executes the fulist CMS command.
</comment>
<screen name="Screen4" entryscreen="false" exitsscreen="true" transient="false">
    <description>
        <oia status="NOTINHIBITED" optional="false" invertmatch="false" />
        <numfields number="5" optional="false" invertmatch="false" />
        <numinputfields number="1" optional="false" invertmatch="false" />
    </description>
    <actions>
        <pause value="500" />
        <input value="&apos;fulist[enter]&apos;" row="0" col="0" movecursor="true"
xlatehostkeys="true" encrypted="false" />
        <extract name="&apos;Extract&apos;" planetype="TEXT_PLANE" srow="1" scol="1"
erow="1" ecol="4" unwrap="false" assigntovar="$ScreenData$" />
        <message title="&apos;data read&apos;" value="$ScreenData$" />
        <if condition="$ScreenData$==&apos; - A&apos;" >
            <message title="&apos;Data Validity&apos;" value="&apos;Everything is
OK&apos;" />
        </if>
        <else>
            <message title="&apos;Data Validity&apos;" value="&apos;Everything is
BAD&apos;" />
        </else>
    </actions>
    <nextscreens timeout="0" >
</nextscreens>
</screen>

</HAScript>

```

20.3 HOD Administrators and Macros

User access to Host On-Demand functions is controlled by the Host On-Demand administrator. The administrator determines which HOD groups, users or emulator sessions are enabled or disabled for the appropriate Host On-Demand functions. The Host On-Demand administrative interface is generally the way user access to Macro functionality is set.

Note: You can use a Macro with every session that is launched from the same icon but not with sessions launched from other icons (unless they are copies of the original session made after the Macro was recorded).

You can save as many Macros as you want, but remember that if your configuration model is to save preferences on the HOD Server, then all the macros will be downloaded when you start the session in which they were recorded.

20.3.1 Controlling Access to HOD Macros

By default all users have access to all Host On-Demand Macro functions. The Host On-Demand administrator can disable various HOD functions by selecting **Users/Groups** then right clicking on the group, user or session for which they wish to limit access to HOD functions. Refer to Figure 20-11.

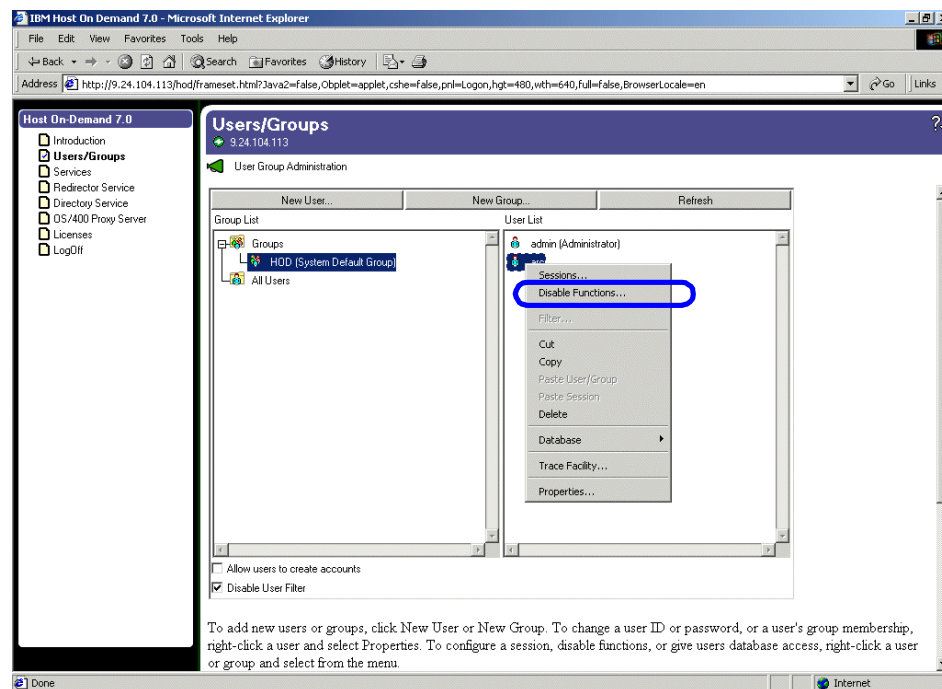


Figure 20-11 Disabling access to HOD Functions

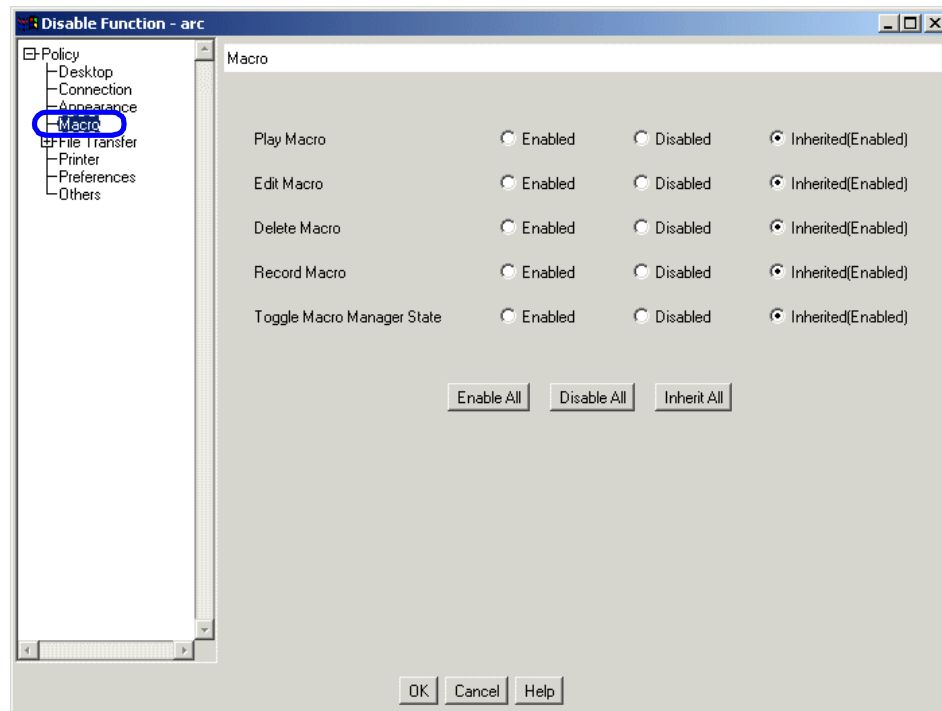


Figure 20-12 Disabling specific Macro functions

On the Disable Function panel, select Macro from the selection tree on the left panel, then enable or disable the various Macro functions as required. Refer to Figure 20-12.

20.3.2 Automatically Starting Macros

The administrator, or a user with appropriate authorities, can set the properties on a HOD session to automatically execute a Macro when the session is started. Configuring a session to AutoStart a Macro is done by right mouse clicking on the HOD session and selecting **Properties**. Next, select the Advanced property page for the session and enter the name of the HOD Macro you want to Autostart in the AutoStart Applet/Macro options field as shown in Figure 20-13.

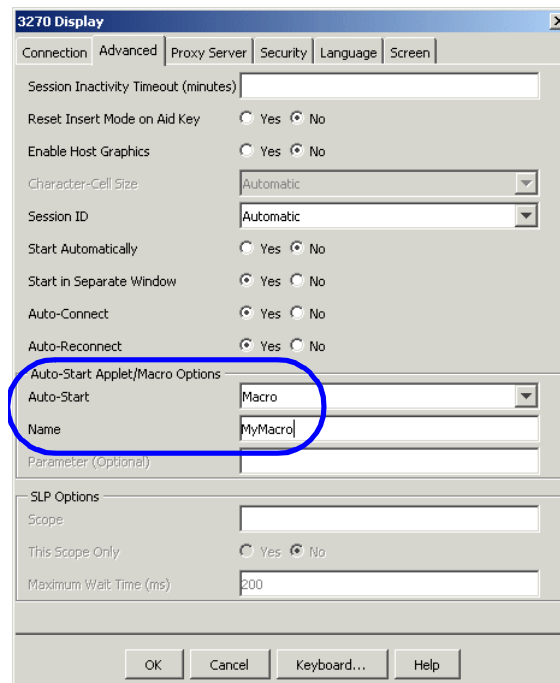


Figure 20-13 Setting a Macro to auto-start with the session

20.4 Deploying Macros

The following sections describe the process of deploying Macros based on the Deployment Wizard model used to create the session.

20.4.1 Setup Macros for Configuration Server or Combined Model

The Host On-Demand administrator may record and assign a Macro to a session or userid. The Macro is saved in the user's account file (HOD.user_id).user. Host On-Demand Administrators must start the emulator session from the Admin Console, create a new Macro or import an existing Macro and save the Macro under the session icon. The Administrator should use the HOD Admin client that has Start Session Enabled.

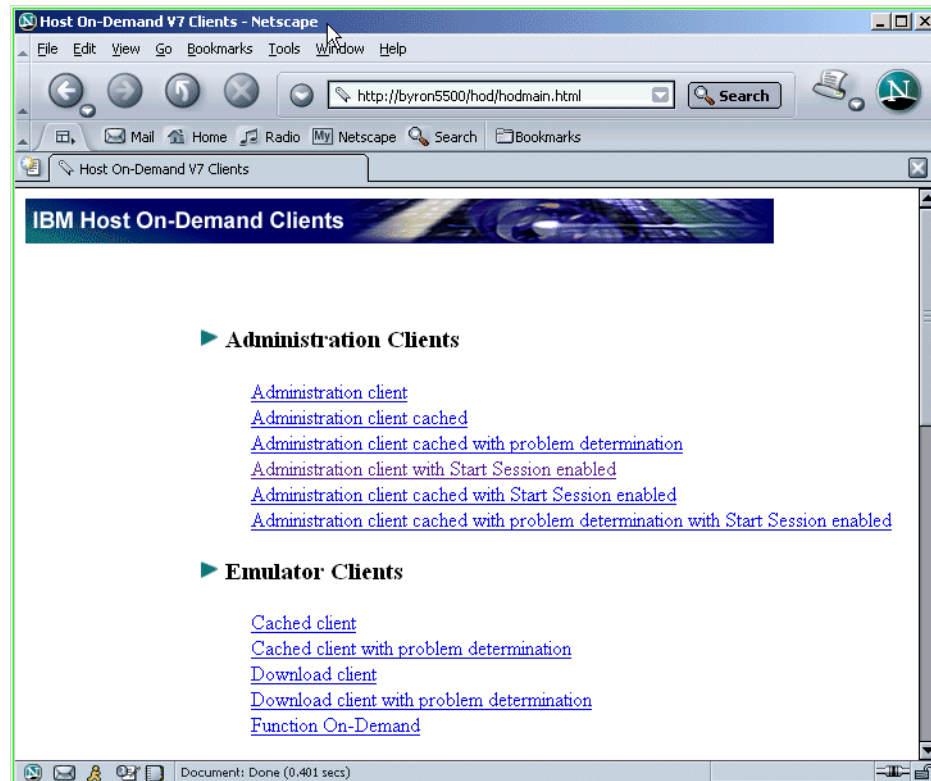


Figure 20-14 Administrator setting up a global Macro

Once you login using the correct Administration client, you can start the session and record a Macro.

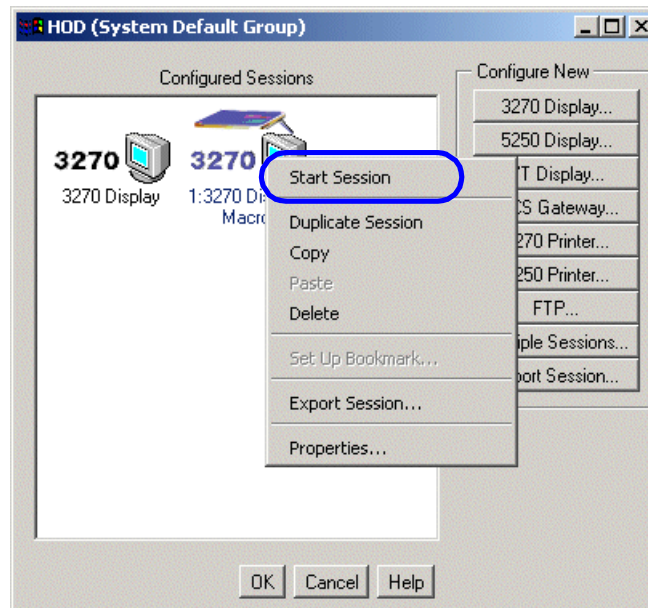


Figure 20-15 Recording a Macro for a specific session

Right click on the session icon and then select Start the session. Once the session is started you simply record the Macro. See Figure 20-15.

20.4.2 Deploying Macros for Configuration Server Model

Tip: If your users are not allowed to save individual preferences, then users will not be able to save any Macros.

It is quite probable that at some point after running HOD for a bit, you find some user has built a sophisticated Macro that would be useful for other users. To make a macro written by a user to other HOD users is quite simple.

- Since you are using the Configuration Server model, all the user configuration information including Macros is stored on the HOD server
 - In this situation the HOD administrator may choose to logon to the HOD server using the userid/password of the user with the interesting Macro. Once logged on, use the Macro Manager to export the Macro to a file.
 - If you do not want to logon to the users HOD session, then ask the user to export the Macro using the MacroManager interface.

- ▶ With the Macro now in hand you can deploy the Macro to other users, see Chapter 20.4.1, “Setup Macros for Configuration Server or Combined Model” on page 716.

20.4.3 Setup Macros for HTML based server model

The Host On-Demand Administrator can very easily deploy Macros when using the Deployment Wizard to build client code based on the HTML model. Since this model does not store user configuration information on the HOD server the Macros must be defined when:

- ▶ Running the Deployment Wizard

or

- ▶ Exporting the Macro, sending to users and asking them to import the Macro

Using the Deployment Wizard, on the screen that allows you to add the host session type, select Start Session. Once the session is started, simply record the Macro or import an existing Macro. Finish running the Deployment Wizard and once the files are deployed on the web server all users accessing them will be able to use the Macro(s) you have defined.

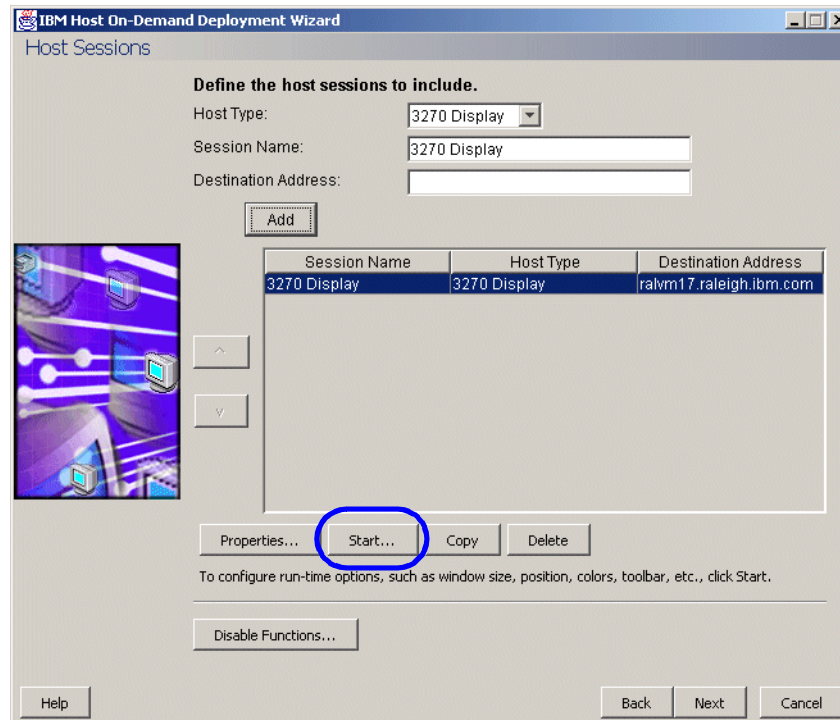


Figure 20-16 Deployment Wizard and starting session to add a Macro

20.4.4 Deploying Macros for HTML Model and Combined Model

It is quite probable that at some point after running HOD for a bit, you find some user has built a sophisticated Macro that would be useful for other users. Have the user with the interesting Macro use the MacroManager to export the Macro to a file.

If the users you will deploy the Macro to are using the Combined Server Model then refer to section Chapter 20.4.1, "Setup Macros for Configuration Server or Combined Model" on page 716.

For users running the HTML Model you have two options:

- ▶ Allow other users access to the Macro by sending to them via email or placing on some location like a network drive for them to access
- ▶ ReRun the Deployment Wizard and on the page where the session is defined you must:
 - Start the Session

- Start the Macro Manager and import the Macro
- Redeploy the HTML generated by the Deployment Wizard to your web server and now your users will have access to the new Macro.

20.5 When Problems Occur with HOD Macros

A Macro is a sequence of instructions that allow you to navigate through some number of host screens. If you are not thorough and complete when defining each individual screen and EVERY possible screen that can follow it, you will experience problems.

For instance if a screen is matched too early then it is quite possible that the entire logic flow of the Macro will be disrupted i.e. Once a screen is matched prematurely then it is almost certain that the logic that follows this screen will be out of sync.

20.5.1 Causes of Screen Mismatches or Non-Matches

During initial screen recognition, when the Macro is first recorded, Macro uses OIA status and number of fields on a screen to determine Screen recognition. If a situation occurs where the screens that follow a recognized screen change and are no longer uniquely defined by their OIA status and field count, it is possible for Macro processing to think it is on the wrong next screen, i.e. a screen mismatch. If the flow of screen information from the host changes it can also cause a screen mismatch. A screen match based simply on OIA status and field counts is not always sufficient.

There are multiple reasons for Screen Mismatches and Timeouts

1. Timing Problems caused by Network/Host delays

To slow down Macro execution use the Macro manager, add either a global pause for all Macros or add a Screen pause on a particular screen you are having problems with. Pauses will certainly resolve the problem of potential screen mismatches, but will also increase the amount of time it takes to run through a Macro. See 20.5.3, “Adding pauses” on page 722 and 20.5.4, “Adding timeouts” on page 723.

2. Transient Screens appearing during normal Macro processing

Unexpected screens from host may occur because of the occurrence of some transient screen appearance, such as a operator message popup. See 20.5.6, “Unexpected Screens from the Host” on page 725.

3. Multiple possible different text phrases on a screen

Defining the logic for what appears on a screen can be complicated, but HOD Macro gives you several options:

- If there are occasions where the current Screen may occasionally have some new text on it that indicates you should move to the Next Screen try using the Optional screen descriptor. See Chapter 20.5.2, “Optional field” on page 722.
- You may also have a screen that you define not by what the screen displays, but what it does not. For this situation using the Inverse Descriptor may work best. See Chapter 20.5.5, “Use of Inverse Descriptor” on page 724.

4. Host Screens being sent in multiple blocks of data

For situations where the screen data is sent in blocks, the screen will be updated as the separate portions of the screen are received. During this screen update the HOD Macro processing may recognize the half written screen in error. To fix this, you may wish to add additional fields to the screen description. A field that occurs in the top portion of the screen and one that appears in the lower portion of the screen is normally sufficient to ensure the entire screen has arrived.

5. Next Screens with same number of Fields and OIA Status

The screens that follow a recognized screen (that is, the Next Screen) may not always have a unique OIA and field Count value that tells them apart. Using the MacroManager, add additional fields to the Screen Descriptions for the screens in question. You may find that simply adding:

- the cursor position
- screen attribute, see Chapter 20.5.7, “Screen attributes” on page 727
- additional text string
- additional fields as described in step 3 above
- to the screen description resolves all your problems.

6. Host Screen changes by the addition of an item in a list of items

The easiest way to handle situations where lists of host data may change is to add looping into the HOD Macro logic. See 20.5.8, “Screen scroll problems - adding looping to Macro” on page 727.

You may ultimately need to debug some problems by adding Trace statements in the Macro. See “Adding trace statements to Macros” on page 729.

20.5.2 Optional field

For every screen definition there is an **Optional** screen descriptor field. When a descriptor is defined as optional, it is used for screen matching when all the non-optional descriptors have been checked but have *failed* to match. This can be useful concept when a screen has the possibility of containing multiple field possibilities that still require you to move onto the next screen

For instance, you may have a screen that displays the text RUNNING in the lower right portion of the screen. On occasion this screen may display CONNECTED rather than RUNNING. This is where using the Optional field descriptor may be useful. The Optional descriptor configures the screen recognition logic to recognize the screen if the text expected on a screen exists or when the Optional descriptor is present.

Example 20-3 Logic Flow of Optional text field

```
If text on screen exists then
    screen match /* screen has Running */
else
    if optional field exists then
        screen match /* screen displays CONNECTED */
else
    no match
```

20.5.3 Adding pauses

The pause time can be set per screen or globally for the Macro. The pause time is defined as the time to wait between actions on a screen once a screen has been displayed. Increase this time to allow for possible performance problems that may occur during interactions with the host.

20.5.4 Adding timeouts

The timeout value in milliseconds determines how long the Macro will wait to match a screen before timing out. If the timeout value is not long enough then it is possible for the Macro to timeout and stop executing before the host screen becomes available. Timeout value in the Macro can be set for an individual screen or globally. Refer to Figure 20-17.

The image shows a screenshot of the 'Host Access Macro Editor' dialog box. The 'Macro' tab is selected, and the 'Screens' sub-tab is active. The dialog contains several input fields and checkboxes for configuring the macro's behavior. The 'Macro Name' field is set to 'logoff'. The 'Description', 'Author', and 'Creation Date' fields are empty. The 'Pause Between Actions' checkbox is checked, with a value of 300 milliseconds. The 'Timeout Between Screens' checkbox is checked, with a value of 60000 milliseconds. The 'Show All Prompts at Start of Macro' and 'Use Variables and Arithmetic Expressions In Macro' checkboxes are also checked. At the bottom, there are buttons for 'Save', 'Cancel', 'Edit Code...', 'Import...', 'Export...', and 'Help'. A status bar at the very bottom indicates 'Define the general attributes of the macro'.

Field	Value
Macro Name	logoff
Description	
Author	
Creation Date	
Pause Between Actions	300 milliseconds
Timeout Between Screens	60000 milliseconds
Show All Prompts at Start of Macro	Checked
Use Variables and Arithmetic Expressions In Macro	Checked

Figure 20-17 Global screen timeouts

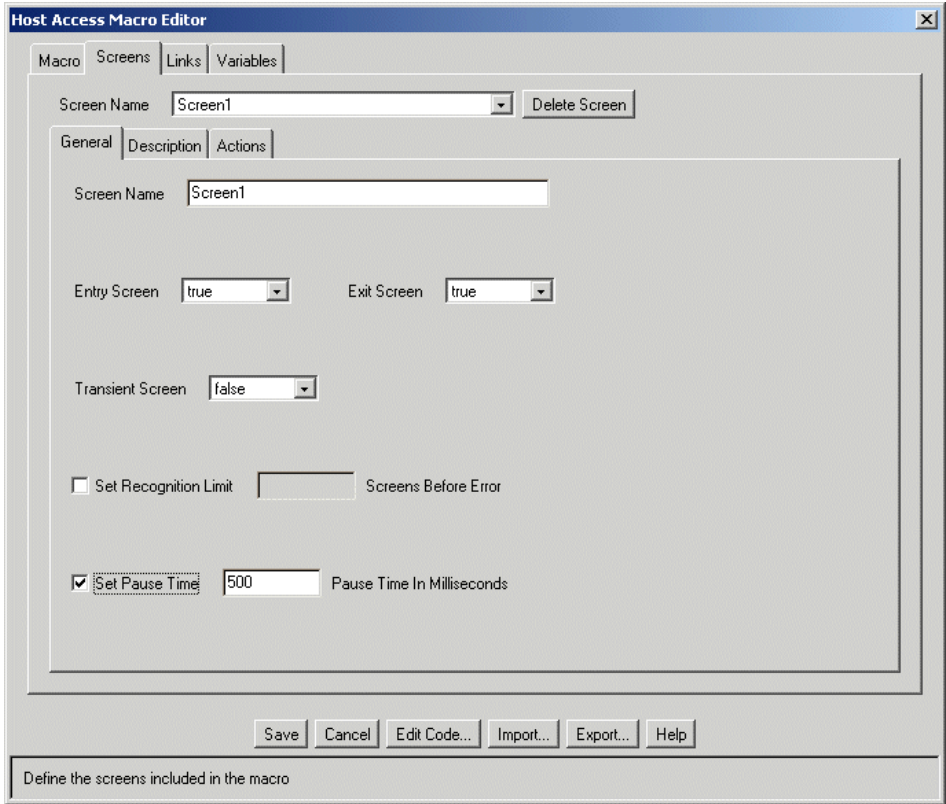


Figure 20-18 Screen unique pause time

20.5.5 Use of Inverse Descriptor

When the option to use Inverse Descriptor for a screen is set, this tells the Macro processing to recognize a screen where the event does NOT appear. This can be useful when defining a screen and you are not sure what will appear on a screen but certainly know what will not appear.

Table 20-1 Inverse Descriptors

Descriptor	Inverse Descriptor Means
Cursor	If you select true, the cursor defined by this descriptor must not be at the specified cursor Row and Column.

Descriptor	Inverse Descriptor Means
Attribute	If you select true, the attribute defined by this descriptor must not appear on the session screen at the specific coordinate specified
String	If you select true, the string defined by this descriptor must not appear in the area defined by Start Row, Start Column, End Row, and End Column.
Field Counts	If Optional is true for Number of Fields, the number of fields on the screen should not equal the descriptor value. If Optional is true for Number of Input fields, the number of input fields on the screen should not equal the descriptor value.

20.5.6 Unexpected Screens from the Host

It is imperative that you define all possible screens the host may send during the time that a Macro executes. If you fail to do this, then the Macro logic will timeout when a new screen shows up that is unexpected while it waits for the expected screen. If the order of screens being presented may change, then you must change the screen logic appropriately.

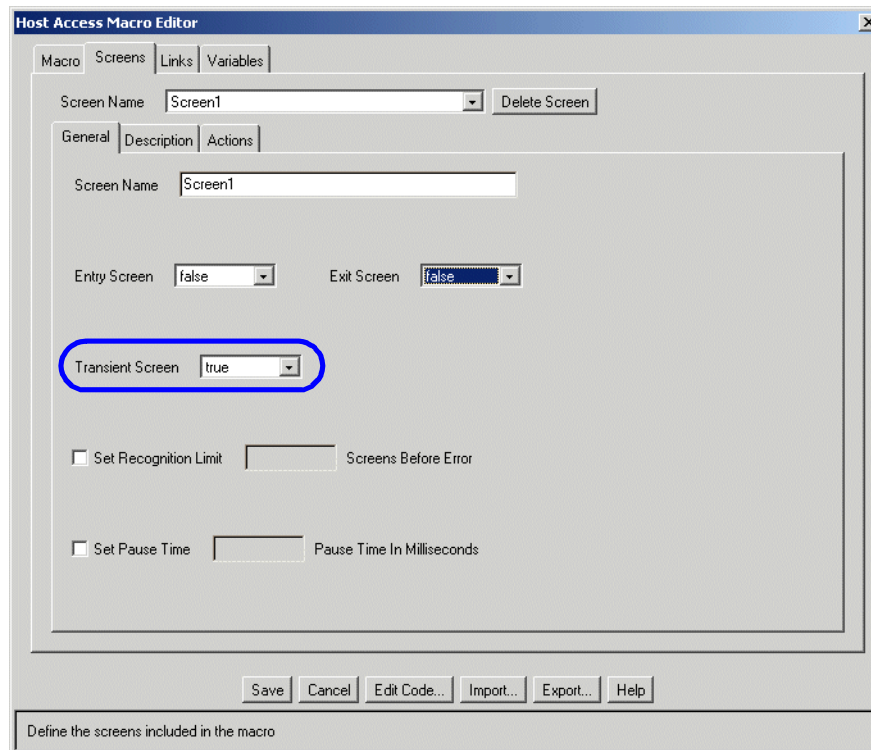


Figure 20-19 Setting Up Transient Screen after Capture

For situations involving a screen that may randomly occur anytime during the Macro execution, you may want to consider capturing this screen as a transient screen. The Macro will check for transient screens on every new host screen update. This can be useful for screens, such as operator messages or system warnings. Use the Macro Manager to capture this transient screen and add it into the Macro logic flows.

Note: Be sure to set the option Transient Screen to True. See Figure 20-19.

20.5.7 Screen attributes

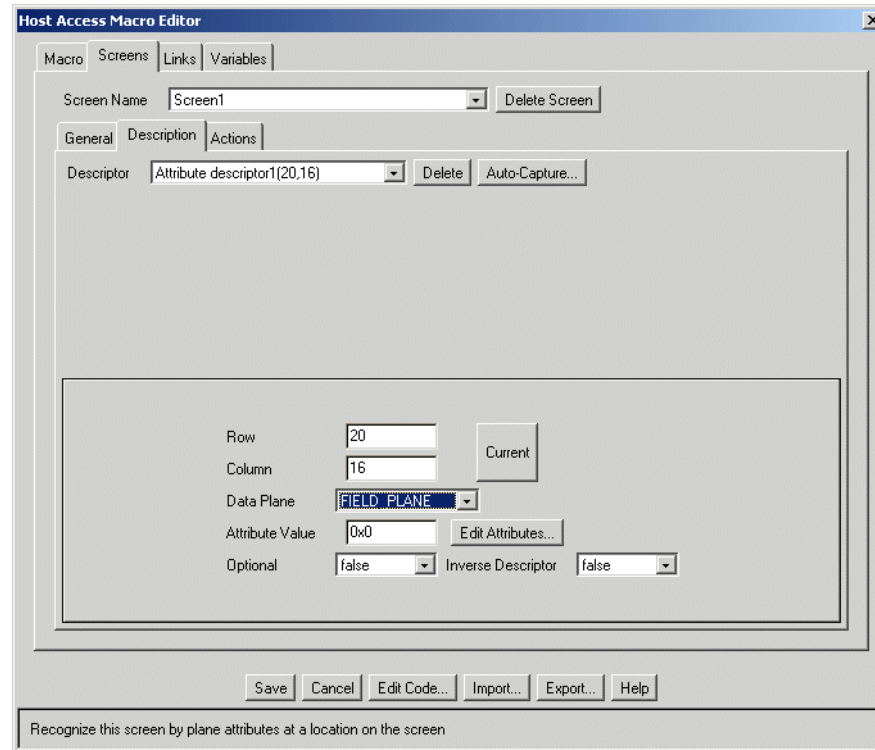


Figure 20-20 Using Attribute Descriptor

You can add additional screen descriptors to your screen to insure recognition of the correct screen. The attribute descriptor allows you to recognize a screen by plane attributes (color, field, or extended field) at a specified row and column position. See Figure 20-20.

20.5.8 Screen scroll problems - adding looping to Macro

If you have a host screen that contains a list that you scroll through, you may face a situation where the list gets added to and flows onto an additional host page that your Macro logic is not prepared for. To handle this you can add looping into the Macro logic.

Assume the logic in your Macro is something simple like:

- ▶ Recognize the first screen in the list, call this screen 1
- ▶ Press F6 to scroll to next page, call this screen 2

- Press F6 to scroll to next page, call this screen 3
- Read the value for an entry from screen 3

At a later time, someone added some new items in front of your selected item in the list. Now the logic would require you to press F6 an additional time. There is now a screen after step 3 above that you must scroll to. You can prepare for this if you use the Macro manager and add the following to screen 2:

- Modify the screen description for screen 2 by adding the Valid Next screen for screen 2 to include screen 2. See Figure 20-21.

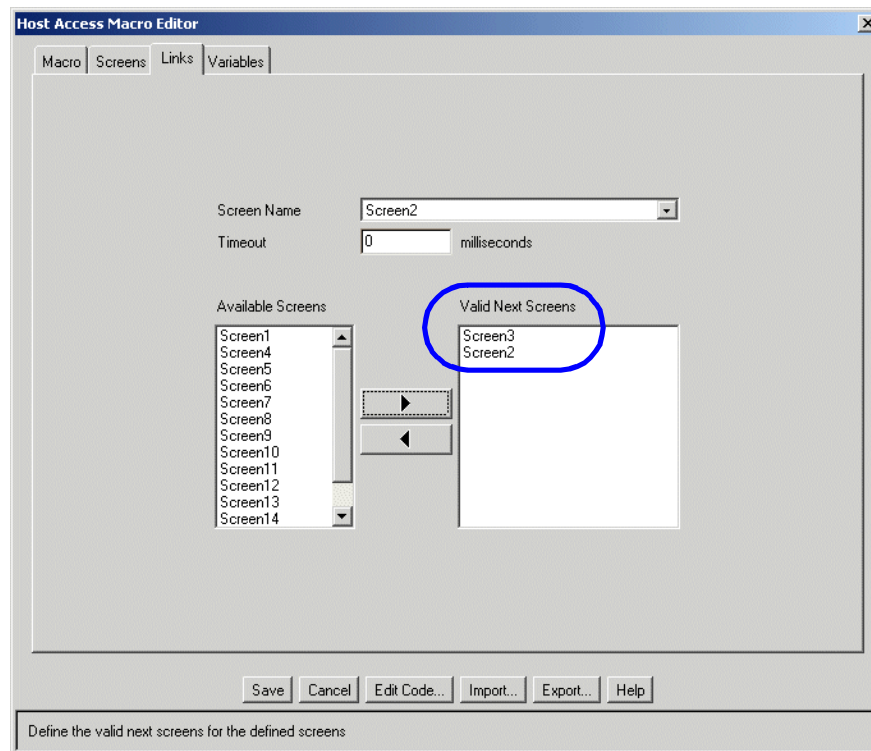


Figure 20-21 Screen links to allow looping

- Add the Inversion String Descriptor (see 20.5.5, “Use of Inverse Descriptor” on page 724) to Screen 2. This will have the effect of allowing screens to continue to execute itself as long as the field you are scrolling to locate does NOT exist on screen 2.
- Set the Screen Recognition Limit to prevent looping forever. See Figure 20-22.

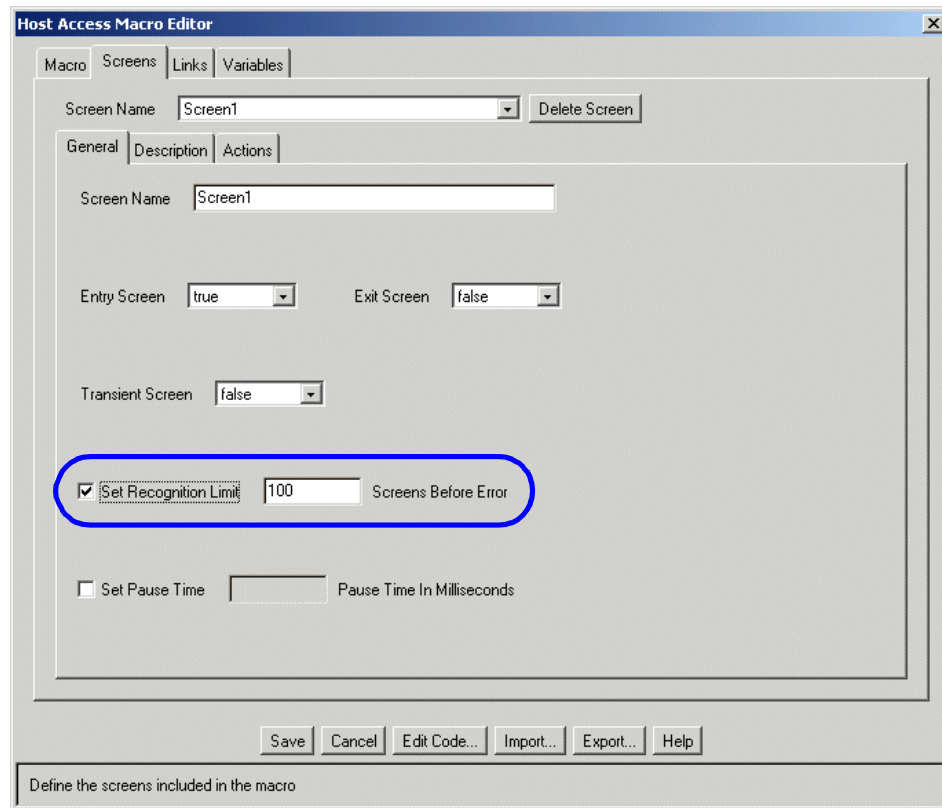


Figure 20-22 Setting screen recognition limit

When you finally exit from screen 2, it will go to screen 3 where you can read the value you have been scrolling to locate.

20.6 Problem determination tools and strategies

20.6.1 Adding trace statements to Macros

Macros are easily debugged using the ability to direct trace output to the java console: Use the Macro Manager to add the Action trace action() to your Macro. Be sure you use the order option to place the trace option in the appropriate location in the sequence of actions that occur for the screen.

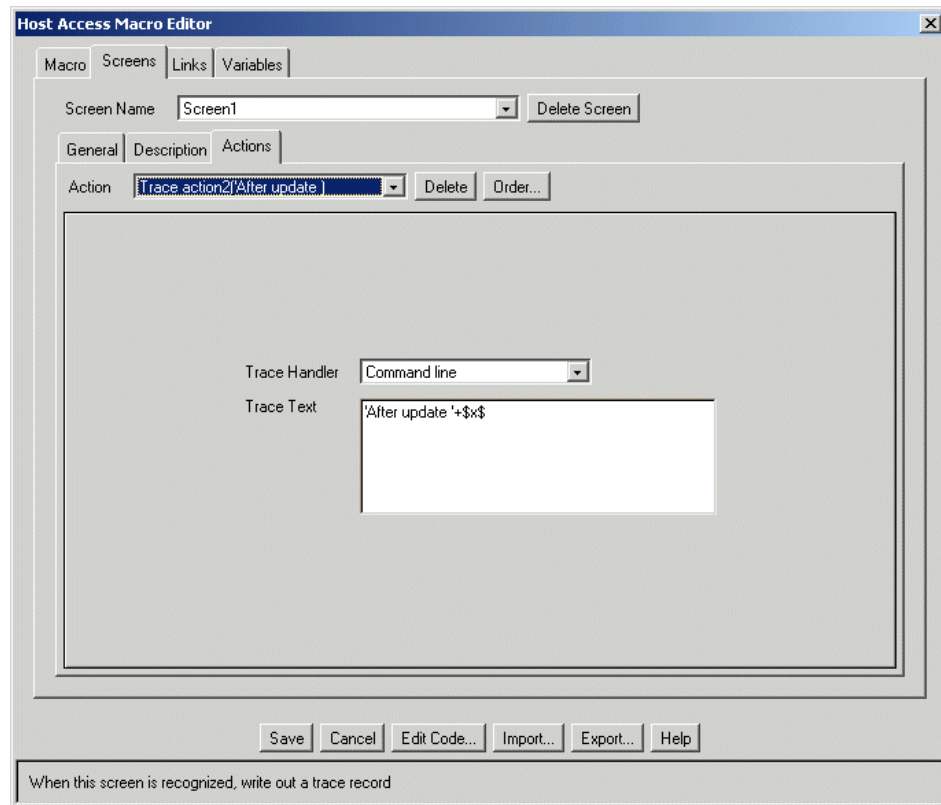


Figure 20-23 Adding trace for Variable \$X\$ in a Macro

20.6.2 Errors with HOD Macro variables

When your HOD Macro becomes large and complicated enough you will undoubtedly have a few programming errors. For instance if you attempt to use a variable before it is defined you may see this error displayed in the browser when the Macro executes:

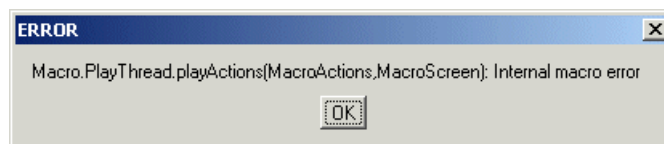


Figure 20-24 Undefined variable error

The Java console in the meantime will trace some additional errors:

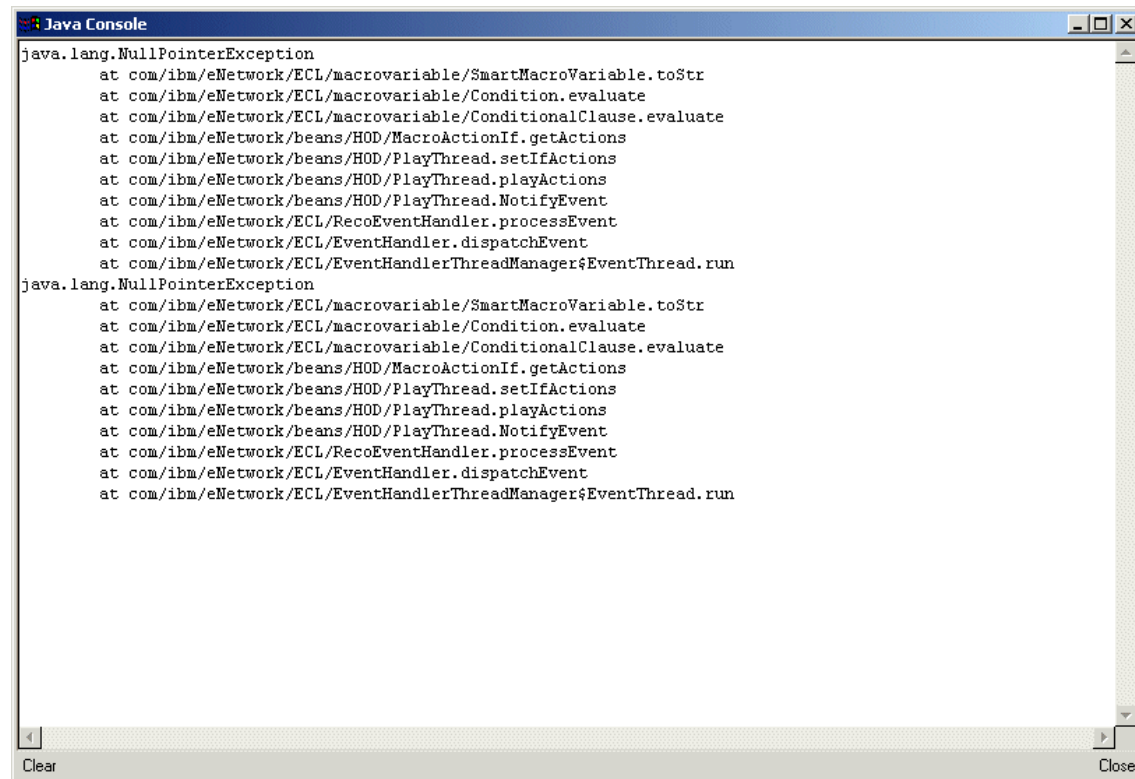


Figure 20-25 Java Console when Macro Error occurs

20.6.3 Using problem determination trace

The most powerful debug tool available when trying to determine why a Macro is not working as you thought is the Problem Determination Trace. This facility is available when you are using HOD Clients with Problem Determination.

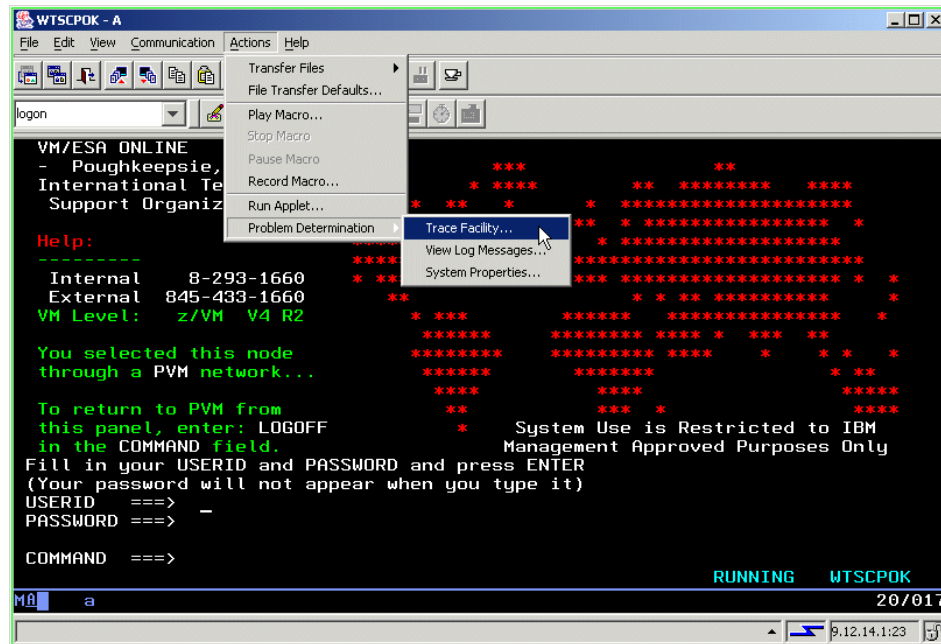


Figure 20-26 Start problem determination trace

To start the Trace facility select from the Emulator Toolbar **Actions->Problem Determination Trace Facility**. See Figure 20-26.

Now you must turn on the appropriate problem determination trace, see Figure 20-27.

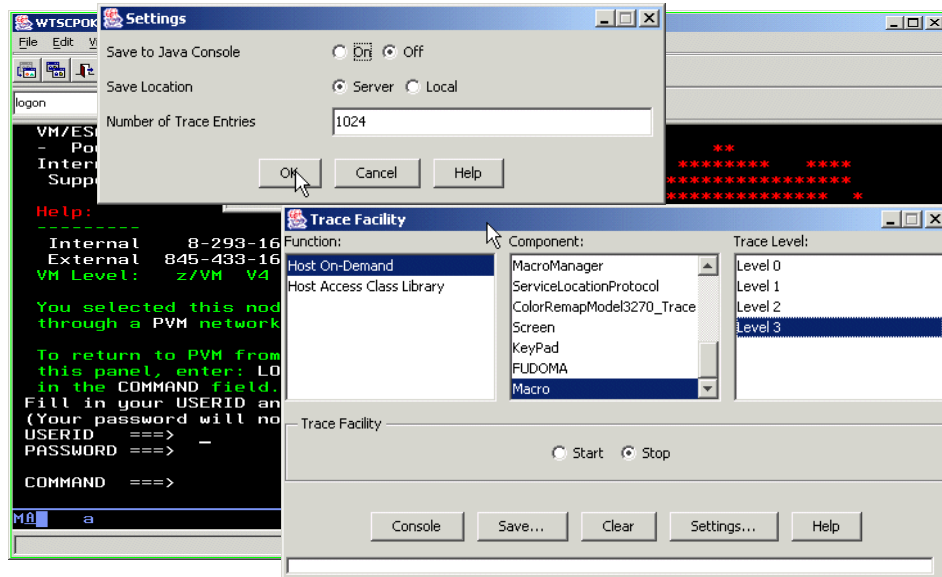


Figure 20-27 Enabling Macro trace

Start the trace and then execute your Macro. Stop the trace and save the trace file. An example of trace output is shown in Example 20-4. This example illustrates the errors that are traced when the initial screen is not recognized.

Example 20-4 Macro problem determination trace

```
4@0@09/20/2002 08:52:16:580@Host On-Demand@Macro@?@setProperty():2: macroName=
4@1@09/20/2002 08:52:16:580@Host On-Demand@Macro@?@setProperty():2: macroDescription=
4@2@09/20/2002 08:52:16:580@Host On-Demand@Macro@?@setProperty():2: empty=true
4@3@09/20/2002 08:52:16:580@Host On-Demand@Macro@?@setProperty():2: state=6
4@4@09/20/2002 08:52:16:590@Host On-Demand@Macro@?@setProperty():2: macroName=logon
4@5@09/20/2002 08:52:16:590@Host On-Demand@Macro@?@setProperty():2: macroDescription=test
4@6@09/20/2002 08:52:16:590@Host On-Demand@Macro@?@setMacro input:
4@7@09/20/2002 08:52:16:590@Host On-Demand@Macro@?@HAScript name="logon" description="test"
timeout="60000" pausetime="300" promptall="true" author="" creationdate=""
supressclearevents="false" usevars="false" > <screen name="Screen1" entriyscreen="true"
exitscreen="false" transient="false"> <description> <oia
status="NOTINHIBITED" optional="false" invertmatch="false" /> </description>
<actions> <input value="arc[tab]hod4mea[enter]" row="0" col="0" movecursor="true"
xlatehostkeys="true" encrypted="false" /> </actions> <nextscreens timeout="0"
> <nextscreen name="Screen2" /> </nextscreens> </screen>
<screen name="Screen2" entriyscreen="false" exitscreen="true" transient="false">
<description> <oia status="NOTINHIBITED" optional="false" invertmatch="false" />
<numfields number="7" optional="false" invertmatch="false" /> <numinputfields
```

```

number="1" optional="false" invertmatch="false" />      </description>      <actions>
<input value="f[enter]" row="0" col="0" movecursor="true" xlatehostkeys="true"
encrypted="false" />      </actions>      <nextscreens timeout="0" >
</nextscreens>      </screen>      </HAScript>
4@8@09/20/2002 08:52:16:620@Host On-Demand@Macro@?@setProperty():2: empty=false
4@9@09/20/2002 08:52:16:620@Host On-Demand@Macro@?@setProperty():2: state=1
4@10@09/20/2002 08:52:16:620@Host On-Demand@Macro@?@setProperty():2: state=2
4@11@09/20/2002 08:52:16:630@Host On-Demand@Macro@?@
4@12@09/20/2002 08:52:16:630@Host
On-Demand@Macro@?@-----
4@13@09/20/2002 08:52:16:630@Host On-Demand@Macro@?@Macro debug starting.
4@14@09/20/2002 08:52:16:630@Host
On-Demand@Macro@?@-----
4@15@09/20/2002 08:52:16:630@Host On-Demand@Macro@?@Warning, the screen could have been updated
between the screen
4@16@09/20/2002 08:52:16:630@Host On-Demand@Macro@?@comparison and the trace log addition. For
this reason, the actual
4@17@09/20/2002 08:52:16:630@Host On-Demand@Macro@?@contents of this trace will be slightly out
of sync and could
4@18@09/20/2002 08:52:16:630@Host On-Demand@Macro@?@indicate a PASS or a FAIL when the opposite
is shown in the trace.
4@19@09/20/2002 08:52:16:630@Host On-Demand@Macro@?@This occurs particularly often with strings.
4@20@09/20/2002 08:52:16:630@Host
On-Demand@Macro@?@-----
4@21@09/20/2002 08:52:16:650@Host On-Demand@Macro@?@
4@22@09/20/2002 08:52:16:650@Host
On-Demand@Macro@?@-----
4@23@09/20/2002 08:52:16:650@Host On-Demand@Macro@?@Screen1, matched = true
4@24@09/20/2002 08:52:16:650@Host
On-Demand@Macro@?@-----
4@25@09/20/2002 08:52:16:650@Host On-Demand@Macro@?@Macro action executing:
4@26@09/20/2002 08:52:16:650@Host On-Demand@Macro@?@ Screen Name = Screen1
4@27@09/20/2002 08:52:16:650@Host On-Demand@Macro@?@ Action Index = 0
4@28@09/20/2002 08:52:16:650@Host On-Demand@Macro@?@ Action Info = <input
value="arc[tab]hod4mea[enter]" />
4@29@09/20/2002 08:52:16:650@Host
On-Demand@Macro@?@-----
4@30@09/20/2002 08:52:16:650@Host On-Demand@Macro@?@
4@31@09/20/2002 08:52:16:991@Host On-Demand@Macro@?@
4@32@09/20/2002 08:52:16:991@Host
On-Demand@Macro@?@-----
4@33@09/20/2002 08:52:16:991@Host On-Demand@Macro@?@Screen2, matched = false
4@34@09/20/2002 08:52:17:001@Host
On-Demand@Macro@?@-----
4@35@09/20/2002 08:52:17:001@Host On-Demand@Macro@?@ Screen Descriptor Details:
4@36@09/20/2002 08:52:17:001@Host On-Demand@Macro@?@ FAIL: <oia status="NOTINHIBITED" />
4@37@09/20/2002 08:52:17:001@Host On-Demand@Macro@?@ ACTUAL: <oia status="NOTINHIBITED"
optional="false" invertmatch="false" />
4@38@09/20/2002 08:52:17:001@Host On-Demand@Macro@?@ FAIL: <numfields number="7" />

```

```

4@39@09/20/2002 08:52:17:001@Host On-Demand@Macro??@
optional="false" invertmatch="false" />
4@40@09/20/2002 08:52:17:001@Host On-Demand@Macro??@
4@41@09/20/2002 08:52:17:001@Host On-Demand@Macro??@
optional="false" invertmatch="false" />
4@42@09/20/2002 08:52:17:001@Host
On-Demand@Macro??@-----
4@43@09/20/2002 08:52:17:001@Host On-Demand@Macro??@
4@44@09/20/2002 08:52:17:001@Host On-Demand@Macro??@
4@45@09/20/2002 08:52:17:001@Host
On-Demand@Macro??@-----
4@46@09/20/2002 08:52:17:001@Host On-Demand@Macro??@Screen2, matched = false
4@47@09/20/2002 08:52:17:001@Host
On-Demand@Macro??@-----
4@48@09/20/2002 08:52:17:001@Host On-Demand@Macro??@ Screen Descriptor Details:
4@49@09/20/2002 08:52:17:001@Host On-Demand@Macro??@ FAIL: <oia status="NOTINHIBITED" />
4@50@09/20/2002 08:52:17:001@Host On-Demand@Macro??@ ACTUAL: <oia status="NOTINHIBITED"
optional="false" invertmatch="false" />
4@51@09/20/2002 08:52:17:001@Host On-Demand@Macro??@ FAIL: <numfields number="7" />
4@52@09/20/2002 08:52:17:001@Host On-Demand@Macro??@ ACTUAL: <numfields number="1"
optional="false" invertmatch="false" />
4@53@09/20/2002 08:52:17:001@Host On-Demand@Macro??@ FAIL: <numinputfields number="1" />
4@54@09/20/2002 08:52:17:001@Host On-Demand@Macro??@ ACTUAL: <numinputfields number="1"
optional="false" invertmatch="false" />
4@55@09/20/2002 08:52:17:001@Host
On-Demand@Macro??@-----
4@56@09/20/2002 08:52:17:001@Host On-Demand@Macro??@
4@57@09/20/2002 08:52:17:001@Host On-Demand@Macro??@
4@58@09/20/2002 08:52:17:001@Host
On-Demand@Macro??@-----
4@59@09/20/2002 08:52:17:001@Host On-Demand@Macro??@Screen2, matched = false
4@60@09/20/2002 08:52:17:001@Host
On-Demand@Macro??@-----
4@61@09/20/2002 08:52:17:001@Host On-Demand@Macro??@ Screen Descriptor Details:
4@62@09/20/2002 08:52:17:001@Host On-Demand@Macro??@ FAIL: <oia status="NOTINHIBITED" />
4@63@09/20/2002 08:52:17:001@Host On-Demand@Macro??@ ACTUAL: <oia status="NOTINHIBITED"
optional="false" invertmatch="false" />
4@64@09/20/2002 08:52:17:001@Host On-Demand@Macro??@ FAIL: <numfields number="7" />
4@65@09/20/2002 08:52:17:011@Host On-Demand@Macro??@ ACTUAL: <numfields number="1"
optional="false" invertmatch="false" />
4@66@09/20/2002 08:52:17:011@Host On-Demand@Macro??@ FAIL: <numinputfields number="1" />
4@67@09/20/2002 08:52:17:011@Host On-Demand@Macro??@ ACTUAL: <numinputfields number="1"
optional="false" invertmatch="false" />
4@68@09/20/2002 08:52:17:011@Host
On-Demand@Macro??@-----
4@69@09/20/2002 08:52:17:011@Host On-Demand@Macro??@
4@70@09/20/2002 08:52:17:011@Host On-Demand@Macro??@
4@71@09/20/2002 08:52:17:011@Host
On-Demand@Macro??@-----

```

```

4@72@09/20/2002 08:52:17:011@Host On-Demand@Macro??@Screen2, matched = false
4@73@09/20/2002 08:52:17:011@Host
On-Demand@Macro??@-----
4@74@09/20/2002 08:52:17:011@Host On-Demand@Macro??@ Screen Descriptor Details:
4@75@09/20/2002 08:52:17:011@Host On-Demand@Macro??@ FAIL: <oia status="NOTINHIBITED" />
4@76@09/20/2002 08:52:17:011@Host On-Demand@Macro??@ ACTUAL: <oia status="NOTINHIBITED"
optional="false" invertmatch="false" />
4@77@09/20/2002 08:52:17:011@Host On-Demand@Macro??@ FAIL: <numfields number="7" />
4@78@09/20/2002 08:52:17:011@Host On-Demand@Macro??@ ACTUAL: <numfields number="1"
optional="false" invertmatch="false" />
4@79@09/20/2002 08:52:17:011@Host On-Demand@Macro??@ FAIL: <numinputfields number="1" />
4@80@09/20/2002 08:52:17:011@Host On-Demand@Macro??@ ACTUAL: <numinputfields number="1"
optional="false" invertmatch="false" />
4@81@09/20/2002 08:52:17:011@Host
On-Demand@Macro??@-----
4@82@09/20/2002 08:52:17:011@Host On-Demand@Macro??@
4@83@09/20/2002 08:52:17:081@Host On-Demand@Macro??@
4@84@09/20/2002 08:52:17:081@Host
On-Demand@Macro??@-----
4@85@09/20/2002 08:52:17:081@Host On-Demand@Macro??@Screen2, matched = false
4@86@09/20/2002 08:52:17:081@Host
On-Demand@Macro??@-----
4@87@09/20/2002 08:52:17:081@Host On-Demand@Macro??@ Screen Descriptor Details:
4@88@09/20/2002 08:52:17:081@Host On-Demand@Macro??@ FAIL: <oia status="NOTINHIBITED" />
4@89@09/20/2002 08:52:17:081@Host On-Demand@Macro??@ ACTUAL: <oia status="NOTINHIBITED"
optional="false" invertmatch="false" />
4@90@09/20/2002 08:52:17:081@Host On-Demand@Macro??@ FAIL: <numfields number="7" />
4@91@09/20/2002 08:52:17:081@Host On-Demand@Macro??@ ACTUAL: <numfields number="1"
optional="false" invertmatch="false" />
4@92@09/20/2002 08:52:17:081@Host On-Demand@Macro??@ FAIL: <numinputfields number="1" />
4@93@09/20/2002 08:52:17:081@Host On-Demand@Macro??@ ACTUAL: <numinputfields number="1"
optional="false" invertmatch="false" />
4@94@09/20/2002 08:52:17:081@Host
On-Demand@Macro??@-----
4@95@09/20/2002 08:52:17:091@Host On-Demand@Macro??@
4@96@09/20/2002 08:52:17:091@Host On-Demand@Macro??@
4@97@09/20/2002 08:52:17:091@Host
On-Demand@Macro??@-----
4@98@09/20/2002 08:52:17:091@Host On-Demand@Macro??@Screen2, matched = false
4@99@09/20/2002 08:52:17:091@Host
On-Demand@Macro??@-----
4@100@09/20/2002 08:52:17:091@Host On-Demand@Macro??@ Screen Descriptor Details:
4@101@09/20/2002 08:52:17:091@Host On-Demand@Macro??@ FAIL: <oia status="NOTINHIBITED" />
4@102@09/20/2002 08:52:17:091@Host On-Demand@Macro??@ ACTUAL: <oia status="NOTINHIBITED"
optional="false" invertmatch="false" />
4@103@09/20/2002 08:52:17:091@Host On-Demand@Macro??@ FAIL: <numfields number="7" />
4@104@09/20/2002 08:52:17:091@Host On-Demand@Macro??@ ACTUAL: <numfields number="1"
optional="false" invertmatch="false" />
4@105@09/20/2002 08:52:17:091@Host On-Demand@Macro??@ FAIL: <numinputfields number="1" />

```

```

4@106@09/20/2002 08:52:17:091@Host On-Demand@Macro@?@ ACTUAL: <numinputfields number="1"
optional="false" invertmatch="false" />
4@107@09/20/2002 08:52:17:091@Host
On-Demand@Macro@?@-----
4@108@09/20/2002 08:52:17:091@Host On-Demand@Macro@?@
4@109@09/20/2002 08:52:17:091@Host On-Demand@Macro@?@
4@110@09/20/2002 08:52:17:091@Host
On-Demand@Macro@?@-----
4@111@09/20/2002 08:52:17:091@Host On-Demand@Macro@?@Screen2, matched = false
4@112@09/20/2002 08:52:17:091@Host
On-Demand@Macro@?@-----
4@113@09/20/2002 08:52:17:091@Host On-Demand@Macro@?@ Screen Descriptor Details:
4@114@09/20/2002 08:52:17:091@Host On-Demand@Macro@?@ FAIL: <oia status="NOTINHIBITED" />
4@115@09/20/2002 08:52:17:091@Host On-Demand@Macro@?@ ACTUAL: <oia status="NOTINHIBITED"
optional="false" invertmatch="false" />
4@116@09/20/2002 08:52:17:091@Host On-Demand@Macro@?@ FAIL: <numfields number="7" />
4@117@09/20/2002 08:52:17:091@Host On-Demand@Macro@?@ ACTUAL: <numfields number="1"
optional="false" invertmatch="false" />
4@118@09/20/2002 08:52:17:091@Host On-Demand@Macro@?@ FAIL: <numinputfields number="1" />
4@119@09/20/2002 08:52:17:091@Host On-Demand@Macro@?@ ACTUAL: <numinputfields number="1"
optional="false" invertmatch="false" />
4@120@09/20/2002 08:52:17:091@Host 0
1@69@09/20/2002 08:52:33:164@pdpsi@?@?@line.separator =
1@70@09/20/2002 08:52:33:164@pdpsi@?@?@java.vm.name = Java HotSpot(TM) Client VM
1@71@09/20/2002 08:52:33:164@pdpsi@?@?@user.region = US
1@72@09/20/2002 08:52:33:164@pdpsi@?@?@file.encoding = Cp1252
1@73@09/20/2002 08:52:33:164@pdpsi@?@?@acl.write.default =
1@74@09/20/2002 08:52:33:164@pdpsi@?@?@browser.vendor = Sun Microsystems, Inc.
1@75@09/20/2002 08:52:33:164@pdpsi@?@?@java.specification.version = 1.3

```



Host Access Toolkit

The HACP 3.0 Host Access Toolkit contains a set of Java libraries with which developers can create applets and applications for host access. The set of application programming interfaces (APIs) is a separate stand-alone product from Host On-Demand (HOD). It is bundled with Host On-Demand and is part of the Host Access Client Package, but shipped on its own CD. The toolkit can be installed and deployed independently of Host On-Demand.

In this chapter, we discuss important up-to-date information on how to use these libraries in today's real-world development and production environments. A previous redbook *Programming with the Programming with Host Access APIs*, SG24-5856, is available as an introduction to the structure of these libraries and how to get started developing with them. This chapter will serve mainly as an extension and revision to the more recent redbook, *IBM Host Access Client Package*, SG24-6182-00.

The following libraries will be discussed in this chapter:

- ▶ Host Access Class Library for Java (HACLJ)
- ▶ Host Access Beans for Java (HABJ)
- ▶ J2EE Connectors

The free-for-downloading EHLLAPI Bridge and Utility technologies will also be discussed.

21.1 Introduction

In this release of the Host Access Toolkit, the associated APIs contain extensions and improvements. The related Personal Communications associated Java APIs remain at the same version as the last release of Host Access Client Package. Those APIs are equivalent to HOD 4.3.

The new API features include:

1. The Toolkit libraries are now supplied in two versions: jdk1.1 (compiled with IBM JDK 1.1.8) and java2 (compiled with IBM SDK 1.3.0). The GUI beans of the java2 library set are compatible with Swing-based applications and applets.

2. Loadable Applet Interface

You can develop custom applications implementing this interface that rely on the Host Access Class Library (HACL) and Host Access Beans for Java (HABJ) libraries. When served from the HOD server, the libraries for HACL and HABJ can be cached to the client browser.

This facility will not cache or manage your non-HOD custom libraries (Java archive (JAR) and cabinet (CAB) files). Nonetheless your custom application will benefit from significantly reduced start up time.

3. Many visual beans in the Host Access Beans API now include Accessibility features. All Accessibility features require installation of a Java 2 plug-in.
4. Support for planes extract. Host On-Demand macros now support extraction of the following host screen information in addition to text:
 - color
 - field
 - extended field
 - DBCS
 - grid

Host On-Demand stores the screen attributes as planes

5. Macro enhancements
 - Macro functionality has been enhanced to support the following:
 - Manipulating variables during macro play
 - Conditional (if-else) statements
 - Launching macros from within other macros (chaining)
 - Launching applications from within macros.
 - Displaying numbers in localized formats.
 - Copy/append and paste/append functionality.

6. The Host On-Demand Java APIs have been extended to support communication through HTTP and Socks proxy servers, and host printing.
7. Applet improvements for the Host Access Class Library (HACL):
 - A new interface, CustomInterface, provides user applets with access to the Host On-Demand components HostTerminal, Applet instance, and Frame.
 - Parameters can now be passed to applets.
8. The Screenable interface has been added to the screen specifications for Host On-Demand 3270, 5250 and Customer Information Control System (CICS) sessions. This new interface provides a standardized record for manipulating data on a screen while supporting the specialized requirements of CICS and other IBM resource adapters.

The Screenable interface is supported along with the older Streamable screen interface. Its advantages over the Streamable interface include the following:

- Screenable records fully support field attributes. Screens do not need to be re-transmitted to send field attributes, improving performance
 - The Screenable interface does not require prior knowledge of the screen's layout. It supports both generic screen formats (where the sender does not know the screen format) and screen formats based on specific layout information.
 - Screenable records do not need to contain blank space information.
 - The Screenable interface supports a public interface for exchanging data between the screen record and the resource adapter.
9. The package com.ibm.eNetwork.ECL has been extended to provide host printing functionality.

IBM supplies additional Host Access APIs with Personal Communications for use in OLE/Automation, C++, and other environments. See the installed or printed documentation of that product for more information. For additional information on the differences between these two API sets, please refer to 21.3, "Host Access Toolkit vs. Personal Communications" on page 744.

Note: The term *application* is used loosely in this document. Unless explicitly stated, an *applet* is also inferred. In some topics, both terms will be used. Application refers to a Java program launched via the *main* entry point, while applet refers to Java code launched in a browser environment via the *init* entry point.

21.2 Host Access Toolkit requirements

Installing and using Host Access Toolkit has some requirements that must be met in terms of the supported operating systems, disk space requirements, Java 2 support and the types of browsers. This section explains each of these requirements.

21.2.1 Operating systems requirements

The installation of the toolkit and development of applications is supported on the following operating systems:

- ▶ Windows 95
- ▶ Windows 98
- ▶ Windows Me
- ▶ Windows NT 4.0 with service pack 5 or higher (SP 6a is recommended)
- ▶ Windows 2000
- ▶ Windows XP

Applications developed using the Host Access Toolkit can be run on other operating systems that support Java. The toolkit JAR files needed to run your application or applet can be packaged with your application or applet and copied to those other systems within the bounds of your licensing agreement.

The typical Host Access Toolkit requires 110MB of disk space to install.

Supported browsers

The following browsers can be used to run a Host Access Beans for Java or Host Access Class Library applet:

- ▶ Netscape Navigator 4.6, 4.7 and 6.x (Windows 95, Windows 98, Windows NT, Windows 2000, Windows XP, UNIX)
- ▶ Netscape Navigator 4.6.1 (IBM OS/2) and IBM Mozilla Web Browser for OS/2
- ▶ Microsoft Internet Explorer 4.01 with SP1, 5.0, 5.1, V5.5 and 6.0 with their most current service packs

Note: As of this writing, there is a special situation for users of Internet Explorer on Windows XP. The details are presented here:

<http://www.microsoft.com/java/xp.htm>

- ▶ Other browsers that support the Java Runtime Environment (JRE) 1.3 Plug-in or later

For the most up-to-date information, refer to the read me file and visit the Host On-Demand Web site.

21.2.2 Supported target environments

IBM supports developing applications derived from the Host Access Toolkit for the following platforms, JREs, and browsers (for applets).

Platforms

Supported platforms encompass Windows, IBM mainframe, IBM midrange and UNIX platforms.

- ▶ Windows: 95**, 98, NT 4.0, 2000, ME, XP**
- ▶ OS/2
- ▶ IBM OS/400
- ▶ IBM OS/390
- ▶ IBM AIX
- ▶ Linux
- ▶ Sun Solaris
- ▶ HP/UX

Note: Supported target Windows platforms vary according to the JVM release.

IBM JVM 1.1.8 and 1.3.0 are not supported on Windows XP (but may work).

IBM JVM 1.3.1 and above will not launch on Windows 95. In contrast, Sun JVMs up to 1.4 are supported for Windows 95. Please consult the java.sun.com website for specific details on supported platforms

Java Runtime Environment (JRE)

We discourage development of API-based software using JDK 1.1.8 and JDK 1.2, although it is certified for those levels. Only limited maintenance should be done with JDK 1.1. Version 1.1.8 is close to going out of service with the specific dates depending on the platform. Look forward to Java 2 V1.3.x and beyond for new development, keeping future V1.4 requirements in mind.

IBM JREs/JDKs should be used in conjunction with these libraries. You may download one from:

<http://www.ibm.com/developerworks/java/jdk/index.html/>

Select the IBM developer kits option and then select the appropriate operating system. The Java Plug-in is in the Windows package and installed by default.

21.3 Host Access Toolkit vs. Personal Communications

Personal Communications V5.6 (PCOMM) also contains HACLJ and HABJ API libraries. However, their functionality and features are at the level found in Host On-Demand V4.0. Source code developed in the PCOMM Java API library environment will generally compile and run in the HOD V7.0 Java API jdk1.1 library environment. A specific exception to this are the HACL screen recognition classes: the flow logic for PCOMM usage is different than the flow logic for HOD 7.0.

Personal Communications V5.6 ships with additional APIs to support Win32-specific development using programming languages such as Microsoft C++, Microsoft Visual Basic and Lotus Notes. For this chapter it is very important to contrast the Java libraries supplied by Host On-Demand 7.0 and Personal Communications Version 5.6.

The Java libraries supplied in the Host On-Demand Toolkit are written 100% in Java and can be executed in a platform-independent manner. The Host On-Demand product does not have to be installed on the client machine. The entire functionality of HACLJ and HABJ is contained in the supplied JAR and CAB files. These JARs/CABs would be installed on the local machine in the case of an application. They may be locally installed or downloaded from a server in the case of a browser-based applet.

The Personal Communications product is Windows- and OS/2-specific. The Personal Communications Version 5.6 version of HACLJ and HABJ functionality is absolutely dependent on the presence of the Personal Communications Version 5.6 emulator product being installed on the local machine. In fact,

sessions created with HACLJ or HABJ work by starting a hidden Personal Communications Version 5.6 emulator session(s). Portions of the HACLJ library are wrapper classes for Java Native Interface (JNI) access to this hidden Personal Communications session(s).

Host On-Demand-based HACLJ and HABJ are Java right down to the transport layer. In contrast, the Personal Communications version is largely Java, but the "transport" layer is the Personal Communications product itself.

The Host On-Demand Toolkit contains a variety of JARs/CABs to supply host access functionality (habasen.jar, ha3270n.jar, and others). Personal Communications only ships one JAR, pcseclj.jar, and no corresponding CAB file. CAB files are the primary library and security mechanism for running Java applets using MS Internet Explorer. Netscape uses JAR files. Therefore, the Java libraries supplied with Personal Communications are not intended and not supported for browser-applet usage. They should be used in stand-alone applications only.

21.4 Host Access Class Library for Java (HACLJ)

Traditionally, specialized terminal equipment was used so that the user could communicate and interact with several kinds of host computer presentation screens: 3270, 5250 or VT (ASCII virtual terminal). The advent of the multi-purpose Personal Computer ushered in terminal emulation software. The Java-based Host On-Demand emulator product is an example. Host On-Demand itself is based on Host Access Beans for Java plus considerable value-added glue logic to simulate an advanced terminal emulator. The Host Access Beans, in turn, rely heavily on the Host Access Class Library for Java.

Note: The HACLJ objects begin with ECL. The Host Access Class Library was originally called the Emulator Class Library (ECL), but the name HACL was deemed more appropriate. The actual code bindings maintain their original acronym, ECL.

21.4.1 HACLJ programming strategy

HACL for Java provides a non-visual API for interacting with back-end host machines running applications originally designed for human interaction. Human-oriented host applications relied on readable character presentation, formatted fields, color-coding and keyboard responses. The HACLJ library provides specialized classes for functionalities needed to mimic traditional human interaction with a series of host screen presentations ("green screens").

HACLJ contains no GUI (visible component) classes. The HACLJ classes interact with “invisible” presentations screens. HACLJ provides the PSDebugger, a simplified terminal which appears on demand by issuing `ECLSession.ShowPSDebugger(true)`.

A client side HACLJ-based application could simply be a straightforward replacement for a repetitive routine. There is no direct need for a terminal screen or decision-making by the operator. This most resembles a batch program.

Another application type deploys a remapped, reformatted or blended presentation of the information content extracted from one to several host screens and databases. The developer may create an application where the user may be completely unaware of the connection to the mainframe. The user does not need to learn the special procedures, commands or menu structure of the host application; these can be automated by a domain expert using HACLJ. As a result, the user can focus more on the task at hand. The HACLJ portion of the application is in complete control of host screen navigation, so if the host side presentation content changes, so must the navigation and recognition logic. The user interacts only with the business logic presented. What used to be done as a series of discrete steps sequentially to move and manage data between several hosts can be turned into a single flow of steps among multiple hosts simultaneously.

In addition to client-side development, one can develop an application in a middle-tier solution for host access. One such example is to create an EJB, or servlet, that utilizes HACLJ to do transactions for a user, such as a Web site to a bank. Here again, the HACLJ code is replacing what would have been human interaction with a remote mainframe back-end. The middle tier functionality receives commands via the Web and translates them into HACLJ-mediated trains of action. Back-end responses are screen scraped and transcoded into Web page-based replies. Some of this work has already been done for you by supplying the J2EE Connectors as part of the toolkit. The choice is simple: would you rather have a client machine or a middle-tier application server accessible to a client performing host navigation and screen scraping? Figure 21-1 demonstrates these concepts.

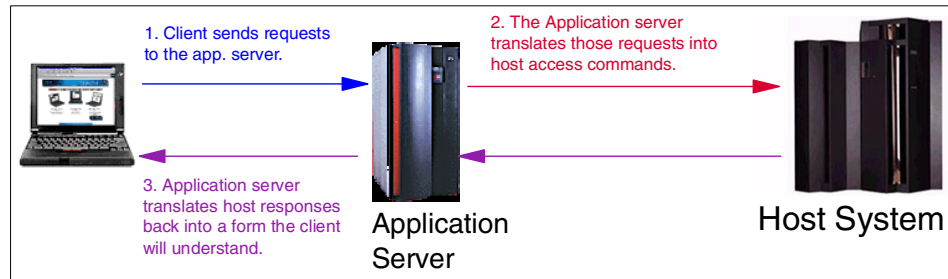


Figure 21-1 Application server with Host On-Demand J2EE Connectors

21.4.2 HACLJ functionality

Table 21-1 presents most of the higher-level functions that can be programmed using HACLJ.

Table 21-1 HACLJ functions and associated objects

Functionality	Associated objects in HACL
Important Global Constants	ECLConstants - used with HACLJ and HABJ
Session	ECLSession - communicates with host, derived from ECLConnection ECLConnection - connects to host ECLCommNotify - communication events listener interface
Presentation Space	ECLPS - logical representation of the Presentation Space (PS) or "green screen" ECLPSEvent - PS change notification event ECLPSListener - PS change notification listener interface ECLFieldList - logical representation of all of the fields present in the PS ECLField - logical representation of a single field within the PS
Operator Information Area (OIA)	ECLOIA - logical representation of the operator information area (OIA) ECLOIANotify - OIA change of state listener

Functionality	Associated objects in HACL
File Transfer	<p>ECLXfer - allows the transfer of files to and from a 3270 or 5250 host, over an established session. ECLXfer supports the 3270 Host File Transfer Program IND\$FILE (for SBCS) or APVUFILE (for DBCS) transfer protocols, which can be controlled by means of the standard IND\$FILE or APVUFILE send and receive options. ECLXfer also supports 5250 File Transfer, which can be controlled by means of the send and receive options specific to OS/400.</p> <p>ECLXferEvent - fired to notify listeners of progress during file transfers.</p> <p>ECLXferListener - interface can be used to implement an object which will receive the file transfer progress events, ECLXferEvent. Events are generated during file transfer as data buffers are transferred to or from the host.</p>

Functionality	Associated objects in HACL
Printer Session Control	<p>ECLHostPrintSession - can be used to establish a print connection with a host. This class defines the behavior and characteristics of the print session with the host. This class inherits operating characteristics and behaviors from its parent class ECLSession. Like ECLSession, ECLHostPrintSession can be constructed with a Properties object which contains all the configuration information for the print session. Configurable information includes the session type (3270 printer session or 5250 printer session), PDT file name and port number.</p> <p>ECLPrintJobEvents - are fired by ECLHostPrintSession to notify interested listeners about a current print job. There are several event types that coorelate to the nature of a print job state change or an error condition occurrence. Some event types contain additional information.</p> <p>ECLPrintJobListener - interface can be used to implement an object which will receive ECLPrintJobEvents. Events are generated whenever a printer job is started or completed. Special events are generated when print job related errors occur.</p>

Functionality	Associated objects in HACL
National Language Version Extensions	<p>ECLPSBIDIServices - interface provides access to the bidirectional (BIDI) language properties in an ECLPS object. An ECLPSBIDIServices object is only available when using bidirectional code pages (420, 424, or 803) in 3270/5250 Sessions.</p> <p>ECLPSTHAIServices - interface provides access to the THAI properties in a THAI ECLPS object. An ECLPSTHAIServices object is only available when using Thai codepages (838 or 1160) in 3270, 5250 and VT sessions.</p> <p>ECLXferBIDIServices - interface provides access to the ECLXfer Bi-directional (BIDI) Language properties in an ECLXfer object. An ECLXferBIDIServices object is only available when using Bi-directional code pages (420, 424, or 803).</p> <p>PSVTBIDIServices - interface provides access to the bidirectional (BIDI) language properties in a BIDI VT session. The PSVTBIDIServices object is only available when using bidirectional code pages.</p>

Functionality	Associated objects in HACL
Screen Recognition	<p>ECLScreenDesc - a target screen “described” via specialized descriptors</p> <p>ECLScreenReco - the screen recognition engine</p> <p>ECLRecoNotify - called when screen recognition engine “recognizes” a described screen</p> <p>ECLSDAttrib - describes a single attribute on a host screen</p> <p>ECLSDBlock - describes a block of strings on a host screen</p> <p>ECLSDCursor - describes the cursor position on a host screen</p> <p>ECLSDCustom - describes a custom recognition handler for a host screen</p> <p>ECLSDFields - describes the total number of fields on a host screen</p> <p>ECLSDInputFields - describes the number of input fields on a host screen</p> <p>ECLSDOIA - describes an OIA condition to wait for after a host screen is recognized</p> <p>ECLSDString - describes a single string on a host screen</p> <p>ECLSDScreenDescriptor - is the base class that all ECLSD* classes are derived from</p> <p>ECLCustomRecoEvent - emitted after standard screen matching logic for “final approval of match”</p> <p>ECLCustomRecoListener - an interface to extend the base ECLScreenReco screen matching logic</p> <p>ECLRecoDebugEvent - debug event emitted upon screen recognition occurrence</p> <p>ECLRecoDebugListener - debug event listener interface</p>
API Error Event	<p>ECLErr - error event classes specialized to HACLJ</p> <p>VariableException - is thrown when problems with Macro variables and arithmetic expressions are encountered. VariableException objects are created and populated with error and diagnostic information, and then thrown as exceptions. The VariableException object can then be caught and queried for the error information. VariableException is informational for Macro bean programming only.</p>

Functionality	Associated objects in HACL
Outboard Function Execution	ECLAppletInterface - access to currently running ECLSession
Trace Facility	ECLTrace - base of trace facility in HACLJ ECLTraceEvent - emitted to transmit internal state and progress information ECLTraceListener - trace event listener interface ECLTraceProducer - interface implemented by an HACLJ component reporting trace information

For full details, please refer to the online *Host Access Class Library Reference*.

21.4.3 Automated host navigation

The Host Access Toolkit provides several assistive technologies to help applications navigate screens. These technologies can minimize the amount of work spent developing the screen-to-screen flow needed to perform the required business logic of an application. An all-HACLJ application would use the screen recognition API of HACLJ. Usually, a HABJ-based application would use the Macro or MacroManager beans. However, Session and Terminal beans both provide access to ECLPS via ECLSession, so that the HACLJ screen recognition API can be used in an HABJ application.

Screen recognition in HACLJ

Screen recognition, or screenreco, provides an event-driven model for recognizing a particular host screen among a succession of screens presented. Using the respective classes enumerated in the table above, a developer defines one or more properties that will be used to “match” a screen. A set of descriptors must uniquely identify the target screen, especially when there are other “similar” screens that could also be reached in a non determinate traversal situation. The programmer must also account for screen completeness; there is no formal data stream signal to announce that the host has completed updating the screen. That requires judgment and experience on the part of the developer of the recognition criteria. Here are the properties that can be formed into various combinations for a unique match:

- ▶ Attribute byte of a field
- ▶ A unique string and position
- ▶ A block of strings
- ▶ The cursor position
- ▶ The number of fields on the screen

- ▶ The number of input fields
- ▶ An OIA condition to wait for after a host screen is recognized

These may not be enough when the same static screen layout is progressively updated in response to user inputs. Each of those updates may represent new states to be recognized in the business logic. Therefore, a fine differentiation is possible by using the “custom reco” facility based on:

- ▶ ECLSDCustom
Describes a custom recognition handler for a host screen
- ▶ ECLCustomRecoEvent
Emitted after standard screen-matching logic for “final approval of match”
- ▶ ECLCustomRecoListener
An interface to extend the base ECLScreenReco screen-matching logic

The screenreco engine triggers on each presentation space update (keyed to ECLPSEvent occurrence). A screen matches when a defined cluster of criteria is fulfilled. A reco event is fired to any registered listener. For more information, including examples, see *Programming with the Host Access APIs*, SG24-5856.

Known limitation

When ECLScreenReco is used with Terminal/Screen in the Personal Communications version, there is an unavoidable performance limitation. When a skilled touch typist reaches top speed, occasional keystrokes are not echoed to the screen. No keystrokes are lost from the input buffer of the virtual terminal. All keystrokes reappear when the presentation space is manually refreshed.

HACLJ suited to all platforms

All supported platforms can take advantage of HACL for Java because the platform does not need to provide a GUI engine. This can be eased a bit with Remote Abstract Windows Toolkit (RAWT), also called Remote AWT.

Timing

Host programs might be designed to indicate when a refreshed presentation is complete, but data stream protocols do not have a formal signal that a screen update has been completed. HACLJ itself does not relieve the programmer of doing a correct and full analysis of host program and communication link timing behavior.

Where am I?

The screenreco facility is a great aid for systematizing and automating the navigation of a series of host presentation screens. Nonetheless, secondary confirmation that the screen currently presented is the screen expected is absolutely essential. No less important is a strategy to gracefully fall back or withdraw when navigational expectations don't match reality.

Beep

The 3270 “bell” is sounded when a connection to a host succeeds. The 5250 “alert” is sounded when the OS presents a momentary overriding screen message. The Java JVM provides an equivalent Toolkit.Beep(), which HACLJ uses to provide an “audible.” When the HACLJ-based Java application is running on OS/390 or OS/400, that Toolkit.beep() call has the effect of stalling ECLPSEvents. The symptom is that the connected host appears not to be sending updated screen buffers.

There are two ways to avoid this.

- Quiet Mode

The ECLSession can be parameterized with the value pair:
ECLSession.SESSION_QUIETMODE, "true". This is valid for 3270 and 5250.

- Remote Abstract Windows Toolkit (RAWT)

Full details concerning RAWT are to be found in the article *Setting up the Remote Abstract Window Toolkit for Java on a remote display* and its associated links, found at:

<http://publib.boulder.ibm.com/pubs/html/as400/v4r4/ic2924/info/java/rzaha/devkit.htm>

21.5 Host Access Beans for Java overview

Unlike HACLJ, Host Access Beans for Java are intended for visual environments, with the exception of the Session and Macro beans. Developers can present to the user different pieces of an emulator to quickly provide core functions one might expect. The Terminal and Screen beans display the actual screens and OIA information generated by the back-end host. Because Host Access Beans for Java are components themselves, rapid development of applications based upon this library is possible. If the API of the beans is not sufficient for an application's needs, the underlying HACL for Java API is accessible.

Applications constructed from the visual HAB for Java do not have a place in the middle-tier host access solution. But the Session bean wired with Macro bean is a potent combination in an Enterprise Java Bean servlet environment. HAB for Java provides the greatest benefit when developing client-side solutions for host access, but don't forget that these solutions can either be installed locally on a machine, or downloaded from a Web server to reap the benefits of a single deployment location. Consider Host On-Demand as a model for this.

The Host Access Beans for Java (HABJ) provide core components that can be used to develop an entry-level terminal emulator.

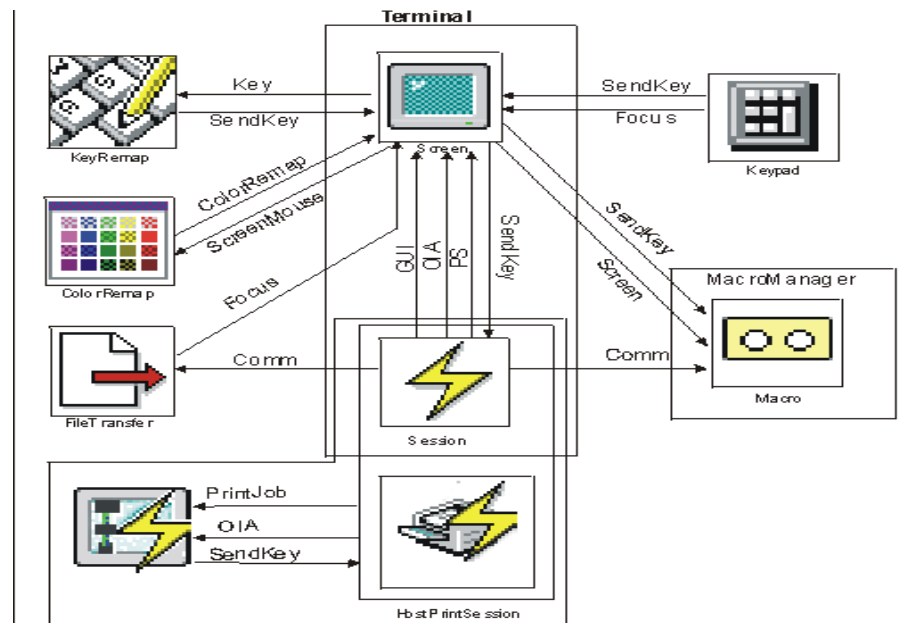














Figure 21-2 Host Access Beans for Java overview

However, wiring the HABJ components together will not provide all of the functionality and usability of the Host On-Demand sophisticated emulator. The purpose of the Host Access Beans is to provide developers with an elementary set of GUI and emulator sub functions that are sufficient to support the main business logic and presentation needs of the deployed custom application. The deployed application should only need elementary host GUI presentation to guide the human user. A custom application that needs the full value-add functionality of an advanced terminal emulator should be designed to use Host On-Demand directly and its RunApplet facility.

The beans available in Version 7.0 of the Host Access Toolkit are listed in Table 21-2:

Table 21-2 Host Access Beans for Java

Icon	Description
Session 	Session is a non-visual bean that provides methods and properties for setting up and establishing communications with the host system. The Session bean fires events that allow listeners to be notified of presentation space, operator information area (OIA), and communication changes.
Screen 	Screen is a visual bean that provides the graphical interface for displaying the host data from a Session bean. The Screen bean listens to presentation space, OIA, and GUI events fired by the Session bean and interprets the events to display the main Presentation Space ("green screen") and the operator information area (OIA). It fires keystroke events to registered listeners, and also provides the clipboard cut, copy, and paste functions.
Terminal 	Terminal bean is a visual bean that combines the Session and Screen beans to provide a composite bean that encompasses both the communication with the host and the graphical interface for displaying the host data.
Keypad 	Keypad is a visual bean that provides a simple grid of buttons representing specialized keys that invoke various host functions.
KeyRemap 	KeyRemap is a visual bean that provides keyboard remapping capability. Using KeyRemap, keystrokes can be mapped to alternate characters, strings, macros, or directly to host functions.
FileTransfer 	File Transfer is a visual bean that provides a toolbar interface for transferring files to and from a host.
Macro 	Macro is a nonvisual bean that records and plays a single macro. Macro employs advanced screen recognition technology to reliably navigate host applications in any environment. Macro also provides the ability to prompt for user input and extract text from the screen during playback.
MacroManager 	MacroManager is a visual bean that provides a toolbar interface for managing multiple macros. It allows you to record, play, load, delete, and edit macros.

Icon	Description
	ColorRemap is a visual bean that provides a simple interface for modifying the colors displayed by the Screen or Terminal beans. The ColorRemap bean is only supported in Host On-Demand.
	HostPrintSession is non-visual bean that extends the Session bean and provides a simple interface for creating and customizing 3270 and 5250 printer sessions.
	HostPrintTerminal extends the HostSessionBean and provides a simple interface for creating and customizing 3270 and 5250 printer sessions. At run-time, the HostPrintTerminal bean visually displays information about the status of the print jobs and the connection with the host.
	The Converter beans performs codepage-to-codepage conversion. For the Arabic and Hebrew languages, Converter performs certain BIDI-specific transformations, including logical-to-visual transformations, and Lam-Alef processing (Arabic only).

21.5.1 Host Access Java Beans explained

This section describes the above-mentioned beans in detail.

Session bean

This is a non-visual bean and has a customizer. It provides methods and properties for setting up and establishing communications with a host system and fires events that notify listeners of presentation space (PS), operator information area (OIA), and communication changes. It determines the behavior and characteristics of the session with the host through its properties, which include the session type (3270, 5250, and VT), host, port, session ID, PS size (for example, 24 rows by 80 columns), and the host code page.

Screen bean

This is a visual bean and has a customizer. It provides the graphical interface for displaying the host data from a Session bean. The Screen bean listens to PS, OIA, and GUI events fired by the Session bean and interprets the events to display the main text area and the OIA. It fires keystroke events to registered listeners, and also provides the clipboard cut, copy, and paste functions.

This bean is sensitive to both the session type and code page and has different behaviors for the different session types and for single-byte, double-byte, and bi-directional code pages. The OIA displays different information according to the session type and code page.

Among its properties are:

- ▶ The `sessionType` property that affects its display of host data. The `sessionType` property is an enumeration for which the valid values are 1 for 3270, 2 for 5250, 3 for VT, and 4 for CICS.
- ▶ The `oiaVisible` boolean property. This causes the OIA to be turned on or off.

Terminal bean

This visual bean combines the Session and Screen beans to provide a composite bean that encompasses both the communication with the host and the graphical interface for displaying the host data.

Keypad bean

This visual bean provides a simple grid of buttons that invoke various host functions.

A user can execute keyboard functions and send aid keys (such as PF1) to the host with the click of a mouse. The KeyPad can be represented as either two horizontal or two vertical rows of buttons on a single panel depending on how it is configured. The KeyPad comprises two pads that are toggled via the NextPad button on the pad itself; it can also be displayed with radio buttons for switching between the pads.

KeyPad is sensitive to both the session type and code page and will display a different pad for 3270, 5250, CICS and VT sessions and for single-byte, double-byte, and bi-directional code pages.

Among its properties is a `shape` property, which determines whether it takes a vertical or horizontal format. The `shape` property is an enumeration for which valid values are `S2X11` and `S11X2`.

KeyRemap bean

This visual bean allows users to remap keystrokes to host functions or aid keys. To configure the key mappings, the KeyRemap bean must be displayed and have focus. When a key is pressed, the user is prompted to select a function to which the key stroke will be mapped. Thereafter, the KeyRemap bean intercepts the standard Java `KeyEvents`, which are fired from the Screen or Terminal bean, and remaps key values that are re-fired as `SendKeyEvents` back to either the Screen or the Terminal bean.

KeyRemap is sensitive to both the session type and code page and allows keys to be mapped to different sets of functions for 3270, 5250, CICS, and VT sessions and for single-byte, double-byte, and bi-directional code pages.

Its properties include a `sessionType` property that affects the set of functions to which keys can be remapped. This property has the same values as the one described above in the section on the Session bean.

FileTransfer bean

This visual bean provides a toolbar interface for transferring files to and from a host. Its properties consist of the default host type and data type. The developer can set these and other properties through its customizer for subsequent modification by the end user.

Macro bean

This non-visual bean records and plays a single macro. It also provides the ability to prompt for user input and extract text from the screen during playback. It must be connected to either a Terminal or a Session bean.

In addition to sending keystrokes, Macro also handles waiting for the host in between key stroke sequences (usually after an aid key). This works in two ways. First, there is a standard wait, whereby Macro will perform a reliable smart-wait that is automatically inserted during recording. These smart-waits take advantage of the Host Access Class Library's screen recognition technology. The second type of wait is a smart wait defined by the user during recording. When the macro is played back, execution will be suspended until the screen described by the user appears. The screen description can be encoded according to the number of fields, number of input fields, and a keyword.

Macro can prompt the user for strings at runtime. When a macro starts, it is scanned for prompt lines; all prompts are presented immediately. Macro can also extract data from the presentation space. During macro recording, the user can specify a rectangular area of the host screen to be extracted. When the macro runs, text in the bounding rectangle is retrieved and loaded into an array of strings. This array is fired in a `MacroExtractEvent` where any listeners can use the data as they choose. This event is synchronous and does not return until the listener handles it.

Also refer to “More about Macro bean” on page 766.

MacroManager bean

The MacroManager bean can do everything that the Macro bean can do and more. It does not, despite its name, manage Macro beans; it provides a toolbar interface for managing multiple macros. Connected to either a Terminal bean or a Session bean, the MacroManager bean allows you to record, play, load, delete, and edit macros.

You can do simple or advanced recording of a macro. With simple recording, general and reliable screen recognition waits will be inserted automatically into the macro. These screen recognition waits take full advantage of the IBM Host Access Class Library's screen recognition technology.

With advanced recording, you can specifically tailor the screen recognition waits to your host application screens. You can define a screen's characteristics by the number of fields, number of input fields, a keyword, and the operator information area state.

You can also insert prompts into the macro with advanced recording. These prompts can be defined as either password or normal display and automatically place the text that the user inputs into the prompt window during playback. Another feature of advanced recording is inserting extracts into a macro. When the extract button is toggled down, you can mark any part of the presentation space of the Terminal or Session bean you are wired to, and the area marked will be retrieved and fired in an event during playback. A listening bean can capture this data and process it as required.

To manage the persistent storage of macros, you should create an object that implements the MacroIOProvider interface. The MacroIOProvider is responsible for listing all available macros and saving and retrieving those macros from persistent storage.

If you do not implement MacroIOProvider in your code, MacroManager will go into its default mode of saving and retrieving macros from the home directory in which it was loaded.

Also refer to "More about MacroManager" on page 768.

ColorRemap bean

This visual bean provides an interface for modifying the colors displayed by the Screen or Terminal beans. The host screen consists of fields. The field types are defined by the host type-3270, 5250 or VT. Each field type has its default color. Color remap bean allows the color of a field type to be modified to another color.

When a color is changed through the bean, a `ColorRemapEvent` is fired to the registered `ColorRemapListeners`. `ColorRemap` also listens to `ScreenMouseEvents`. When a `Screen` or `Terminal` bean fires `ScreenMouseEvents`, `ColorRemap` automatically displays the correct foreground and background colors for that location.

HostPrintSession

This non-visual bean is an extension of the `Session Bean` and is used to establish custom print sessions for a host 3270 or a 5250 printer session. It defines the behavior and characteristics of the print session with the host.

This bean provides an extended interface of setter-getter calls to programmatically control the printer device represented by the bean

HostPrintTerminal

This visual bean is wrapped around the `HostPrintSession Bean` and is used to establish connection with a host for 3270 and 5250 printer sessions and present the end user with a graphical image that reflects the state of this printer session

Converter

The `Converter` bean performs a codepage-to-codepage conversion. For the Arabic and Hebrew languages, `Converter` performs certain BIDI-specific transformations, including the logical-to-visual transform...or vice-versa, and Lam-Alef processing (Arabic only).

The `Converter` does not use the standard JVM converters. Instead it uses HOD-supplied converters. Here is the list of supported codepages:

Table 21-3 Converter Code Pages

Country	Code Page
Latin	8859_1, 037, 1046, 1047, 1140, 1146, 1148, 285, 437, 500, 850, 924
Western Europe	1141, 1142, 1143, 1144, 1145, 1147, 1149, 273, 274, 275, 277, 278, 280, 284, 297, 871
Eastern Europe	8859_2, 1153, 852, 870
Arabic	8859_6, 1256, 420, 864
Greek	8859_7, 869, 875
Hebrew	8859_8, 1255, 424, 803, 856, 862
Hindi	1137
Japanese	1390, 1399, 290, 930, 939, 942
Korea	1364, 933, 949
Russian	8859_5, 1025, 1112, 1122, 1123, 1154, 1156, 1157, 1158, 855, 866
Thai	1160, 838, 874
Turkish	8859_9, 1026, 1155, 857
China	1381, 1388, 935
China Taiwan	937, 948, 950, 964, 1371

Two Code Samples of the Conversion Bean

Using Conversion Bean

```
package com;
```

```
import com.ibm.eNetwork.beans.HOD.Session;
import java.io.*;
```

```
public class TestConverter{
```

```

    public static com.ibm.eNetwork.beans.HOD.cpc.Converter
        c = new com.ibm.eNetwork.beans.HOD.cpc.Converter();

    public static void main(String[] Args){

        c.init(); //set default attributes

        //set general attributes

        c.setRecordLength(80);
        c.setBinMode(true);

        c.setInputFileName("input.txt");
        c.setInputCodepage("Cp420");
        c.setInputHostType(c.OS390);

        c.setOutputFileName("output.txt");
        c.setOutputCodepage("Cp1256");
        c.setOutputHostType(c.WIN);

        //set BIDI-specific attributes

        c.setInputTextType(Session.VISUAL);
        c.setOutputTextType(Session.LOGICAL);
        c.setInputTextOrientation(Session.LEFT_TO_RIGHT);
        c.setOutputTextOrientation(Session.LEFT_TO_RIGHT);
        c.setSymSwap(false);

        //set Arabic-specific attributes

        c.setNumeralShaping("NOMINAL");
        c.setLamAlef(false);

        //perform the conversion

        c.processConversion();
    }

```

Example 21-1 Example of Conversion Bean

```

package com;
import com.ibm.eNetwork.beans.HOD.cpc.*;

import java.awt.*;
import java.io.*;
import java.util.*;

```

```

import java.awt.event.*;
import com.ibm.eNetwork.beans.HOD.event.ConvertListener;
import com.ibm.eNetwork.beans.HOD.event.ConvertEvent;
import com.ibm.eNetwork.beans.HOD.Session;

public class TestConverter1 implements java.awt.event.ActionListener
{
    private static Vector ConvertListeners;
    public static ActInfo  currentActInfo;
    public static Converter c;

    public void addConvertListener(ConvertListener l) {
        ConvertListeners.addElement(l);
    }

    public void removeConvertListener(ConvertListener l) {
        ConvertListeners.removeElement(l);
    }

    private void fireConvert(ConvertEvent cevent) {
        if (ConvertListeners != null) {
            Vector l;
            synchronized(this) {
                l = (Vector)ConvertListeners.clone();
            }
            for (int i = l.size() - 1; i >= 0; i--) {
                ((ConvertListener)l.elementAt(i)).execute(cevent);
            }
        }
    }

    public void dispose(){
        ConvertListeners.removeAllElements();
    }

    public void startConversion(){
        ConvertEvent cevent=new ConvertEvent(this, "event ID?",
currentActInfo.toString());
        cevent.setActInfo(currentActInfo);
        fireConvert(cevent);
        return;
    }

    public void actionPerformed(ActionEvent e){
        System.out.println("Ready for next conversion");
    }

    public void init(){

```

```
        ConvertListeners = new Vector(1,1);
        addConvertListener(c);
        c.addActionListener(this);
    }

    public static void main(String[] Args){

        currentActInfo = new ActInfo();
        c = new Converter();

        currentActInfo.setRecordLength(80);
        currentActInfo.setBinMode(true);

        currentActInfo.setInputPathName("C:"+File.separator+"input.txt");
        currentActInfo.setInputCodepage("Cp420");
        currentActInfo.setInputHostType(c.OS390);

        currentActInfo.setOutputPathName("C:"+File.separator+"output.txt");
        currentActInfo.setOutputCodepage("Cp1256");
        currentActInfo.setOutputHostType(c.WIN);

        //set BIDI-specific attributes

        currentActInfo.setInputTextType(true);
        currentActInfo.setOutputTextType(false);
        currentActInfo.setInputTextOrient(true);
        currentActInfo.setOutputTextOrient(true);
        currentActInfo.setSymSwap(false);

        //set Arabic-specific attributes

        currentActInfo.setNumeralShaping("NOMINAL");
        currentActInfo.setLamAlef(false);

        TestConverter1 theTest = new TestConverter1();
        theTest.init();
        theTest.startConversion();

    }
}
```

21.5.2 Automated host navigation

Please read 21.4.3, “Automated host navigation” on page 752 as an introduction to conceptions of screen recognition.

More about Macro bean

The Macro bean in the Host Access Beans for Java API is another assistive technology. Macro is a non-visible component, although it is typically used in a GUI application. The Macro bean serves as a tool for the developer (or application at runtime) to define a macro or a script to navigate host screens.

Macro is a scripting-oriented abstraction of the screen recognition technology. The user creates an XML-based script to define screens and the actions associated with those screens. A macro script used by the Macro bean contains three elements:

A screen description	The criteria used to identify a unique screen or screen state.
Actions to take	What to do when the screen is recognized.
Next screen (optional)	A pointer to the next screen definition depending on the result of actions on the current screen.

Scripts are interchangeably executable on the general-purpose scripting engine. The engine can be extended and enhanced, while maintaining backward compatibility to older script syntax. Alternative scripts can be deployed according to scripting engine capabilities. Macro scripts can be serialized or loaded from a text file. A provider interface is available to insulate the scripting engine from the storage or transport mechanism used to load/store macro scripts.

Macro bean uses XML for scripting because a macro is better suited to the state machine model (the main reason for the move: XML is tailor made for a state machine).

The idea of a state machine may be fairly new to you. The idea behind a state machine, especially in the Macro bean context, is simple. Think of how you use a host system from a terminal or a terminal emulator (like Host On-Demand). The process you follow when you interact with a host system is illustrated in these steps:

1. The host sends an expected screen down to you at your terminal.
2. You look at and understand which screen is presented to you.
3. You take the required actions based on your understanding (type keystrokes, and so forth).
4. Another screen is presented after these actions.

5. If you see the screen you expected, repeat steps 2, 3, and 4.
6. If you do not see the screen you expected, call the help desk or handle the error.

This is the idea behind a state machine in the Macro context (although the Macro can't call the help desk for you). The states are the screens you expect to see, and you take actions on those screens to change from one state, or screen, to another. That's it, see a screen, perform the action, see the next screen. It is easier to understand (and program) a macro with this approach than having several if-then-else and do-while programming statements. Remember, see a screen, perform the action, see the next screen.

The following are valid macro elements. The tag names are suggestive of their function:

<HAScript>

<vars>

<create>

<screen>

<comment>

<description>

<oia>

<cursor>

<numfields>

<numinputfields>

<string>

<attrib>

<customreco>

<varupdate>

<actions>

<prompt>
<input>
<extract>
<message>
<trace>
<filexfer>
<pause>
<mouseclick>
<boxselection>
<commwait>
<custom>
<varupdate>
<playmacro>
<if>
<else>
<runprogram>
<nextscreens>
<nextscreen>
<recolimit>

These XML elements and their attributes are valid in the Host On-Demand Macro XML namespace. This description of the elements is structured like an actual macro file. Element and attribute values are not case sensitive.

More about MacroManager

MacroManager provides an end-user convenient toolbar when it makes sense to let the user load a macro script appropriate to the situation from a library.

MacroManager must have an associated Macro bean. The API allows that association to be set in different ways.

MacroManager on the Host On-Demand toolbar is an excellent way to record macro scripts for using in custom applications programmed with HABJ. There is an excellent designer-editor customer window associated with MacroManager.

Please see Appendix C, “An example of MacroIOProvider” on page 1031 for a demonstration of MacroManger in action in the context of Host On-Demand client.

HOD V7 vs. PCOMM V5.6 Macro

The HACLJ and HABJ Java code level of Personal Communications V5.6 is at the level of Host On-Demand 4.0. Host On-Demand 7.0 contains improvements and extensions in the Macro bean. If you made a strict comparison, the differences might present incompatibilities: all of Personal Communications V5.6 code compiles using the Host On-Demand 7.0 JARs, but the reverse is not true. HOD V7.0 has extensions and features not found in PCOMM V5.6 of Macro.

Personal Communications V5.6 desktop emulator has a macro facility available on its toolbar. The underlying mechanism and scripting languages bear no relation to the Host On-Demand-descendant Java Macro bean.

21.6 Java 2

Since the release of JRE 1.1.8, Sun has introduced several new technologies and methodologies that should be addressed to ensure a properly designed, implemented, and executed applet or application. IBM urges all developers to use a 1.3.x JRE or higher for any technologies that were added after the JRE 1.1.8 specification.

The Host Access Beans for Java now supplied as separate library sets targeted for JDK1.1.8 or Java2 (JDK 1.3.0). The debug and release libraries are installed in these directories:

<install dir>\toolkit\jars\jdk1.1

<install dir>\toolkit\jars\java2

The files supplied include a 2 suffix when the files are unique for Java 2. For instance habasen.jar is for jdk1.1 and habasen2.jar is for Java2

The Java 2 JAR libraries can only be run in a browser that has the Java 2 plug-in enabled or launched in a Java 2 virtual machine. The Java 2 version is Swing-based and should integrate well with other Swing components. There are known issues when building applications or applets that use both AWT components and Swing components. For more information on compatibility between AWT and Swing components, see the Sun Java Web site, java.sun.com.

The Java 2 version of the Host Access Beans offers accessibility, autoIME/on the spot conversion for double-byte character set (DBCS) languages, and print screen enhancements. Pure JDK 1.1 applets and applications that use the Host Access Toolkit JDK 1.1.8 libraries are runtime compatible with the Java 2 Virtual Machine. However, applets and applications built with the Java 2 libraries are strictly incompatible with a JVM version 1.1.

The traditional JDK1.1.8 version of our APIs is most suited for applets intended for JDK1.1-based browsers or for applications launched into a Java Virtual Machine version 1.1. These JDK1.1.8 libraries (JAR and CAB) are compatible with previous versions of the Host Access Toolkit.

21.6.1 Security

Note: Java security is an evolving area. The security model has become more “granular.” Some browser implementors took an individual proprietary path. Vocabulary usage has been inconsistent. You, the end user, will have to develop a mental picture that rationalizes all of the apparent confusion.

The classes of the HACLJ and HABJ API libraries were written to be compatible with the native security mechanisms of the two main popular Web browsers: Microsoft Internet Explorer and Netscape Navigator. Those two are JDK 1.1-based. The libraries have subsequently been updated to operate in the Sun Java 2 1.3.x Plug-in environment.

Netscape uses its proprietary netscape.security package. Internet Explorer uses its proprietary com.ms.security package. Sun Java 2 is based on the java.security package. Each is unrelated to the others. Therefore, the library code (and by implication, your code) must be written to accommodate usage in various browser and security environments. The coding of the Toolkit libraries specifically distinguish between Netscape, Internet Explorer and “other.” The “other” implies Java 2.

A library that has been signed digitally may also be “granted” permissions. The “release” versions of the Toolkit libraries have been signed with IBM's certificate and granted “AllPermissions.” Your digitally signed custom libraries should grant the level of permission(s) appropriate to your purpose

A “trusted” action is an operation or external access that is prohibited by default. The requestor is not trusted to perform it, by default. Permission (Internet Explorer) or privilege (Netscape) must be asserted (Internet Explorer) or enabled (Netscape) in order to overcome the default prohibition.

The installed security manager will allow certain prohibited-by-default trusted actions to succeed provided that all of these conditions are met:

1. That the executing code was loaded from a library
 - a. That was digitally signed by a trusted certificate
 - b. And that library was granted sufficient permissions
2. That the executing scope has already asserted enabled the appropriate specific permission(s) or privilege(s)

At certain places in the HACLJ and HABJ code, trusted actions are taken in a certain scope of execution. Prior to that action and within that same execution scope, appropriate privileges must be enabled. The policy rules of the security manager of the Java virtual machine determine what constitutes a trusted action. Scope of execution means that an enabled privilege remains in effect for the duration of the enclosing method call and its sub-method calls; privilege ends when the enclosing method returns, or when the privilege is disabled, whichever occurs first.

If your application code must perform a trusted action, you must code the necessary privilege enablement. In certain situations, a trusted action might originate as a call to a public method in the API library. It must be preceded by the appropriate privilege enablement, which is determined by experimentation.

In the subsequent discussion, the example trusted action is an instance of an applet class creating or truncating a file on the local hard drive and writing to it. Normally the “sandbox” created by the browser security manager prevents that action (it is a “trusted” action). Therefore, your code must specifically enable the

appropriate privilege, the security manager must agree that the privilege is the correct one, and the security policy in force must allow that privilege to be enabled in the runtime code. Additionally your code must load from a digitally signed library that has been granted sufficient permissions. If any of these conditions are not met, then the trusted action will fail with a characteristic exception being thrown.

“Digital signing”, or “code signing”, is the practice of placing an unalterable, authentication “stamp” on a code library. This stamp asserts that the encompassed library is actually developed by a particular organization. To the extent a user trusts the organization, the end user will trust the library and allow all requested instructions to occur. Java supports this security model because it guards against running malicious code (such as inappropriately erasing files) provided by organizations and entities unfamiliar or untrusted to the user. To sign code, one must obtain an authentic certificate from a Certificate Authority such as Thawte Consulting or VeriSign, Inc. These certificates are recognized by the browser or plug-in and are presented to the user for approval or rejection of the requested privileges.

The security manager

Internet Explorer running with the Microsoft Java virtual machine installs its own security manager. Likewise, Navigator installs its own security manager. When appletviewer.exe is run, it installs its own security manager, which cannot be disabled. Usually when an application is started from the command line, there is no security manager installed by the virtual machine. Under Java 2, there is a command-line flag for java.exe, `-Djava.security.manager`, which signals the virtual machine to install the Sun Java 2 security manager before executing code.

When a JVM does not have a security manager installed, all trusted actions are automatically enabled. Failure of a trusted action, if it occurs, will be on the merits and not on permissivity. This mode is equivalent to a superuser. It is an excellent mode for code logic and flow development, but unsuited to deployment of a mission-critical application. The implementation for the application design will require identification of all trusted actions and accommodation to the security policy of the deployed application.

Netscape 4.7

When Navigator loads and executes an applet class, it searches the libraries named in the ARCHIVE parameter of the APPLET tag and searches for “loose classes” (see “Effect of “Loose Classes”” on page 780) in the folder where the HTML file is located. A packaged library may be a JAR or a ZIP file. The JAR file may, or may not, be digitally signed.

When code execution arrives at:

```
<PrivilegeManager>.enablePrivilege("<a NS privilege string>");
```

the Navigator security manager will check whether that path-and-distinguished-name-specific class is known to the browser. If it is not known, Navigator will display a window to inform the user of a needed decision. If the user makes a "grant" decision, that is policy momentarily stored by the browser. The user may also mark that policy decision as "remember."

The specific policy decision is stored in a browser database entry in the form of a pair of strings representing "key" and "value". For a trusted, digitally signed JAR, the key is the trusted certificate and the value is the specific privilege. For an unsigned library, the key is the URL of the library and the value is the specific privilege. Normally, an entry in the database lasts only as long as the browser session. A "remembered" entry persists across browser sessions.

Internet Explorer

When Internet Explorer loads and executes an applet class, it searches the libraries named in the "cabinets" PARAM tag within the scope of the APPLET tag. The library format is CAB, which is created by the Microsoft cabarc.exe tool. The CAB may be signed with the signcode.exe tool.

When code execution arrives at:

```
PolicyEngine.assertPermission(PermissionID.<permission mnemonic>);
```

Internet Explorer does not offer the user an opportunity to decide policy. If the CAB library is not digitally signed by a trusted certificate, the assertion is denied and the subsequent trusted action fails with a characteristic exception thrown.

Normally Internet Explorer does not use JAR or ZIP files, unless they are path-specifically mentioned in this Windows Registry entry:

```
HKLM\Software\Microsoft\Java VM\TrustedClasspath
```

That entry is a global CLASSPATH and the enumerated JARs, ZIPs and class files are granted all permissions (superuser). A CAB file, signed or unsigned, named in that Registry entry is disregarded by the IE security manager.

Java 2 Plug-in

The Java 2 security manager responds to two policy files. See Sun's documentation at:

<http://java.sun.com/docs/books/tutorial/security1.2/summary/tools.html>

We are interested in a user's ".java.policy", which is the target of the Java 2 policytool.exe. The installation of the Plug-in JRE does not normally install .java.policy.

The purpose of the .java.policy file is to express the baseline permissions of the virtual machine sandbox whenever the Java 2 virtual machine is called into action for that logged in user. The Java 2 Plug-in can be parameterized (on Windows) to be the default JVM for Netscape 6.2 and for Internet Explorer 5.+.

When a trusted action call is made, the Java 2 security manager relies on the permissions in the .java.policy file. If there is no .java.policy file, the programmer may have used Java 2 syntax to assert a permission. Absent that, the Java 2 security manager displays a window to inform the user of the permission request and asks for a GRANT type reply. This happens only once on the first encounter to a trusted action. GRANT can last just for the Plug-in session or “always.” All trusted actions are covered by this GRANT action. An approved GRANT, which was indicated to the browser as one to remember, can be revoked by removing the mentioned certificate copy from the Java 2 Plug-in properties window in the Services folder.

Additional information about Java 2 security

We recommend two books dealing with Java 2 security:

- ▶ *Java 2 Network Security* by M. Pistoia et al., June 1999, IBM Form Number: SG24-2109-01, ISBN: 0-130-15592-6, covers many topics of interest to Java programmers dealing with the practicalities of security programming and digital signing. It was written when the Java 2 security API was in its formative stage.
- ▶ *Java Security: 2nd Edition* by Scott Oaks, May 2001, O'Reilly & Associates, Inc. ISBN: 0-596-00157-6, is an exhaustive presentation of the Java 2 security model and associated Java technologies.

21.6.2 Permissions programming examples

The sample code shown in Example 21-2 illustrates alternative permissions codings. Example 21-3 shows a sample HTML page to execute the applet shown in Example 21-2.

Example 21-2 Permissions programming example

```
package trustedaction;
import java.awt.*;
import java.awt.event.*;
import java.applet.*;
import java.io.*;
import netscape.security.*;
import com.ms.security.*;
import com.ibm.eNetwork.HOD.common.*; // Obtain Environment.class for
detecting the browser
```

```

/**
 * AS-IS sample : not intended as example of best Java practices
 *
 * To compile, use \hostondemand\HOD\hoddhg.jar, java40.jar from NS 4.7 and
 com.ms.security.*
 */
public class TrustedActionApplet extends Applet {
    // borrowed from HOD: capable of detecting browser
    Environment env = Environment.createEnvironment();
    boolean isStandalone = false;
    /**Get a parameter value*/
    public String getParameter(String key, String def) {
        return isStandalone ? System.getProperty(key, def) :
            (getParameter(key) != null ? getParameter(key) : def);
    }
    /**Construct the applet*/
    public TrustedActionApplet() {
    }
    /**Initialize the applet*/
    public void init() {
        try {
            jbInit();
        }
        catch(Exception e) {
            e.printStackTrace();
        }
    }
    /**Component initialization*/
    private void jbInit() throws Exception {
        this.setBackground(Color.red);
        // is it Internet Explorer?
        if
(Environment.getUseSecurityManager().equals(Environment.SECURITY_MGR_IE)) {
            System.out.println("Detected IE");
            jbInit_IE();
        }
        else //is it Netscape?
        if
(Environment.getUseSecurityManager().equals(Environment.SECURITY_MGR_NS)) {
            System.out.println("Detected NS");
            jbInit_NS();
        }
        else { // will assume that it is Java2 PlugIn
            System.out.println("Fell through to OTHER");
            openFile();
        }
    }

    // Internet Explorer proprietary PolicyEngine and Permission.ID
    private void jbInit_IE() throws Exception {

```

```

        System.out.println("ATTEMPTING:
PolicyEngine.assertPermission(PermissionID.FILEIO");
        PolicyEngine.assertPermission(PermissionID.FILEIO);
        System.out.println("SUCCEED:
PolicyEngine.assertPermission(PermissionID.FILEIO");
        openFile();
    }
    // Netscape 4.+ proprietary PrivilegeManger and privilege string
    private void jbInit_NS() throws Exception {
        System.out.println("ATTEMPTING:
<PrivilegeManager>.enablePrivilege(UniversalFileWrite)");
        netscape.security.PrivilegeManager pm = new PrivilegeManager();
        pm.enablePrivilege("UniversalFileWrite");
        System.out.println("SUCCEED:
<PrivilegeManager>.enablePrivilege(UniversalFileWrite)");
        openFile();
    }
    // Assuming that Java2 PlugIn and .java.policy file controls
    private void openFile() throws Exception {
        System.out.println( "OPEN FILE");
        FileOutputStream fos = new FileOutputStream("c:\\F020117.txt");
        PrintStream prtS = new PrintStream(fos);
        prtS.println( "The TIME=" + System.currentTimeMillis());
        fos.close();
        prtS = null;
        fos= null;
    }

    /**Start the applet*/
    public void start() {
    }
    /**Stop the applet*/
    public void stop() {
    }
    /**Destroy the applet*/
    public void destroy() {
    }
    /**Get Applet information*/
    public String getAppletInfo() {
        return "Applet Information";
    }
    /**Get parameter info*/
    public String[][] getParameterInfo() {
        return null;
    }

```

Example 21-3 Sample HTML page

```

<HTML>
<HEAD>
<META HTTP-EQUIV="Content-Type" CONTENT="text/html; charset=windows-1252">
<TITLE>
HTML Test Page
</TITLE>
</HEAD>
<BODY>
trustedaction.TrustedActionApplet will appear below in a Java enabled
browser.<BR>
<APPLET
  ARCHIVE   = "hoddgbg.jar,appletCode.jar"
  CODEBASE  = "."
  CODE      = "trustedaction.TrustedActionApplet.class"
  NAME      = "TestApplet"
  WIDTH     = 400
  HEIGHT    = 300
  HSPACE    = 0
  VSPACE    = 0
  ALIGN     = middle
>
<PARAM NAME="cabinets" VALUE="hoddgbg.cab,appletCode.cab">
</APPLET>
</BODY>
</HTML>

```

Example 21-4 is a sample of how to compile the example.

Example 21-4 Compile the example

```

SET JDK=%ibmjdk1%
%JDK%\bin\javac -d . -classpath
hoddgbg.jar;.\PolicyEngine;.\PrivilegeManager\java40.jar;%JDK%\lib\classes.zip;.
trustedaction\TrustedActionApplet.java

```

21.6.3 Debugging runtime failures

This section discusses how to distinguish between security exceptions and programming bugs as the cause of runtime failures.

Messages sent to the Java console are an important source of information in developing and debugging Java applets and applications. When the program execution is started from a command-line window, that window usually serves as the Java console display area. For applets running in a browser, there is a menu bar option to display the Java console associated with the browser's native Java virtual machine (JVM). In the case of a browser employing the Sun Java 2 Plug-in, the Java console is accessed via the runtime icon for the JVM. In other words, when an applet is loaded, the browser first loads the Plug-in before starting the applet. Usually an icon appears in the desktop system tray to represent the Plug-in process. That icon gives menu access to the Java console.

Java methods are often designed to throw an exception for error conditions. The try-catch control block is used to catch exceptions at strategic points for the purpose of recovering from an error or to exit gracefully. Try-catch blocks are important debugging points where the "printStackTrace" method can be called. A stack trace can be obtained at any point with a "Thread.dumpStacktrace();" statement.

When a "trusted" action is performed, it will succeed provided the proper permission is in effect and the security manager concurs. An exception will be thrown by the security manager (if it is installed) when a "trusted" action is attempted without sufficient permission(s).

Suppose you have developed an applet in such a manner that it can also be run as a stand-alone application (it has a "main(...)" entry point). A puzzling situation arises. When the application is run from the command line, everything works as expected. In contrast, when that same applet is run in a browser via an HTML file, there are exceptions being reported on the Java console. In the case of a HAC LJ or HABJ applet, a connection to the target host cannot be established. The exceptions all seem to originate deep in the code of the HAC LJ/HABJ classes. Are these unexpected bugs in the API library?

If this failure is due to insufficient or incorrect permissions exerted, they will disappear when the applet is run in a "superuser" mode. If these are determinate bugs in the library, then even in superuser mode they will persist.

Under the Java 2 security model

AllPermissions, in the Java 2 Security model, is the equivalent of a superuser. Any trusted action is permitted always. Therefore, if your applet in a browser were to run as a superuser, only bugs would be the source of exceptions. It is simple to create the superuser mode under Java 2.

Under the Java 2 Security model, JVMs are traditionally configured with a `java.policy` file. This file identifies which permissions a class, be it an applet or application, may obtain from the JVM. All Java 2 SDKs contain a `policytool.exe` to modify these settings. See the Sun document at:

<http://java.sun.com/docs/books/tutorial/security1.2/summary/tools.html>

For example, if you wanted to effectively turn off security for all classes on the C drive, you would:

- ▶ Launch `(JRE)\bin\policytool.exe`. If a window appears stating that the `.java.policy` file cannot be found, write down the location it attempted to read from.
- ▶ Click **Add Policy Entry**. Specify the codebase as `file:/C:/-` (Note the required type of slash even for Windows.)
- ▶ Click **Add Permission**. From the Permission pull-down, select **AllPermission**. Click **OK** to close both windows.
- ▶ Save the file. If creating a new file, use the location specified when you first opened `policytool.exe`.

For all JVMs that use this policy file, “all permissions” (superuser privilege) will be granted to any class located on the C drive. This mechanism works identically for Netscape and Internet Explorer when Sun's 1.3.1 Plug-in supplies the JVM.

Under the Internet Explorer TrustedClassPath model

The Windows Registry entry

`HKLM\Software\Microsoft\Java VM\TrustedClasspath` can be very helpful in quickly creating a “superuser” mode. Any fully qualified JAR or ZIP file path name in that entry is:

1. A global `CLASSPATH` library, and
2. Granted all-permissions for any trusted action performed by code in that library

Typically an HTML file for Internet Explorer will have the following line:

```
<PARAM NAME="cabinets" VALUE="<list of CAB files>"
```

Remove that line. As a further simplification, remove these typical lines:

```
CODEBASE = "."
ARCHIVES = "<list of JAR and ZIP file>"
```

Next, use **Regedit** to modify the Windows Registry (carefully). In the key `HKLM\Software\Microsoft\Java VM\TrustedClasspath`, write a semi-colon separated list of full path names to all needed JARs and ZIP file.

The use of CAB files in an Internet Explorer context includes, but goes beyond packaging Java classes. This is an entry point Microsoft Knowledge Base webpage to cover the entire topic:

HOWTO: Make Your Java Code Trusted in Internet Explorer

<http://support.microsoft.com/default.aspx?scid=KB;EN-US;q193877&>

Under LINUX

Netscape 6.x with Sun Java 1.3.x Plug-in is available for Linux; therefore, the appropriate information in “Under the Java 2 security model” on page 778 applies here.

Effect of “Loose Classes”

We define “loose classes” as a folder hierarchy of classes partially duplicating the content of digitally signed JAR libraries. Suppose the JVM searches the loose classes first and then the signed JAR libraries. In an extended search for members of a given package, it may report (typically) a `NullPointerException` or `ClassDefNotFound` exception. The underlying reason would be that members of the same package are located in two “libraries” that do not have the same digital signature. Loose classes cannot be signed; therefore their “signature” is different. Consolidation and unified signing is the solution.

Remember that these loose classes themselves have no digital signature and no assigned level of “permissions granted.” If one of those classes attempts a “trusted” call, even if the code exert a “permission;” it will fail with a security exception.

Environmental CLASSPATH failures

Some development environments and installed applications set the local user or global environmental CLASSPATH variable. Suppose that your application launch command also enumerates the %CLASSPATH% environmental variable. When you attempt to run your application in a local hard drive, the JVM may inadvertently be searching duplicate but different versioned package libraries. That may cause unexpected runtime failures or component misbehaviors.

21.7 Host On-Demand EHLLAPI Bridge

The IBM Host On-Demand EHLLAPI Bridge (EHLLAPI BRIDGE) is a technology that allows an application written using a variety of HLLAPI-based languages to use Host On-Demand as the target emulator. There is a large base of existing EHLLAPI code developed before the advent of Java and Host On-Demand. Much new EHALLPI code is being developed. The benefit of the EHLLAPI Bridge is

that client applications can readily migrate from one emulator, such as Personal Communications or Attachmate, to Host On-Demand. Sometimes the original EHLLAPI binary can be used directly, if no minor code adjustments are needed for Host On-Demand compatibility. This technology is not shipped with either Host On-Demand or the Host Access Toolkit. The packaged file, ehllapi.exe, can be downloaded by registered users free of charge from the following Web site:

<http://www.ibm.com/software/webrowsers/hostondemand/downloads.html>

21.7.1 Operational configuration

The typical EHLLAPI custom application is written in a Win32 API-based programming language for the Windows operating system, such as C/C++ or Visual Basic or REXX. The application is used in conjunction with one or more emulator sessions running on the desktop. The EHLLAPI program communicates with the emulator program via Windows and OS/2 compatible DLLs. The emulator window opens and maintains a session with the target host. The stand-alone EHLLAPI program uses a set of EHLLAPI API functionalities to interact with the host terminal program presented in the emulator session window. EHLLAPI API programming is discussed in detail with code examples in the *Emulator Programming* document (pcep.pdf), available in the companion Personal Communications V5.6 product.

21.7.2 Supported interfaces

These are the supported interfaces:

- ▶ Industry Standard EHLLAPI 16-bit
- ▶ Industry Standard EHLLAPI 32-bit
- ▶ IBM Enhanced EHLLAPI 32-bit
- ▶ WinHLLAPI 16-bit
- ▶ WinHLLAPI 32-bit
- ▶ IBM REXX EHLLAPI
- ▶ DOS HLLAPI

21.7.3 Supported Platforms and JVM environments

The platforms supported are:

- Windows 95
- Windows 98
- Windows Me
- Windows NT (at least SP5 and preferably SP6a)
- Windows 2000 (preferably with the latest Service Pack)

If the HLLAPIEnabler Applet fails to load and the Java console shows an error saying `RNIGetCompatibleVersion` not found, this means the user does not have the appropriate level of the Microsoft VM. For most MS VM updates please visit:

<http://windowsupdate.microsoft.com>

As of this writing there is a special situation for users of Internet Explorer on Windows XP. The details are presented here:

<http://www.microsoft.com/java/xp.htm>

21.7.4 Installation

Provision is made for running existing code written for Windows and also written for the earlier DOS environment.

Windows installation

To automatically install the EHLLAPI enablement software on a Windows workstation using InstallShield, you must log in as Administrator or a user that is a member of the Administrators group before you can install the EHLLAPI Bridge on your computer. Locate the `Ehllapi.exe` installer file that was downloaded from the webpage mentioned above. Double click on the icon and choose all default values during the installation process, unless a different installation location is desired. Restarting your computer is necessary, except for Windows 2000 and Windows XP.

Upon restart the `PATH` variable value may be similar to this example from Windows 2000. Notice that the system search path finds the EHLLAPI Bridge folder first.

Tip: Check the content of the System PATH environmental variable. In the case of the default installation location, "C:\Program Files\IBM\EHLLAPI;" should precede the path location of any other Win32API-based emulator product, such as Personal Communications. The reason is that the DLLs installed for EHLLAPI Bridge have the same standard names as the DLLs of EHLLAPI-capable emulator products. For this bridge to work in conjunction with Host On-Demand, the Windows operating system must discover the EHLLAPI Bridge DLLs before those of the other emulator product. Consequently, if you try to use that other emulator product, it will likely fail with an unexpected error: the Windows system has located the "incorrect DLLs." The implication is that deployment of Host On-Demand EHLLAPI Bridge may require the removal of other Win32API-based emulator products so as to avoid "DLL confusion"

Configure a Host On-Demand emulator icon. In the Properties window for any sessions to be used with EHLLAPI applications, do the following on the Advanced tab:

- ▶ Set the Auto-start pull-down to Applet.
- ▶ Type in the Auto-Start Name Field this class name:
com.ibm.eNetwork.h11bridge.HLLAPIEnabler.
- ▶ Alternatively, you may run this applet after the session has been started by clicking **Assist --> Run applet** from the session menu bar.

DOS installation

The EHLLAPI Bridge can be used to run DOS EHLLAPI programs but it requires some additional setup. The following assumes the default installation location of the EHLLAPI Bridge:

C:\Program Files\IBM\EHLLAPI

Windows NT

To Enable DOS EHLLAPI Programs on Microsoft Windows NT:

- ▶ The additional binary files are located in:
C:\Program Files\IBM\EHLLAPI\DOSHLLAPI\NT.
- ▶ Place the file HLLDRV.RSYS in your "windows directory"\System32\Drivers subdirectory.
- ▶ Place the file HLLVDD.DLL in the same directory as the rest of your HLLAPI Bridge DLLs.
- ▶ In the subdirectory "windows directory"\System32 is a file called Config.NT. Modify this file by placing the following line at the end:

```
device="windows directory"\System32\Drivers\HLLDRVR.SYS
```

- ▶ Restart your computer.

The bridge will now work without any further special actions.

Windows 9x

To Enable DOS EHLLAPI Programs on Microsoft Windows 95 or Microsoft Windows 98:

- ▶ The additional binary files are located in:

```
C:\Program Files\IBM\EHLLAPI\DOSHLLAPI\Win9x.
```
- ▶ Place the file DOSHLL.VXD in your "windows directory"\System subdirectory.
- ▶ Place the file DOSHLL.EXE in the same directory as the rest of your HLLAPI Bridge DLLs, such as C:\Program Files\IBM\EHLLAPI.
- ▶ In the "windows directory" is a file called System.INI. Modify this file by placing the following line at the end of the section headed by "[386Enh]":

```
device="windows directory"\System\doshll.vxd
```
- ▶ Restart Windows.

To use the bridge for DOS EHLLAPI programs, start the program DOSHLL.EXE and then run your program.

21.7.5 Operation

First start one or more properly configured Host On-Demand emulator sessions. Then start the EHLLAPI application. The EHLLAPI program should run normally and expected host actions should occur in the Host On-Demand emulator window.

Known limitations

The EHLLAPI Bridge is an evolving tool with certain known limitations:

- ▶ Structured Fields, related functions (120-127) are not supported.
- ▶ WinHLLAPI Extensions for Asynchronous calls and blocking functions are not supported.
- ▶ LockPS(60)/LockWindowServices(61) are not supported (used when multiple apps are connected to the same session).
- ▶ StorageManager(17) is supported for WinHLLAPI only.
- ▶ Screen Customizer and the EHLLAPI Bridge are not a supported combination.

If any of the above unsupported options are used by the invoking application, a message window will be displayed notifying the user that the application may not work as expected.

Additionally, the following restrictions apply:

- ▶ SendFile(90)/ReceiveFile(91) - not supported for 5250 sessions.
- ▶ SetSessionParms(9) EAD/NOEAD,SO/NOSO/SPACESO - these DBCS-only parameters are not supported.
- ▶ EXTENDPS/NOEXTENDPS (5250 only) - CopyPS, CopyPSToString, CopyStringToPS, CopyStringToField, CopyFieldToString, and SearchField - will always return any messages on line 24 (like EXTEND_PS), but will never return a 25th line (such as NOEXTEND_PS).
- ▶ SUPER_WRITE, WRITE_SUPER, WRITE_WRITE, WRITE_READ, WRITE_NONE, READ_WRITE - these parameters will be ignored; standard EHLLAPI supports only one application connection to a session at a time.
- ▶ NOBLANK - this parameter will be ignored, standard EHLLAPI default of BLANK will always be used.
- ▶ KEY\$xxxx - this parameter will be ignored.
- ▶ Prior to the loading and availability of Host On-Demand 4.0 CSD 1, closing a Host On-Demand session with the X button in the upper right-hand corner of the Host On-Demand Session Frame can have the side effect of requiring a Browser restart.
- ▶ To support CICS and VT gateways, it will be necessary to obtain and load Host On-Demand 4.0 CSD 1 or later. Prior to the availability and loading of this level of code, using the CICS or VT gateways (Icon Types) will abort the Bridge enablement. CICS users can use a 3270 connection to a CICS session and everything will work appropriately.
- ▶ REXX support is required for starting and stopping sessions using PCSAPI32.dll.

21.8 Programming notes

The following topics relate to programming aids and license limitations.

21.8.1 JARs and CABs

The Java libraries supplied in the Host Access Toolkit consist of the underlying class files packaged into JAR and analogous CAB files. JAR libraries are for Netscape and Java applets and applications and browsers employing the Java 2 Plug-in. CAB libraries are specific to Internet Explorer.

Deployed custom applications may follow one of two forms. The JAR/CAB libraries are installed on the machine where the Java application is run locally, or some or all of the required libraries may be downloaded over the network at runtime. In either case, these libraries should be digitally signed to denote their level of trustworthiness. The Host Access toolkit supplies both signed and unsigned libraries, distinguished as “debug” and “release” versions.

The “debug” JARs contain extra logging information when problem determination is of primary importance. These JARs are not digitally signed and ideally should not be used in a production environment. Because of the extra debugging code, their size is considerably greater than their “release” counterparts. That makes them less desirable in a network downloadable scenario. The “release” JARs are digitally signed and do not contain the extemporaneous logging and debugging code.

Any custom libraries used to create an applet should be digitally signed or else the security manager of the browser will deny certain instructions from executing. A valid certificate of the appropriate class must be obtained from a Certificate Authority (CA) such as Thawte Consulting or VeriSign, Inc. Signing the Host Access Toolkit JAR/CAB libraries with your own certificate is prohibited under the Host Access Toolkit license agreement.

Because of the size and organization of the HACLJ and HABJ libraries, the toolkit provides “componentized” JARs and CABs. To minimize the amount of code required to be downloaded/transferred, the Host Access Toolkit partitions the functionality of the APIs into multiple JARs and CABs. It is the responsibility of the developer to decide which JARs/CABs must be incorporated into the application's classpath to provide the required functionality. Refer to the installed documentation for a table of debug and release JARs and which functions are provided in each file:

HACL - toolkit directory\en\doc\hac\API_users_guide.html

BEANS- toolkit directory\en\doc\beans\API_users_guide.html

The custom applet or application that you write should be packaged into its own library file and digitally signed. Your code will use HACLJ and HABJ classes that reside in their own library files that have a trusted digital signature. Your digital signature certifies the trustworthiness of your custom classes, and the IBM digital signature certifies IBM's classes.

You should not subclass the HACLJ and HABJ classes, but should implement the needed abstract interfaces. And you should not create new classes within the com.ibm.* package domain without expressed written consent of IBM.

21.8.2 Swing components and Host Access Beans

The Host Access Toolkit in HACP 3.0 has introduced a new parallel library set. Developers may use either the “traditional” jdk1.1 set or the “new” java2 set of libraries.

The new java2 version of the Host Access Beans for Java are suited for use in Swing-based applications. There are Accessibility features available in this library that are not possible in the jdk1.1 version.

The jdk1.1 Host Access Beans for Java are based upon the Abstract Windowing Toolkit (AWT) components. IBM recommends using AWT components with the jdk 1.1 Host Access Beans. Sun warns that incompatibilities may exist when combining components from both the AWT and Swing libraries.

For more information about the caveats of mixing Swing and AWT components, go to the following site:

<http://java.sun.com/products/jfc/tsc/articles/mixing/index.html>

Our experience has shown that there are cases where embedding a jdk1.1 Terminal bean in a java.awt.JFrame experiences difficulties. If you must use this mixture, we recommend experimenting with the order of statements used to build the custom Terminal-containing display frame. We have noted that the “permissive” order may vary according to both platform and SDK version service release.

21.8.3 Subclassing and additional notes on licensing issues

Certain uses and practices are not supported or are prohibited by the IBM License Agreement.

Subclassing classes and beans

Subclassing is an object-oriented approach for extending or overriding the functionality defined by a particular class. The Java programming language allows the classes and beans provided in the Host Access Toolkit to be subclassed and public methods to be overridden (if not marked “final”). There are certain classes, such as abstract interfaces, that must be implemented with developer code. But in the main, subclassing and overriding of published HACLIJ classes and HABJ beans is not recommended and is not supported by IBM. Design your Java beans and classes to use IBM classes and beans without subclassing and overriding them.

Use of unofficial APIs

While allowable by the Java programming language, IBM will not support any use of any class not identified by the official API documentation. Many classes in IBM libraries are “public” in the Java sense, but have been intentionally excluded from the published Javadocs. Therefore any class that does not appear in the Toolkit Javadocs should not be called or programmed to directly.

Re-packaging of Host Access Toolkit JARs

While allowable by the Java programming language and available tools, the Host Access Toolkit JARs and CABs for both HACL for Java and the Host Access Beans for Java may not be re-packaged. IBM supplies a set of digitally signed JARs and CABs that internally assert certain needed permissions under a security manager. The Java code composing your classes, beans, packages, applets or applications may need to be digitally signed and to explicitly assert the proper permissions to work correctly under a security manager.

21.9 Using the Loadable Applet Interface

The Host On-Demand (HOD) server provides optional support for storing HOD-specific libraries in the cache of the client browser. Netscape Navigator and Microsoft Internet Explorer browsers are specifically supported. This mechanism operates for applications implementing the loadable applet interface, `com.ibm.eNetwork.HOD.common.cached.LoadableAppletInterface`.

You can develop custom applications of this type using the Host Access Class Library (HACL) and Host Access Beans for Java (HABJ). When served from the HOD server, the libraries for HACL and HABJ can be cached to the client browser. In this discussion, the term “application” indicates Java code that does work.

This facility will not cache or manage your non-HOD custom libraries (Java archive (JAR) and cabinet(CAB) files).

Your custom application class files and libraries should reside in the HOD published directory. Do not combine them into any of the HOD libraries, because that is not permitted by your license agreement. Reconstituted Host On-Demand libraries will be rejected by the caching mechanism.

The Essentials of a LoadableAppletInterface Application

An HTML page created with Deployment Wizard can launch an application implementing the `LoadableAppletInterface` interface. First, the browser loads and launches a HOD-proprietary class, `CachedAppletLoader`. That class in turn loads and launches your custom `LoadableAppletInterface` application.

This is accomplished dynamically by HOD JavaScript functions that write a new HTML page using `document.write(...)` statements. See a JavaScript manual for more information on how to write dynamically written HTML pages that are auto-launched in a browser environment.

The Deployment Wizard-based HTML page generates a final HTML page with the following line:

```
<APPLET
...

CODE="com.ibm.eNetwork.HOD.cached.appletloader.CachedAppletLoad
er"
...
>
```

That class takes a parameter named: `hod_AppName`, which is available through the HTML line:

```
var hod_AppName="";
```

It is customized by filling in the name of your custom application that implements the `LoadableAppletInterface` interface:

```
var hod_AppName='my_custom_app_name';
<APPLET
...
ARCHIVE=CachedAppletSupporter.jar,someCustomLib.jar
MAYSCRIPT NAME="\HODApplet\"
...
>
```

The custom supporting CAB library is specified as:

```
<APPLET ...>
```

```
...
<PARAM NAME=Cabinets
VALUE=CachedAppletSupporter.cab,someCustomLib.cab">;
...
</APPLET>
```

What the Browser Does and Doesn't Know

The browser Java virtual machine (JVM) loads and launches `CachedAppletLoader`. The browser is unaware that `CachedAppletLoader` has loaded and launched your custom class.

For this discussion, the `CachedAppletLoader` instance is referred to as `theRealApplet`. It is the “real applet” because the browser loaded and launched it. The browser is only aware of `theRealApplet`.

Your custom application is a child of `theRealApplet`. The relationship is supported by four interface methods:

```
public void setApplet(Applet theRealApplet);
public void init();
public void start();
public void stop();
```

Three of these methods, `init()`, `start()` and `stop()` have the same meaning as the analogous methods in `java.awt.Applet`.

21.9.1 `setApplet(Applet theRealApplet)` Is Called First

When `CachedAppletLoader` loads and launches your custom application, the first method called is `setApplet(Applet a)`. `CachedAppletLoader` passes in a reference to itself. The following is an example of custom application code:

Example 21-5 setApplet method

```
...
protected Applet theRealApplet = null;
...
public void setApplet(Applet a) {
```

```

    this.theRealApplet = a; //remember who my parent, an applet, is
}

```

The following is an example of code for calling the start() method of your application by using CachedAppletLoader:

Example 21-6 CachedAppletLoader

```

public void start() {
    ... //do some preparation and then...
    if(this.theRealApplet != null ) {
        displayThis(<a GUI component that does the real work>);
    }
}

// show the work component in the browser page to the user
public void displayThis(Component component) {
    this.theRealApplet.removeAll(); // just to be safe
    this.theRealApplet.add(component,null); // insert the new GUI component

    //ask my parent for the top level non-Window Component
    Component topComponent = getTopComponent(this.theRealApplet);
    //visualize me reliably in the browser page
    topComponent.setVisible(true);
    topComponent.validate();
    topComponent.repaint();
}

// climb the heirarchy as needed
public Component getTopComponent( Component containingThisComponent ) {
    Component topComponent = containingThisComponent;
    Component nextComponent = topComponent.getParent();
    while( (nextComponent != null) && !(nextComponent instanceof Window) ) {
        topComponent = nextComponent;
        nextComponent = topComponent.getParent();
    }
    return topComponent;
}

```

The RealApplet Is the Displayer

The repaint code above is needed to display your custom GUI component. Without it, your custom GUI component may not be visible in the browser page! Notice that there is no requirement that your LoadableAppletInterface application be the visible GUI component. For example, your custom application may be a server that marshals successive GUI components for display in the browser page, only to be removed and replaced by others. Remember to trigger the display of each GUI component after its insertion into “the real applet.”

Using the Deployment Wizard

The HOD Deployment Wizard is an integral part of this process. The terse instructions that follow assume that you have familiarized yourself with the Deployment Wizard. Please first read the documentation and practice using Deployment Wizard to configure customized client HTML pages.

The Deployment Wizard is used to create a custom HTML page in the HOD published folder. The following sections show how to modify that HTML file to specify your custom application that implements the LoadableAppletInterface:

21.9.2 Deploying in a Java 1 Environment

In a Java 1 environment, the custom application is built using JDK1.1 and is intended to run in a JDK 1.1 browser environment without a Java 2 Plug-in.

First, use the Deployment Wizard to create a custom HTML file. Specify “Client Java Type: Java 1”. Since you will be providing your own Terminal (or Session), you do not need to worry about selecting a configuration model. You can skip over most panels except for those that specify the Client Java Type, the Preload Options, and the final panel that allows you to specify the name of the output HTML file. Under Preload Options, uncheck all options that are not needed to support your custom application. This minimizes the total size of the downloaded and cached libraries. Finally, name this customization CustomJdk11Caching.html. Create it and exit Deployment Wizard.

This example assumes that the JDK1.1-based application that implements the LoadableAppletInterface interface is named:

lai0.JDK118App, where lai0 is the package name.

In CustomJdk11Caching.html look for this:

```
var hod_AppName="";
```

and change it to:

```
var hod_AppName='lai0.JDK118App.class';
```

or this:

```
var hod_AppName='lai0.JDK118App';
```

Either way is acceptable in the Java 1 environment, but the “.class” style is preferred.

Using loose classes in a Java 1 Environment

If you are planning to use loose classes (that is, no JAR or CAB files), copy the lai0 folder and its contents to \hostOn-Demand\HOD. No further modification of CustomJdk11Caching.html is needed.

Using a signed JAR or CAB library in a Java 1 Environment

Let's assume that your digitally signed custom library has the base name: myLib_J1. You need to further modify CustomJdk11Caching.html as follows:

To support Netscape Navigator, look for this line:

```
document.write("<APPLET ARCHIVE=CachedAppletSupporter.jar  
MAYSCRIPT NAME=\"HODApplet\"")
```

and change it to:

```
document.write("<APPLET  
ARCHIVE=CachedAppletSupporter.jar,myLib_J1.jar MAYSCRIPT  
NAME=\"HODApplet\"")
```

To support Internet Explorer, look for this line:

```
document.write("<PARAM NAME=Cabinets  
VALUE=CachedAppletSupporter.ca>");
```

and change it to:

```
document.write("<PARAM NAME=Cabinets  
VALUE=CachedAppletSupporter.cab,myLib_J1.cab>");
```

Running the custom application with caching in a Java 1 Environment

The first time a local browser accesses CustomJdk11Caching.html on the HOD server, the caching mechanism will detect that this is “the first time.” Therefore the caching operator will download all of the HOD-specific libraries enumerated in the Deployment Wizard step. Then a restart of the browser will launch the custom application. Thereafter any latency is due to the first download of the non-HOD libraries. These will be stored in the browser temporary cache and will be available without further downloading until the browser is closed.

21.9.3 Deploying in a Java 2 Environment

In a Java 2 environment, the custom application is built using SDK 1.3 (or higher) and is intended to run in a browser that does have a Java 2 Plug-in installed. In this environment, only JAR libraries are meaningful. Internet Explorer with a Java 2 Plug-in is equivalent in its operation to Netscape with a Java 2 Plug-in. The Java 2 Plug-in is the dominant, controlling factor.

First, use the Deployment Wizard to create a custom HTML file. Specify "Client Java Type: Java 2." Since you will be providing your own Terminal (or Session), you do not need to worry about selecting a configuration model. You can skip over most panels except for those that specify the Client Java Type, the Preload Options, and the final panel that allows you to specify the name of the output HTML file. Under Preload Options, uncheck all options that are not needed to support your custom application. This minimizes the total size of the downloaded and cached libraries. Finally, name this customization CustomJava2Caching.html. Create it and exit the Deployment Wizard.

This example assumes that you have written a Java 2-based application that implements the LoadableAppletInterface interface and that application is named: lai0.Java2App, where lai0 is the package name.

In CustomJava2Caching.html look for this line:

```
var hod_AppName ="";
```

and change it to:

```
var hod_AppName ='lai0.Java2App';
```

Note: In the Java 2 case the following does not work:

```
var hod_AppName ='lai0.Java2App.class';
```

Using loose classes in a Java 2 Environment

The Java 2 version of this mechanism operates only with loose classes. Therefore, you need to copy the lai0 folder and its contents to \hostondemand\HOD. No further modification is needed.

Running the custom applet with caching in a Java 2 Environment

The first time a local browser accesses CustomJava2Caching.html on the HOD server, the caching mechanism will detect that this is “the first time.” Therefore the caching operator will download all of the HOD-specific libraries enumerated in the Deployment Wizard step. When the download of cached libraries has completed, the custom application will start immediately (browser restart is purely optional). Thereafter any latency is due to the first download of the non-HOD libraries.

Compatibility of the Java 1 and Java 2 Versions

Remember that JDK 1.1-based custom classes run compatibly in a Java 2 environment. Likewise, JDK 1.1-based HOD classes will run compatibly in a Java 2 environment. However, the reverse is not true: Java 2-based HOD classes will fail when run in a JDK 1.1 browser

21.10 Host On-Demand J2EE Connectors

The J2EE Connector architecture provides a standard set of services allowing developers to quickly connect and integrate their applications with virtually any back-end Enterprise Information Systems, and to any application servers conforming to the J2EE Connector architecture. This is part of the Java 2 Enterprise Edition.

These services are supplied as Plug-in connectors (sometimes called resource adapters). Before the J2EE Connector architecture was introduced, there was no standard architecture for integrating heterogeneous Enterprise Information Systems (EIS). Host On-Demand users had to use Host Access Class Library (HACL) to access hosts; other vendors provide specific architectures for this purpose.

Figure 21-3 is a diagrammatic representation of the Host On-Demand J2EE Connector architecture.

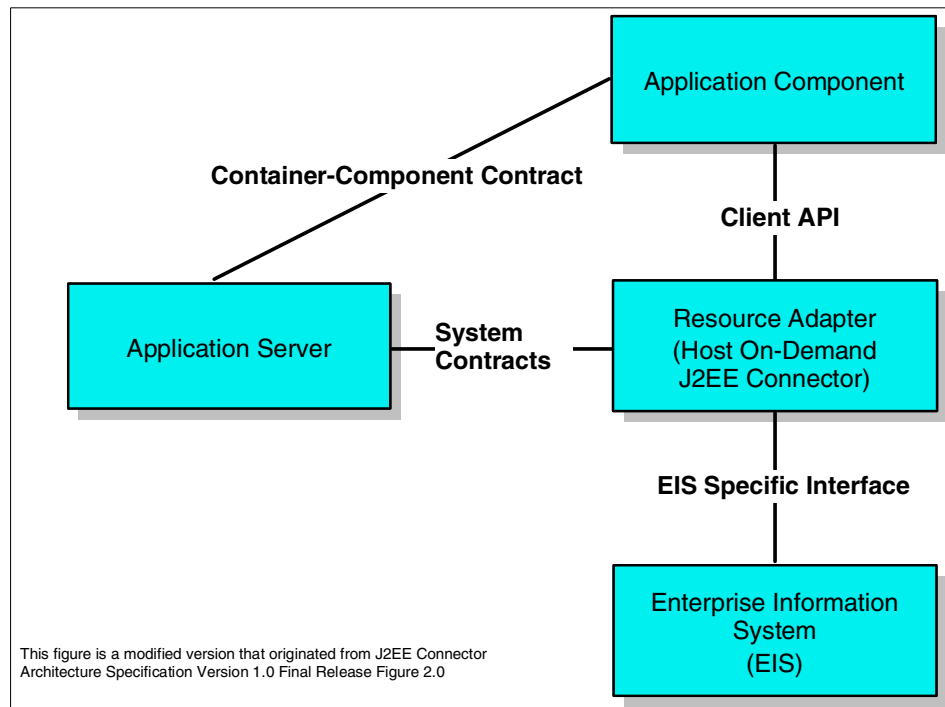


Figure 21-3 Host On-Demand J2EE Connector architecture

21.10.1 Why use Host On-Demand J2EE Connectors?

The Host On-Demand J2EE Connector provides access to 3270, 5250, Customer Information and Control System (CICS), and Virtual Terminal (VT) hosts from the Internet.

The Host On-Demand J2EE Connector is a Java programming interface which conforms to the J2EE Connector Specification Version 1.0 - Proposed Final Draft 2 from Sun Microsystems. This translates to a standard set of services for accessing any system that is J2EE Connector architecture compliant, whether it be the mainframe-based host systems or any other system.

The J2EE Connector is integrated into Websphere Studio Application Developer-Integraton Edition V5.0 and later (WSAD-IE) which provides the tools to capture, generate screens and to develop HOD J2EE Connector applications

An applet or servlet can be written to use the Host On-Demand J2EE Connector classes for host access over TCP/IP using standard Telnet protocols: TN3270, TN5250, CICS or VT emulation, which otherwise would require the use of a HACL-like API or other emulator APIs.

Since J2EE Connector is a part of the J2EE standards the following standard system level services between the application server and the back end host system are provided by the J2EE compliant application server.

- ▶ Connection Management
- ▶ Transaction Management
- ▶ Security

Along the lines of JDBC, which is the de facto standard to integrate object-oriented Java technology applications with relational databases, the J2EE Connector architecture has become the preferred method to integrate component-based Java technology programs with non-relational back-end enterprise applications.

21.10.2 Host On-Demand J2EE Connectors development cycle

Host On-Demand J2EE Connector provides a set of Resource adapters that communicate to 3270, 5250, CICS, and VT hosts. These resource adapters (.RAR files) are deployed to a conforming application server, such as IBM WebSphere Application Server, ideally Version 4 and above.

The users can write applets or servlets using the application programming interfaces (APIs) provided in Host On-Demand J2EE Connector via Websphere Studio Application Developer-Integration Edition V5.0 and later (WSAD-IE). WSAD-IE tools are used to write and test user applets or applications. WSAD-IE is recommended for rapid application development.

These applets and applications are then deployed on to the application server such as WebSphere Application Server. Having done this, we are internet/intranet ready. Of course, there are security and other parameters that need to be addressed before it really can be used.

21.10.3 Host On-Demand J2EE Connectors classes

The HOD J2EE Connector consists of the following classes:

- ▶ J2HODConnection
- ▶ J2HODConnectionFactory
- ▶ J2HODConnectionRequestInfo
- ▶ J2HODConnectionSpec
- ▶ J2HODInteraction

- ▶ J2HODInteractionSpec
- ▶ J2HODManagedConnection
- ▶ J2HODBaseManagedConnectionFactory
- ▶ J2HOD3270ManagedConnectionFactory
- ▶ J2HOD5250ManagedConnectionFactory
- ▶ J2HODCICSManagedConnectionFactory
- ▶ J2HODVTManagedConnectionFactory
- ▶ J2HODScreenRecord
- ▶ J2HODScreenableRecord

Please refer to the online documentation provided along with Host Access Toolkit for detailed description of these classes. It can be found at:

/Program Files/IBM/Host Access Toolkit/en/doc/connector2/

21.10.4 A sample program

In this section, let's explore a sample program to understand Host On-Demand J2EE Connector and its significance.

The Host Access Toolkit comes with sample programs to understand the HOD J2EE Connector architecture. Well documented and highly self-explanatory programs are provided for TN3270 host access and TN5250 based host access.

The sample programs are found in \Program Files\IBM\Host Access Toolkit\toolkit\connector2\samples.

Detailed step-by-step procedure for running these sample programs are available as part of the documentation at \Program Files\IBM\Host Access Toolkit\en\doc\connector2\J2EEFirstApplet.html.

Documentation as JavaDoc is available for this API as part of the toolkit it can be found at \Program Files\IBM\Host Access Toolkit\en\doc\connector2\index.html.

21.10.5 Documentation and more information

Detailed documentation is available as part of the Host Access Toolkit package. It is installed on the user workstation when the toolkit is installed. By default it will be installed in C:\Program Files\IBM\Host Access Toolkit\en\doc.

21.11 Additional help

A number of Web resources are available for developers using these APIs:

- ▶ The IBM Host On-Demand Web page
<http://www.ibm.com/software/webservers/hostondemand/>
- ▶ The IBM Personal Communications Web page:
<http://www.ibm.com/enetwork/pcomm/>
- ▶ The Sun Microsystems home page for accessing and downloading various Java APIs and tools:
<http://java.sun.com/products/>
- ▶ The IBM public “Java technology zone”:
<http://www.ibm.com/developerworks/java/>
- ▶ The IBM public “IBM developer kit porting” Web page:
<http://www.ibm.com/developerworks/java/jdk/index.html>

The following news groups require access to a Usenet newsgroup server and a client news reader:

- ▶ `ibm.software.hostondemand`
- ▶ `ibm.software.pcomm`



Problem determination

IBM provides the following online documentation to assist you in using Host On-Demand:

The Host On-Demand 7 InfoCenter contains problem determination information under the topic of Troubleshooting. To access the InfoCenter on Windows operating systems, select Start > Programs > IBM Host On-Demand > InfoCenter. You can also access the InfoCenter from your own Host On-Demand Server at this URL:

http://HOD_Server Address/hod/en/help/2tabcontents.html

or on the internet from the IBM Host On-Demand Library page

<http://www.ibm.com/software/webservers/hostondemand/support.html>

In the InfoCenter shipped with the product, you can find Troubleshooting under the “Online Help”. This topic contains the following information to help troubleshoot problems you are having with the product:

- ▶ Pointers to help find the latest service updates and product documentation
- ▶ Troubleshooting checklists
 - Information to help direct you to solutions for problems with your Host On-Demand installation.
- ▶ Help for messages

Find an explanation and possible solution or explanation for any error messages that might occur while executing Host On-Demand.

- ▶ Help for the symbols and codes at the bottom of the session window (OIA)

The OIA (Operator Information Area) is the area at the bottom of the screen where session indicators and messages appear. Listed below are the session information fields, with an explanation for each.

- ▶ Troubleshooting Topics

A collection of online articles to direct you to help for categories of problems such as printing or Redirector issues.

- ▶ Checklist for gathering information before calling IBM Service

A short list of useful information to gather before calling IBM support. Obtaining this information before you call IBM allows the IBM support technician to have all the problem details needed to quickly provide a solution to your problem.

- ▶ Procedures to follow in gathering trace and debug information.

Information and lists of procedures to help you obtain problem traces and logs for your Host On-Demand problem.

Web Resources

The InfoCenter available on the IBM Host On-Demand Library page <http://www.ibm.com/software/web servers/hostondemand/support.html> provides the following additional information:

1. Access to all Host On-Demand documentation
2. Host Access Class Library Reference
3. Host Access Beans for Java Reference
4. Program Directory
5. Host Integration Redbook
6. Tutorials
7. White Papers
8. Troubleshooting Guide
9. Link to the Host On-Demand Hints and Tips
10. Link to Service update/Product update information



Part 2

Personal Communications Version 5.6

IBM Personal Communications Version 5.6 is the component of Host Access Client Package V3 that provides a full-function terminal emulator. We will limit our discussion to Personal Communications Version 5.6 for Windows to the enhancements as made since the base version 5.0



Enhancements

Personal Communications Version 5.6 is based on the popular Personal Communications Version 5.0. This chapter describes the new functions that have been added to Personal Communications Version 5.6.

23.1 New enhancements of Personal Communications 5.6

The following sections describes the major new functions in Personal Communications Version 5.6.

23.1.1 Windows XP compatibility

Personal Communications Version 5.6 is compatible with Microsoft Windows XP. It has received the certification *Compatible with Microsoft Windows XP*. This certification confirms that Personal Communications Version 5.6 executes on Microsoft Windows XP and will not interfere with the operating system or application stability

23.1.2 Accessibility enhancements

Personal Communications Version 5.6 provides functionality with accessibility devices such as screen readers. The **Sounds** option in the **Edit - Preferences - Appearance - Display Setup** menu has in **Category Sound** a **Mute** function which allows Personal Communications program sounds to be muted. The Windows **Control Panel - Show Sounds** option can be used to display status bar messages when a program alarm, beep, or sound is played, even if Personal Communications sound has been muted. You can check the **Use SoundSentry** box in **Control Panel - Accessibility Options** so that the window which caused a sound will flash.

Several configuration panels of Personal Communications Version 5.6 have been redesigned for accessibility compliance and screen reader compatibility like e.g. the popup keypad configuration

In general Personal Communications Version 5.6 has been enhanced for accessibility to support functions as:

- ▶ support interfaces used by screen readers and magnifiers
- ▶ can be operated using only the keyboard
- ▶ support customizing of display attributes such as color, contrast and font size
- ▶ communicate all information independently of color
- ▶ allow the user to take as much time as needed to respond
- ▶ do not rely on audio
- ▶ do not flash the screen at rates that could induce epileptic seizures
- ▶ Screen Reader Assist

Users can configure a toggle key to enable Personal Communications Version 5.6 to replace blank and null characters in the input field with another character. This option allows screen readers to report the length of the field to visually impaired users. Data sent to and from the host is not changed - only the screen display and the screen reader's voicing of the display are affected. By default, this function is not enabled.

For 3270 and VT emulators, the default padding character is a blank. For 5250, the default replacement character is an underscore. You can choose another character if you prefer.

During the emulation session, you can turn the screen reader assist on or off, as needed. To map the screen reader toggle to a key, select **Edit - Preferences - Keyboard**. Click **Customize** to access the keyboard setup dialog. In the pane **Select a Key-Action** you can choose in the **Function** drop down box the function **Screen Reader Toggle** and assign it to any keyboard key.

23.1.3 Detect and repair feature

Authorized users can initiate the *Detect and Repair* capabilities of Windows Installer from either a session window or the Session Manager and as well from the Windows Add/Remove Software utility.



Figure 23-1 Program Maintenance window

With the Program Maintenance (shown in Figure 23-1 on page 807) as started from the Windows Add/Remove Software utility users can initiate a manual repair, a remove and can as well add or remove features to the product.

The detect and repair operation will perform a check on the Personal Communications files to determine if the installation has been damaged and subsequently perform a repair if necessary. Providing this capability within the application allows users a fast and easy way to repair a damaged installation without requiring the assistance from the help desk. To allow for the invocation of the detect and repair operation within the application a new menu item was added to the Help pull-down menu as shown in Figure 23-2 on page 808.

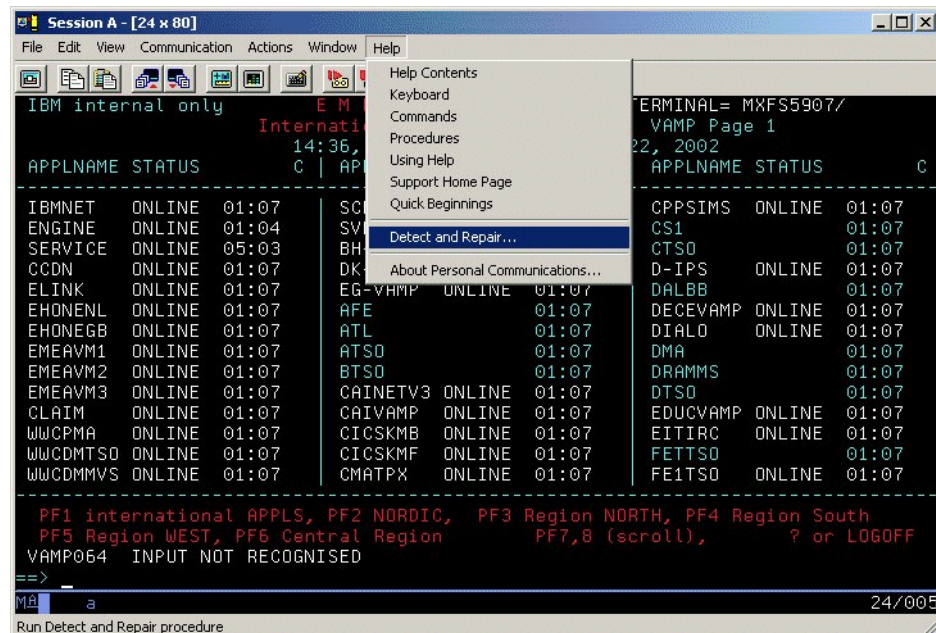


Figure 23-2 Invoking Detect and Repair

To enable Personal Communications Version 5.6 for that function, all executables have been enhanced with file versioning according to the need of Windows Installer. You can see the contents using Windows Explorer on files of Personal Communications Version 5.6: Click in right mouse menu the Properties button and view the Version tab contents of the file properties as shown in Figure 23-3 on page 809

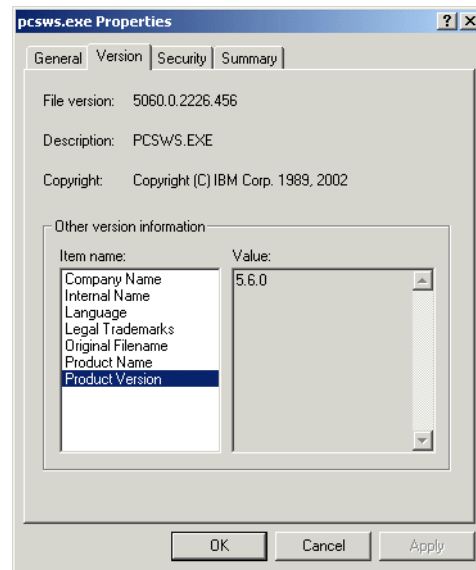


Figure 23-3 Showing version information of PCSWS.EXE with Windows Explorer

This also allows better control during installation so that the restriction to close all running applications during installation or repair could be removed for Personal Communications Version 5.6.

Also it was needed that the components and features had to be restructured for a more granular installation. This again had benefits to the feature selection tree for installation as shown in “Feature selection” on page 809

This new detect and repair will even allow to detect wrong versions executables and non-executables during startup of Personal Communications Version 5.6. If an error is detected the user might get prompted for the medium from which the personal communication was installed in order to copy the valid file.

23.1.4 Feature selection

In order to have more granularity for installing features and as well to support the *Detect and Repair* feature the selection menu of the panel for Custom Setup was enhanced. Figure 23-4 on page 810 shows as an example the enhanced SNA selection options

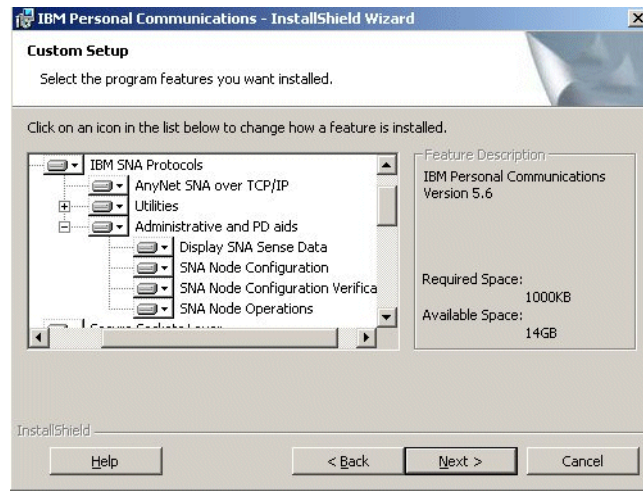


Figure 23-4 Enhanced custom setup

Sub-features have been added to the feature tree as follows. New sub-features are marked *italic*:

PCOMM

- ▶ Emulators
 - 3270 emulation and services
 - *ZipPrint*
 - 5250 emulation and services
 - Data Transfer
 - AS400 Connection Configuration
 - vt emulation
 - fonts
 - font-1
 - font-2
 - font-x
- ▶ IBM SNA protocols
 - AnyNet SNA over TCP/IP
 - *Utilities*
 - *Check Connection APING*
 - *Database Access*

- *Transfer File AFTP*
- *Administrative and PD aids*
 - *Display SNA Sense Data*
 - *SNA Node Configuration*
 - *SNA Node Configuration Verification*
 - *SNA Node Operations*
- ▶ *Utilities*
 - *CM Mouse*
 - *Convert Macro*
 - *DOS EHLAPPI*
 - *Menu Bar Customizing*
 - *Multiple Sessions*
 - *User Preference Manager*
- ▶ *Administrative and PD aids*
 - *Info Bundler*
 - *.Internet Service*
 - *Log Viewer*
 - *Migration Utility (De-selecting will only eliminate the shortcut.)*
 - *Product Update Tool*
- ▶ *API's*
 - *HLLAPI*
 - *PCS API*
 - *Beans*
 - *Samples*
 - *HACL*
 - *VB*
 - *MISC*

23.1.5 Installing Personal Communications using an ini file and using system variables and UNC paths

Personal Communications Version 5.6 can be installed silently using system variables and UNC paths (Universal Naming Convention = \\machine_name\resource).

First we want to show the general usage of response (ini) files without variables and UNC paths:

General usage of ini files

We will do an installation of pcomm, select the features which we want to install and record the install process in a response file using the SAVEINI feature. But we do not really want to install Personal Communications on that machine yet - we only want to get the response file recorded. So to prevent the installation itself we add the ONLYINI=1 parameter while using the SAVEINI feature. Once we have the response file we will use it to finally install Personal Communications using the USEINI parameter. After this has been used successfully at one machine the installation with that response file and the USEINI feature can be done on any similar machine.

According to online documentation *Personal Communications for Windows, Version 5.6: CD-ROM Guide to Installation*, Chapter 4 we record an installation (ini) file for an installation Personal Communications Version 5.6 using the following command at the root command prompt:

```
"c:\pcomm from network\pcomm\install\pcomm\setup.exe" /v"/L*v  
\"%temp%\pcsinst.log\" SAVEINI=\"%temp%\pcomm.ini\" ONLYINI=1"
```

Please use the blanks in the command as shown in the example - some are delimiter for the parameters

"c:\pcomm from network\pcomm\" is the location of the image of the CD (in your case it might be simply the drive letter of the cd). If the call to the setup.exe contains blanks it needs to be set into double quotes as in the example. Be sure to use the **setp.exe** from the subdirectory **\install\pcomm**.

/v is a command line parameter for MSI (Microsoft Installer) to get the following parameters passed on to it

/L*v \"%temp%\pcsinst.log\" defines that a verbose log file should be saved named pcsinst.log and should be placed into the temporary subdirectory of that user (you can see what subdirectory is currently defined by using the command **set temp**)

SAVEINI=\"%temp%\pcomm.ini\" defines the name and subdirectory for a response file (ini file) which we record for later use.

Note: SAVEINI does not create new subdirectories. The defined subdirectory has to exist prior to execution of command. Otherwise no ini file is written! (no error message appears on screen or in log!)

"ONLYINI=1" defines that Personal Communications is not installed during this process - we are only recording the response file and want to install Personal Communications later, using that response file

After you have issued this command you will see the following sequence of screens which you have to fill in according to your desired configuration:

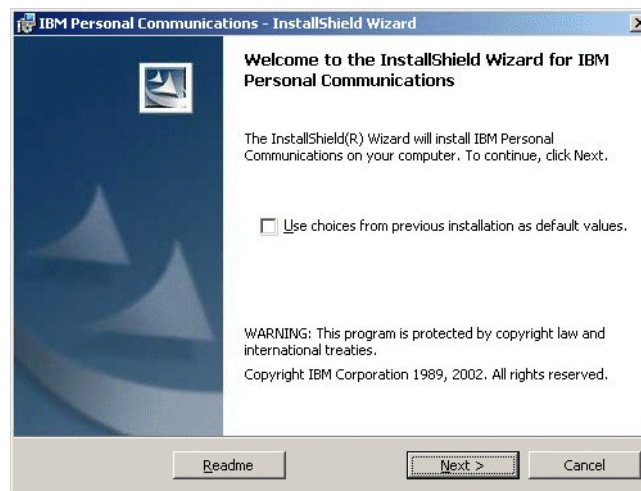


Figure 23-5 Start of the installation of pcomm



Figure 23-6 Software licences agreement

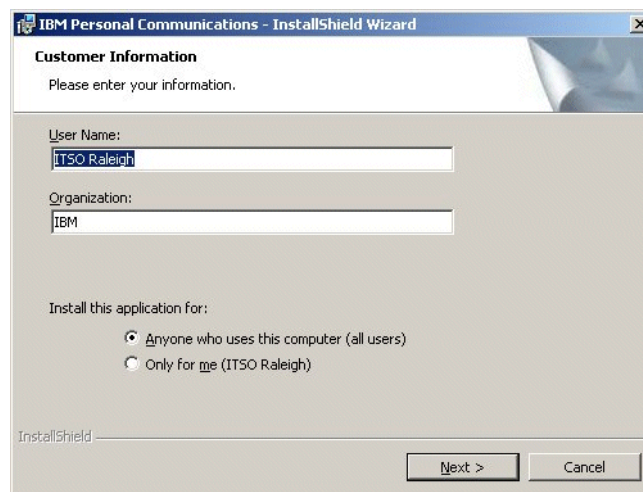


Figure 23-7 User information and selection for which user Personal Communications is installed



Figure 23-8 Setup type panel

If you select in panel Figure 23-8 on page 815 the typical installation you will get the Telnet related features but not the SNA related features (like Node functions)



Figure 23-9 Language selection panel

Between appearing of languages selection panel and custom setup panel the disk space is checked.

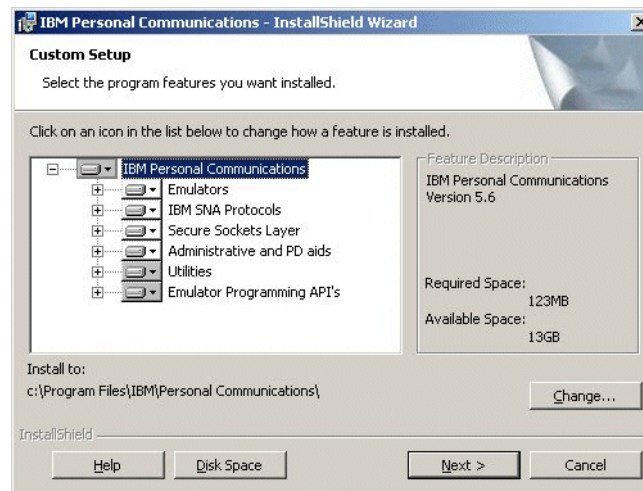


Figure 23-10 Custom setup panel for selecting installable features

For details on the new enhanced custom setup panel see Chapter 23.1.4, “Feature selection” on page 809



Figure 23-11 Data location selection.

We have selected the classic private subdirectory to be consistent with the structure as Personal Communications used it throughout the past. So users will find their macros, keyboard layouts, session configuration files etc. in the old known places. For details on the data location please refer to Chapter 24.1, “Migration during installation” on page 868 and online documentation *Personal Communications for Windows, Version 5.6: CD-ROM Guide to Installation*, Chapter 3.

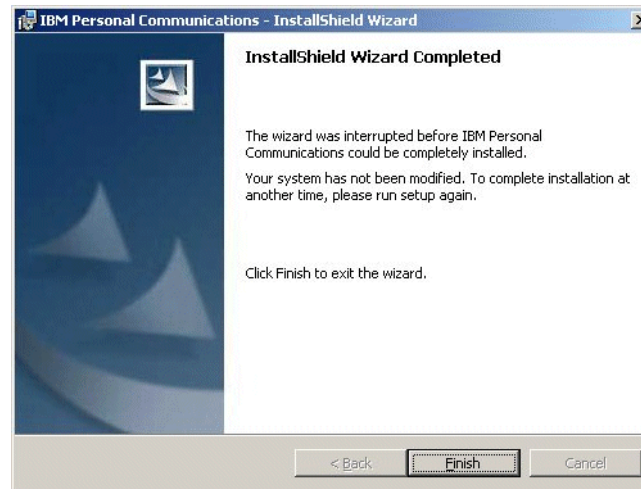


Figure 23-12 Completed recording of ini file without installation

The panel in Figure 23-12 on page 817 does show that we obviously had used parameter ONLYINI=1 and Personal Communications was not installed in this case.

You can verify that in the log file which we had created during this process. At the end you will find that the ONLYINI=1 caused the termination of the installation:

Example 23-1 Log file after recording a response file without installing HOD

```

1: 8/27/2002 11:51:36 IniUtils/InterruptSetup: ONLYINI=1; Installation
termination requested.
1: 8/27/2002 11:51:36 IniUtils/InterruptSetup: Note: Log file will
indicate installation failed.
Action ended 11:51:36: InterruptSetup.D49A5578_D28E_4C88_9DAE_3249953B70DA.
Return value 3.
MSI (c) (B8:70): Doing action: SetupCompleteError
Action start 11:51:36: SetupCompleteError.
Action 11:51:36: SetupCompleteError. Dialog created

```

Because Personal Communications was not installed you are not prompted to re-boot the system

Here is the resulting ini file which was recorded:

Example 23-2 Recorded response file

```
[Notes]
Note1=This is an installation initialization file for the product named
below.
Note2=Feature values: 1 = on demand, 2 = do not install, 3 = install, 4 =
run from source.
[Product]
Manufacturer=IBM
ProductName=IBM Personal Communications
ProductVersion=5.6.0000
ProductCode={C26FC7AE-2A5E-11D6-982D-006094EB6655}
[Properties]
INSTALLDIR=c:\Program Files\IBM\Personal Communications\
ALLUSERS=1
LAPAgree=Yes
LAPEnAgree=No
APPLICATIONUSERS=AllUsers
APPDATALOCATION=PcsPrivateDir
COMPANYNAME=IBM
USERNAME=ITSO Raleigh
_IsSetupTypeMin=Custom
RUNSOURCE=False
AUTO_MIGRATE_PROFILES=True
AUTO_MIGRATE_LEVEL=MigrateLevel3
AUTO_MIGRATE_LEVEL1=MigrateLevel2
LANG_CSY=0
LANG_DAN=0
LANG_DEU=1
LANG_ESP=0
LANG_FIN=0
LANG_FRA=0
LANG_HUN=0
LANG_ITA=0
LANG_JPN=0
LANG_NLD=0
LANG_NOR=0
LANG_PLK=0
LANG_PTG=0
LANG_PTB=0
LANG_RUS=0
LANG_SLV=0
LANG_SVE=0
LANG_TUR=0
[Features]
```

Pcomm=3
MigrateLevel3=3
MigrateLevel2=2
MigrateLevel0=3
MigrateLevel1=2
RunLocalRegEntries=3
RunSourceRegEntries=2
IBM_SNA_Protocols=3
Emulators=3
Emul_JRE=2
Fonts=3
Font_fon_ibm_3270_heb_nt=3
Font_fon_ibm_3270_heb_9x=2
Common_Uilities=3
MultipleSessions=3
CommonAdminPdAids=3
LogViewer=3
Emul_5250=3
Data_Transfer=3
Admin_PD_Aids_SNA=3
SnaNodeConfig=3
ConnectionConfig=3
Emul_3270=3
ZipPrint=3
ITF=3
SSL=3
Emul_APIs=3
AparTool=3
DOS_EHLLAPI=3
Emul_VT=3
Certificate_Management=3
Certificate_Wizard=3
AnyNet_SNA=3
Convert_Macro=3
CM_Mouse=2
MenuBar_Customization=3
Samples=2
Host_Access_Class_Lib=2
VisualBasic=2
Misc_APIs=2
Root_Combo=4
Utilities_SNA=3
CheckConnectionAPING=3
DatabaseAccess=3
TransferFileAFTP=3
DisplaySenseData=3
SnaNodeConfigVerify=3
SnaNodeOps=3
UPM=3

```

InfoBundler=3
InternetService=3
MigUtility=3
Font_ttf_jpnpbarc=3
Font_fon_ibm_vt=3
Font_fon_pcslaos=3
Font_fon_ibm_3270_1250=3
Font_fon_ibm_3270_1251=3
Font_fon_ibm_3270_1253=3
Font_fon_ibm_3270_1254=3
Font_fon_ibm_3270_1257=3
Font_fon_ibm_3270_1258=3
Font_fon_ibm_3270_arb=3
Font_fon_pcsthai=3
Font_fon_pcsviet=3
Font_fon_ibm_3270_avt=3
Font_ttf_cumrheb=3
Font_ttf_khamla=3
Font_ttf_ThaiPhuket=3
Font_ttf_typing_arabic=3
Font_ttf_devanagari=3
Font_ttf_typing_arabic_vt=3
Font_ttf_vt_graphics=3
AppDataUserPm=2
AppDataPrivate=3
AppDataCommon=2
AppDataUser=2

```

Now we use the recorded ini file for installation of Personal Communications Version 5.6 on the same machine with the following command in which we included the parameters for suppressing reboot and to run quiet:

```

"c:\pcomm from network\pcomm\install\pcomm\setup.exe" /v"REBOOT=ReallySuppress
/L*v \"%temp%\pcsinst2.log\" USEINI=\"%temp%\pcomm.ini\" /qn"

```

You will see no progress indicator and no messages once the process is running. Besides a flickering disk or CD LED the only indication for this process to be running is the appearance of MSIEXEC.EXE in the task manager.

At the end of the process you will find at the end of the log file:

```

MSI (s) (A0:5C): Product: IBM Personal Communications -- Installation operation
completed successfully.

```

Using system variables and UNC paths

Administrators might want to go one step further when installing Personal Communications using different response files depending on machine types or user groups. This could be accomplished by using system variables or UNC paths (Universal Naming Convention)

To demonstrate the usage of a system variable we have set up the variable `pcomm_ini` to point to the subdirectory `c:\pcomm_temp`. We used for that the GUI of My Computer: Click right mouse at the My Computer icon on the desktop and select Properties.

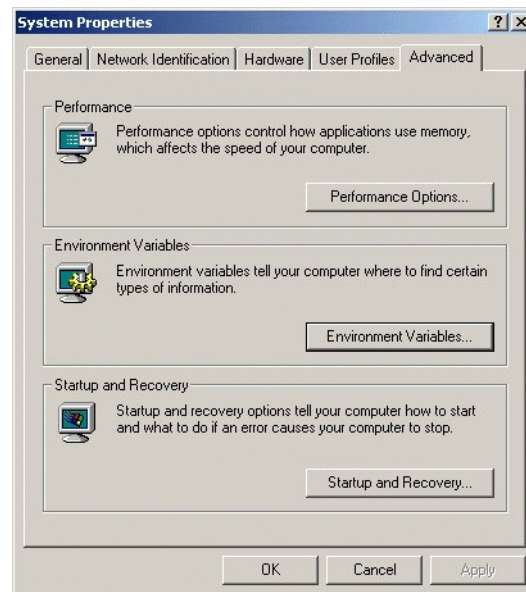


Figure 23-13 System properties

In system properties select the advanced tab and click the Environment Variables button.

In the pane System Variables click New and fill in your desired system variables

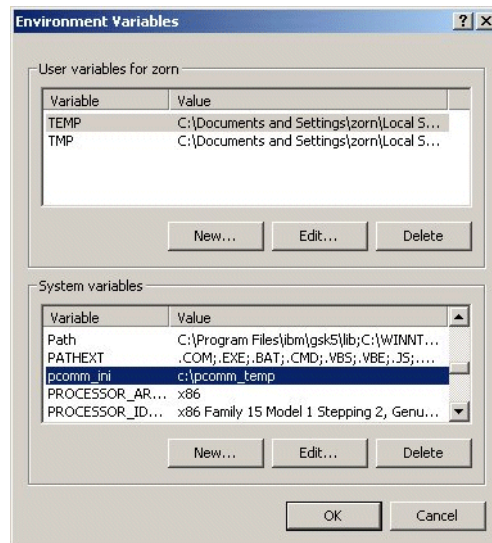


Figure 23-14 User and system variables

Be sure that the subdirectory as entered in the system variable exists and is accessible for the machine and user. After this has been saved execute the following command:

```
"c:\pcomm from network\pcomm\install\pcomm\setup.exe" /v"/L*v
\ "%temp%\pcsinst.log\" SAVEINI=%pcomm_ini%\myCustom.ini ONLYINI=1
transforms="c:\pcomm from network\pcomm\1033.MST\""
```

The **transforms="c:\pcomm from network\pcomm\1033.MST\"** will bring up the GUI panels in english as seen in "General usage of ini files" on page 812. For using other languages please refer to table *NLS Abbreviations and Language Codes* in Appendix C of *Personal Communications for Windows, Version 5.6: CD-ROM Guide to Installation*. Please add the full path for the MST file as shown in example.

Now we use instead of an system variable a UNC path - in our case we used a TCP/IP address of a remote machine in dotted notation:

```
"c:\pcomm from network\pcomm\install\pcomm\setup.exe" /v"/L*v
\ "%temp%\pcsinst.log\" SAVEINI=\\9.39.1.33\e-disk\myCustom.ini ONLYINI=1
transforms="c:\pcomm from network\pcomm\1033.MST\""
```

(this IP address was as well mapped to a local drive letter).

With the following command we tested successfully the UNC for another machine (m23vnx72) in the same domain:

```
"c:\pcomm from network\pcomm\install\pcomm\setup.exe" /v"/L*v  
\"%temp%\pcsinst.log\" SAVEINI=\\m23vnx72\shared\myCustom.ini ONLYINI=1  
transforms=\"c:\pcomm from network\pcomm\1033.MST\""
```

23.1.6 Improved security features

Personal Communications Version 5.6 provides MSCAPI support for Certificate Management in Windows 2000 and XP, for use with Smart Card and Microsoft Certificate Store. An enhanced command-line interface (IKEYMAN) for certificate management is also provided, which can perform the tasks of the Certificate Management utility, without the graphical interface. Refer to the Administrator's Guide and Reference for more information about security.

23.1.7 Tivoli support

Personal Communications maintains Tivoli-Ready certification for Version 5.6.

23.1.8 Sound functionality enhancements

You can configure specific program sounds through the Windows Control Panel, using sound files included with the Personal Communications product. In addition, you can use the Mute option in the Display Setup dialog to suppress program sounds.

23.1.9 Enhanced run java applet facility

The Run Java Applet Facility now provides the ability to make use of the JAVA_HOME environment variable, which allows use of an alternate Java Runtime Environment.

This function is available in the Actions pull down menu of the Personal Communications sessions as shown in Figure 23-15 on page 824. See the Help page of the Run Java Applet panel for further information

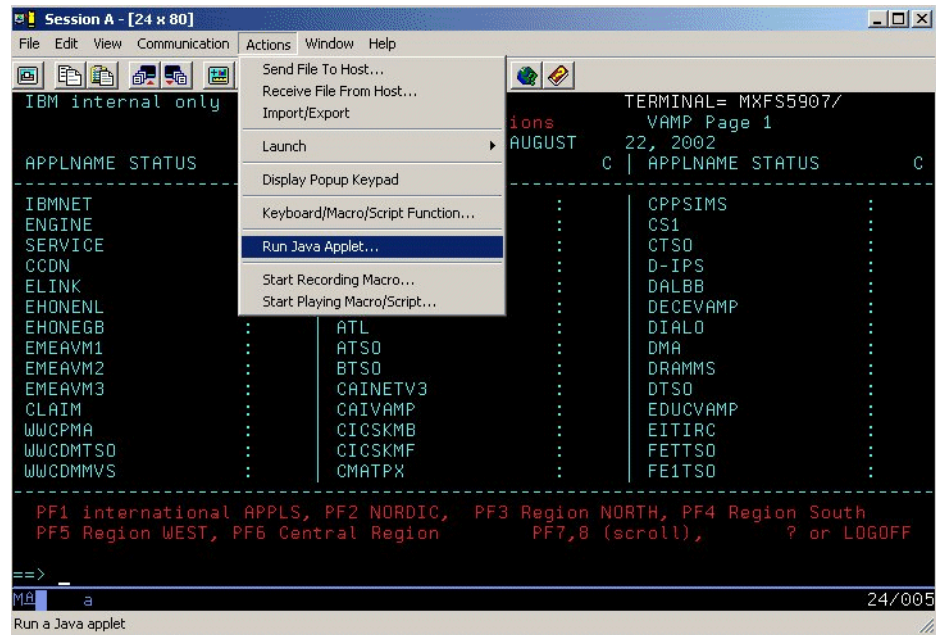


Figure 23-15 Run Java Applet

23.1.10 Bidirectional RTL print orientation

For bidirectional 3270 sessions, you can choose to print a file in RTL (right to left) orientation, from the Printer Setup dialog in the display session menu.

23.1.11 Enhanced macro conversion utility

The Convert Macro utility converts recorded native Personal Communications macros into XML or VBScript. The macro must exist in the application data directory specified during installation of Personal Communications! The macro conversion utility has been enhanced to allow saving the converted file into a location that you specify. This has been accomplished by removing the Output File Name box in the Convert Macro panel and use instead of that the Save As dialog of the operating system.

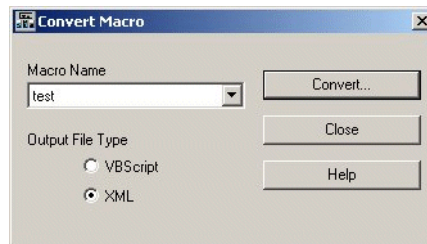


Figure 23-16 Convert Macro panel

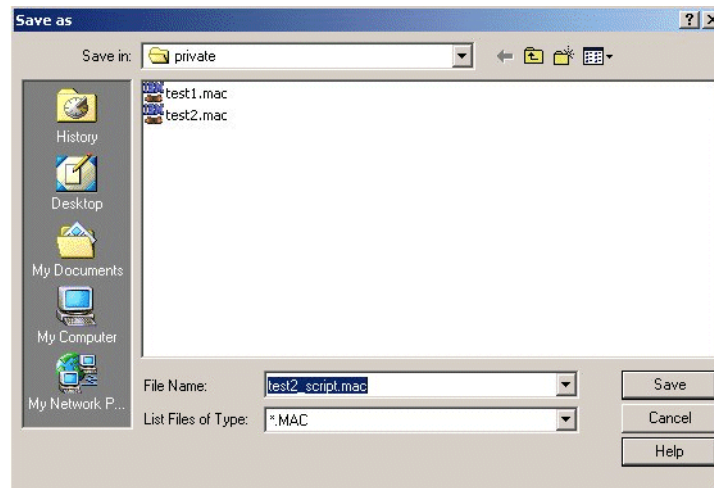


Figure 23-17 Saving macro using dialog of operating system

Note: Macros that are converted to XML are intended for use in Host On-Demand and will not function in Personal Communications emulation sessions. These converted macros will not appear in the list of available Personal Communications macros.

Hint: If macros used for execution by Personal Communications are not stored in your own Personal Communications subdirectory (e.g. supplied on a server for common use), you can switch Personal Communications to use another subdirectory to look for macros by the following change to your pcsws.ini:

```
[Macro]  
DIR=(drive):\\(directory-name)\\...
```

The Convert Macro utility will not be impacted by this

23.1.12 Arabic code page conversion for macro

Arabic code page conversion is performed when an XML/VBScript macro is created, so that Arabic characters in the macro are displayed properly. The appropriate reverse conversion is done when the macro is played back.

23.1.13 Map printer setup to key and add button to menu

For 3270, 5250, and VT emulators, you can map a key sequence to bring up the Printer Setup dialog. There is no default key combination for this function. To map the Printer Setup dialog to a key, select **Edit - Preferences - Keyboard**. Click **Customize** to access the keyboard setup dialog. In the pane **Select a Key-Action** you can chose in the **Function** drop down box the function **Printer Setup** and assign it to any keyboard key.

In addition, the PageSetup can be added as icon to the toolbar as follows:

In an empty area at the toolbar click with the right mouse. The context menu appears. Click at Create. The Create Toolbar Item menu is displayed. In the File tap as shown in Figure 23-18 on page 827 you can click at the PageSetup Icon. When it appears as depressed you click OK and it will be added to the tool bar.

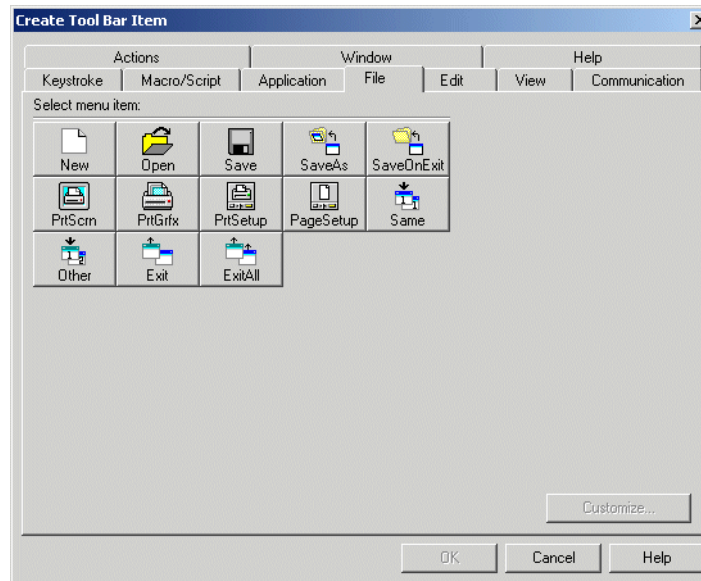


Figure 23-18 Add PageSetup icon to toolbar

23.1.14 ASCII text pdf

The new **basic_ascii.PDF** in the **pdtpdf** subdirectory of Personal Communications Version 5.6 does not contain printer commands—this allows ASCII text to be sent to a printer. An accompanying PDT file is also supplied.

23.1.15 Enhanced SNA link configuration dialogs

The SNA link configuration panels in both the configuration wizard and the SNA Node Configuration facility have been enhanced to display the status of the adapters at configuration time. Installed adapters are listed and shown as enabled or disabled.

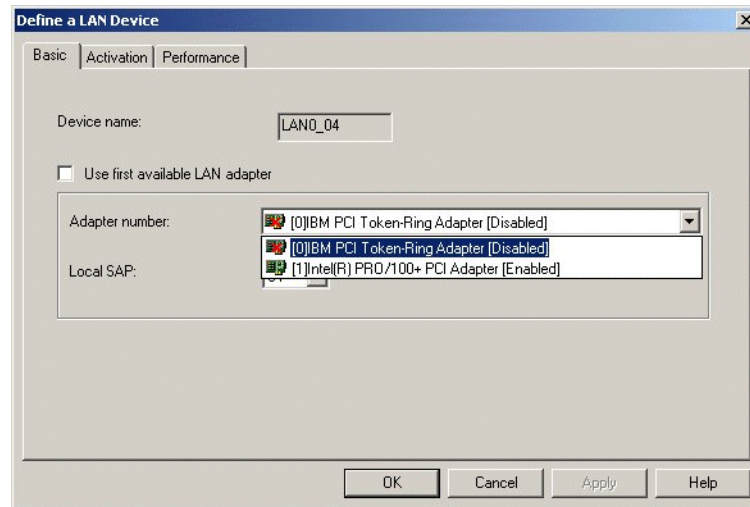


Figure 23-19 Showing available adapters in Node Configuration's Define a Lan Device (manual selection)

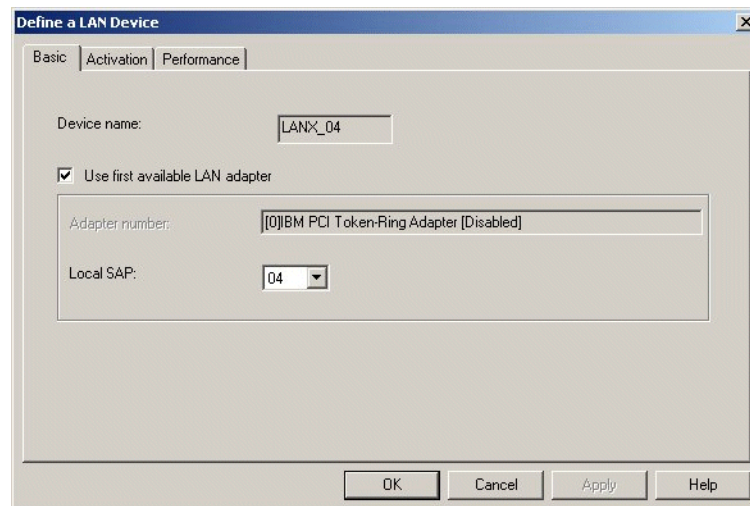


Figure 23-20 Showing available adapters in Node Configuration's Lan Device (automatic selection via LAN_X)

Using automatic adapter selection (LANX)

When choosing the automatic adapter selection (LANX), the first network card which is listed by the operating system and to which the LLC2 protocol is bound is picked by Personal Communications as LANX_04. But in this example that adapter is currently not enabled (no network cable plugged in). However it is not relevant for LANX which adapter is picked during configuration time - it will always show the first adapter at that time. Don't worry: LANX will choose its adapter again dynamically at run time. Than it will use the first adapter which is enabled at that point in time.

For our test with LANX we have used the setup as shown inFigure 23-20. But running that node we see that at run time a different adapter was used:

Start the Node Operation by clicking at the Windows screen at **Start - Programs - IBM Personal Communications - Administration and PD Aids - SNA Node Operations**. At this panel click the green Start button or select from the menu **Operation - Start Node** and chose your node configuration file to apply.

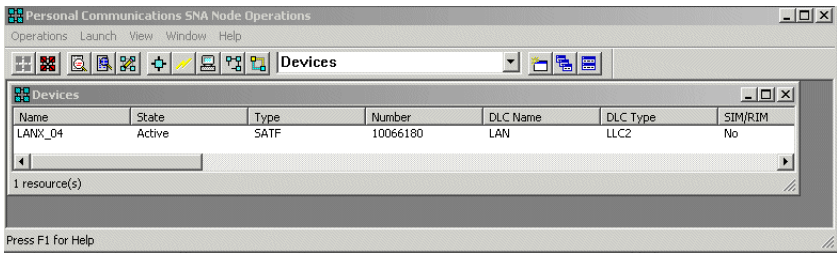


Figure 23-21 Node Operations panel

Once the node is up and running click the Log Viewer icon or choose from the menu **Launch - Log Viewer**. You will see the used adapter and its address.

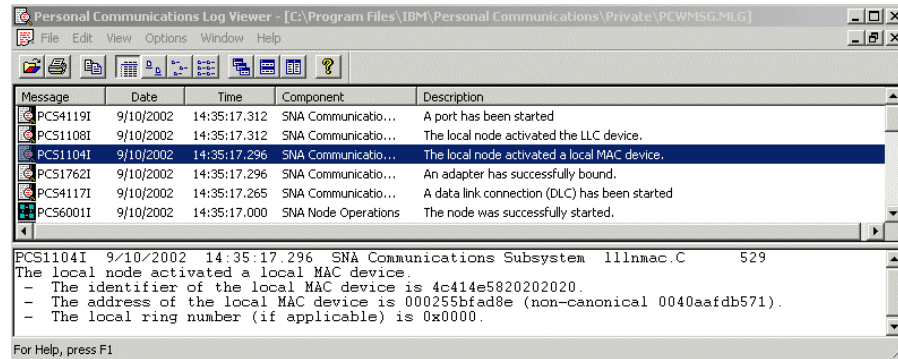


Figure 23-22 Log viewer showing address of adapter in use

In our example we had only one adapter plugged in into the network which as well was used for IP - as it would be on most workstations nowadays. So we could make a quick comparison for the adapter address used by issuing from the command prompt:

ipconfig /all

Example 23-3 Output of ipconfig /all

C:\Documents and Settings\zorn>ipconfig /all

Windows 2000 IP Configuration

```
Host Name . . . . . : mkaOkkh1
Primary DNS Suffix . . . . . :
Node Type . . . . . : Hybrid
IP Routing Enabled. . . . . : No
WINS Proxy Enabled. . . . . : No
DNS Suffix Search List. . . . . : itso.ral.ibm.com
```

Ethernet adapter Local Area Connection 2:

```
Connection-specific DNS Suffix . : itso.ral.ibm.com
Description . . . . . : Intel(R) PRO/100+ PCI Adapter
Physical Address. . . . . : 00-02-55-BF-AD-8E
DHCP Enabled. . . . . : Yes
```

Both SNA and IP used the same adapter, which was the only one plugged in and enabled.

23.1.16 Manual adapter selection

For using the manual configuration here is how the adapter should be selected: The operating system starts counting the adapters beginning with 1 - Personal Communications starts counting with 0 (= LAN0_04 - wheras 04 is the SAP)- see Figure 23-23 for adapter listing in Windows registry. Because we know we have plugged in the adapter #7 which is the second adapter in the registry LAN1_04 would be the enabled adapter in this example as it can be seen as well in Figure 23-19 on page 828.

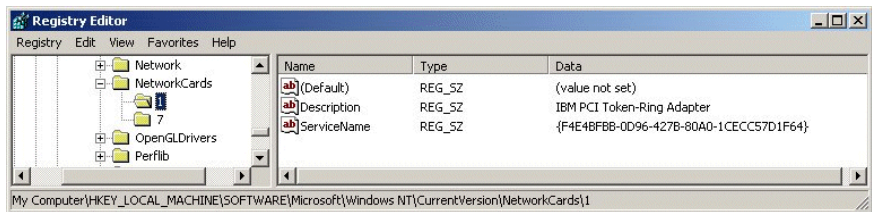


Figure 23-23 Network cards as listed in registry

23.1.17 Drivers for communication adapters

Personal Communications Version 5.6 includes a set of drivers for communication adapters. XP certification allows only PnP adapters to be installed along with its application. Required adapters which are not PnP but are still supported by Personal Communications have to be installed manually. They are contained on the product CD at subdirectory \admin\drivers. Use the drivers from that subdirectory when prompted during the installation process for the driver. For Plug and Play adapters the operating system will prompt you to install the driver after the adapter has been plugged in. For non-PnP adapters you have to install the driver via the Add a new hardware dialog or wizard. Details for those procedures are pointed out in online documentation *Personal Communications Version 5.6: CD-ROM Guide to Installation Appendix A*. Please follow the step by step instruction carefully. Additional information is contained in online documentation *Administrator's Guide and Reference, Chapter 5 "Attachment Considerations and Adapter Setup Information"*

In case problems occur during adapter installation delete the adapter using the device manager, close device manager and re-boot. Start the installation process again.

Table 23-1, Table 23-2 and Table 23-3 show available adapter drivers.

When Personal Communications Version 5.6 is uninstalled it will as well uninstall the communication adapters which had been installed during the new installation. When installing Personal Communications Version 5.6 over a previous version of Personal Communications such an uninstall process is executed and the device drivers of the previous version are deleted. Please install the new device drivers which came with the new version of Personal Communications which you are currently installing.

Table 23-1 3270 Adapters

3270 Adapters	Type	Plug and Play	Supported for WinXP & Win2000
IBM PCI 3270 Emulation Adapter	PCI	PnP	yes
IBM PCMCIA 3270 Emulation Adapter	PCMCIA	PnP	yes
IBM ISA 3270 Emulation Adapter	ISA	non-PnP	yes

Table 23-2 MPA, WAC and SDLC Adapters

MPA/WAC/SDLC Adapters	Type	Plug and Play	Supported for WinXP & Win2000
IBM PCI Multiprotocol Adapter	PCI	PnP	yes
IBM PCI Multiprotocol Adapter II	PCI	PnP	yes
IBM PCMCIA SDLC Adapter	PCMCIA	PnP	yes
IBM PCMCIA SDLC Modem	PCMCIA	PnP	yes
IBM PCMCIA SDLC Modem-2	PCMCIA	PnP	yes
IBM ISA Multiprotocol adapter	ISA	non-PnP	yes
IBM Async/SDLC ISA adapter	ISA	non-PnP	yes
IBM ISA Wide Area Connector adapter	ISA	non-PnP	yes

Table 23-3 5250 Adapters

5250 Adapters	Type	Plug-and-Play	Supported for WinXP & Win2000
IBM 5250 Express PCI Adapter	PCI	PnP	yes
IBM 5250 Emulation PCMCIA Adapter	PCMCIA	PnP	yes

5250 Adapters	Type	Plug-and-Play	Supported for WinXP & Win2000
IBM 5250 Express PC Card	PCMCIA	PnP	yes
IBM 5250 PCMCIA Adapter Card (Asia-Pacific) PCMCIA	PCMCIA	PnP	yes
IBM 5250 Express ISA Adapter (in PnP mode)	ISA	PnP	yes
IBM 5250 Express ISA Adapter (in non-PnP mode)	ISA	non-PnP	yes
IBM Enhanced 5250 Display Station Emulation Adapter	ISA	non-PnP	yes
IBM 5250 AT-Bus Communications Adapter (Asia-Pacific)	ISA	non-PnP	yes

Micro channel adapters are not supported!

23.1.18 Enhanced Mouse Marking

Personal Communications Version 5.6 has enhanced the marking of a text area in the emulator window. The trim rectangle now snaps to the boundary of the character cell while it is stretched with the mouse. This feature is controlled by the Expand Trimm Rectangle during drag check box in the session pull down menu Edit - Preferences - Edit as shown in Figure 23-24 on page 834. It is enabled by default.

This feature can be enabled in Personal Communication Version 5 by adding a parameter to the WS file: trimRectJumpToChar=Y.

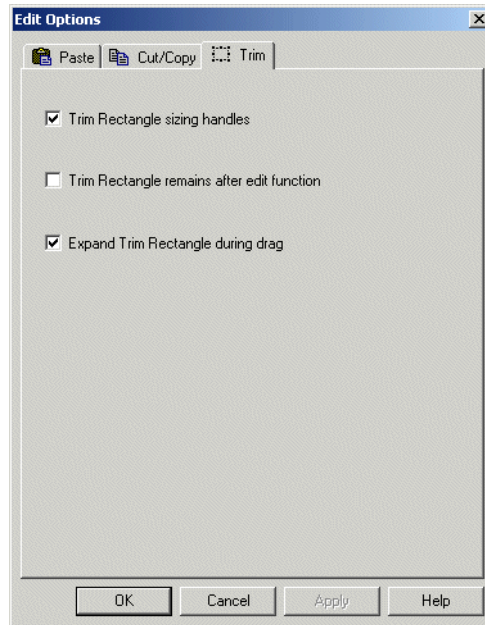


Figure 23-24 Expand Trim Rectangle during drag

23.2 Previous enhancements made to Personal Communications 5.0

The following chapter explains the major enhancement which had been made to the Personal Communications 5.0 before version 5.6 was introduced

23.2.1 Windows 2000 certification

Personal Communications Version 5.5 holds Microsoft Windows 2000 certification.

23.2.2 Session Manager

The Personal Communications Session Manager provides an improved convenient graphical interface for starting, creating, and managing workstation profiles and batch files.

The Session Manager is a folder that shows the available single (*.ws files) and multiple (*.bch files) session profiles and definitions. Its functions have been improved in Personal Communications Version 5.5 to be more versatile. From Session Manager, sessions can be opened, re configured, or deleted, and new sessions can be added.

The standard windows drag-and-drop functions are implemented in the Session Manager. To move a profile you can drag it with the left mouse button and drop the icon where you want the profile to be. To copy a profile instead of moving it, you can drag it with the left mouse button with the Ctrl key held down. To create a shortcut to the profile, you can drag it with the left mouse button with the Alt key held down. If you don't remember the control keys, you can drag with the right mouse button and a context menu displaying the choices will pop up when you drop the icon.

You can also import definition files for single and multiple sessions from other locations so that they are added to the Session Manager view. You may have a directory of Personal Communications profiles that you want to have in your Session Manager that are not currently in your Application Data location (for example they are in a folder on a network server). You can drag these files from Windows Explorer and drop them into the Session Manager's icon container. The Session Manager will copy the profiles to the correct location based on the current Application Data location and also migrate them to the current release of Personal Communications.

Note: If you only want to run some of the profiles once, you can use the Change Directory menu item under the File pull-down menu to change your directory to the one on the server; however, the next time you start the Session Manager, it will return to the current Application Data location. The profiles will not have been imported to your Application Data location.

If you are copying a session icon to the Start Menu or the desktop, you will want to create a shortcut to the profile and not move or copy the file. More details of the Session Manager are provided in Chapter 4 of the *Quick Beginnings* online documentation.

To start the Session Manager, select **Start -> Programs -> IBM Personal Communications -> Start or Configure Sessions**. If no session is selected, the status bar shows the number of available single and multiple sessions. If a session is highlighted, the status bar shows the subdirectory and the name of the session profile.

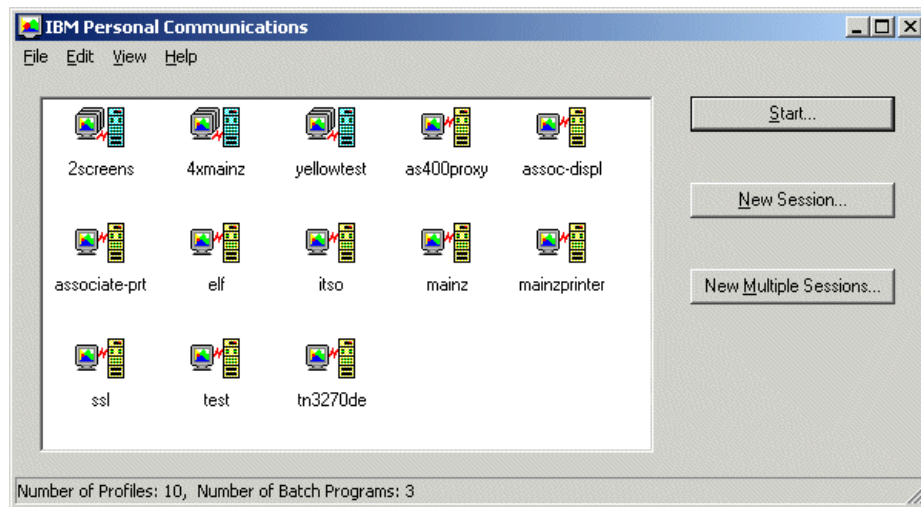


Figure 23-25 Session Manager window

Click **View -> Details** to see for each session the host address, attachment type, host type, and other values.

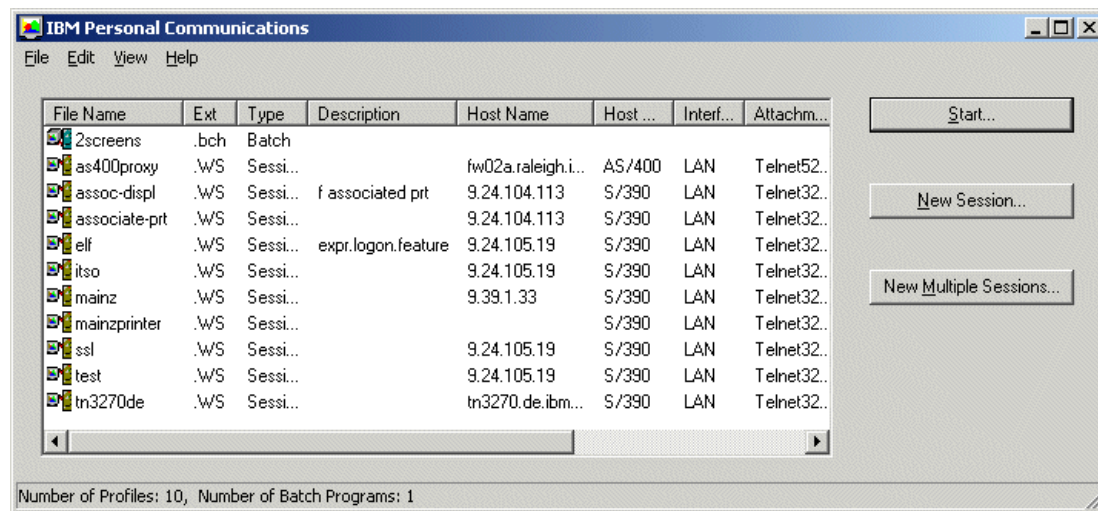


Figure 23-26 Showing Session Manager window with details

Changing or adding sessions is different for single sessions (.ws files) and multiple sessions (.bch files). See the following list:

- ▶ Multiple sessions
 - Add session: click **New Multiple Session**
 - Modify existing session: select session from left pane with right mouse click and click **Modify**
- ▶ Single sessions
 - Add session: click **New Session**
 - Modify existing session: double-click at session so that session starts. Click **Communication** -> **Configure** from the session menu bar.

Multiple session configuration files and workstation profile configuration files are ASCII files that can be edited with an editor such as NotePad.

Note: Modify is only available for multiple sessions (.bch files). Single sessions must be started first so that the Configure option is available via the menu bar of the session

Delay between starting sessions

One option for multiple sessions that is not described in the online documentation is the delay option,!pxs, where x is the number of seconds for the delay. The !pxs parameter might be useful if specific timing is required. Example 23-4 shows the usage for a 9-second delay between the start of two sessions.

Example 23-4 Delay example

```
Run13=c:\Program Files\IBM\Personal Communications\pcsws.exe "C:\Documents and
Settings\zorn\Application Data\IBM\Personal Communications\itso.WS"
Run14=!p9s
Run15=C:\Program Files\IBM\Personal Communications\pcsws.exe "C:\Documents and
Settings\zorn\Application Data\IBM\Personal Communications\itso.WS"
```

Options for the RUN (pcsws.exe) command

You can use the following options to control the start of sessions within the.bch batch file:

Table 23-4 Available options for the RUN command

Option	Result
/q	quiet - no logo window
/i	start session iconized
/ h	start session hidden

Option	Result
/s= <i>m</i>	show short session ID " <i>m</i> "
/v= <i>myview</i>	use for session the save view " <i>myview</i> "

For example:

```
c:\pcomm\pcsws.exe c:\pcomm\private\tn3270.ws /i /q
```

This command will start the session, but instead of placing the emulation window on the desktop, an icon will be placed onto the task bar. No Personal Communications startup logo will appear.

For further details, see Chapter 4 in the online documentation in Personal Communications *Quick Beginnings*.

Note: Using the Create/Modify Batch File utility, the "RUNxx=" lines are not seen, while they exist in the *.bch file when editing it with a common editor.

23.2.3 CSD and APAR tool

Authorized users may use the WebUpdate tool in the Administrative and PD Aids menu to check for Personal Communications Corrective Service Distributions (CSDs) and authorized program analysis reports (APARs) via the Internet.

The Product Update Tool allows you to manage CSDs (Corrective Service Distributions) and APARs (authorized program analysis reports) for pcomm.

To launch the utility click **Start -> Programs -> IBM Personal Communications -> Administrative and PD Aids -> Product Update Tool**. The window shown in Figure 23-27 will be displayed.

Before you can use this tool, you must configure your Internet connection. If you have not previously configured your Internet connection, you may do so by clicking **Connection Configuration**. If you have already set up your Internet connection for your Internet browser, there is no need to fill in any data in the Connection Configuration window. The data from your Internet browser will be used.

If you are behind a proxy firewall, update that information in Connection Configurations. If you connect through a SOCKS server, place socks= in front of the SOCKS server address in the Proxy Server Address field. The Product Update Tool does not support the use of Automatic Configuration Scripts.

The following are the major functions available via the Product Update Tool:

- ▶ Test an APAR
- ▶ Remove an APAR
- ▶ Committing an APAR
- ▶ Check for CSDs

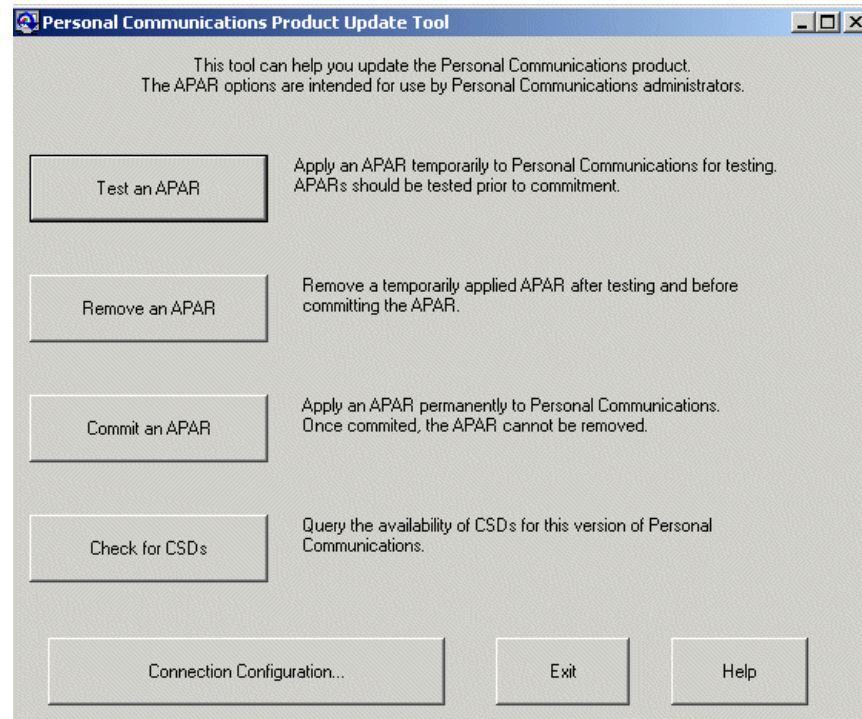


Figure 23-27 Product Update Tool

Rebooting of the system is only necessary in cases where the modules updated cannot be updated, which occurs when the module is locked because it is in use. The tool will recognize this and will inform the user if a reboot is required.

Test an APAR

This option allows the user to test an APAR before installing permanently. An APAR installed via this option can be removed with the Remove APAR option. If you select **Test an APAR**, the Product Update Tool will go to the IBM support site on the Internet, obtain a list of available APARs for pcomm, and display the list as shown in Figure 23-27. APARS that are already applied to this instance of Personal Communications are not displayed in the list.

You may then select the APAR(s) desired and apply them.

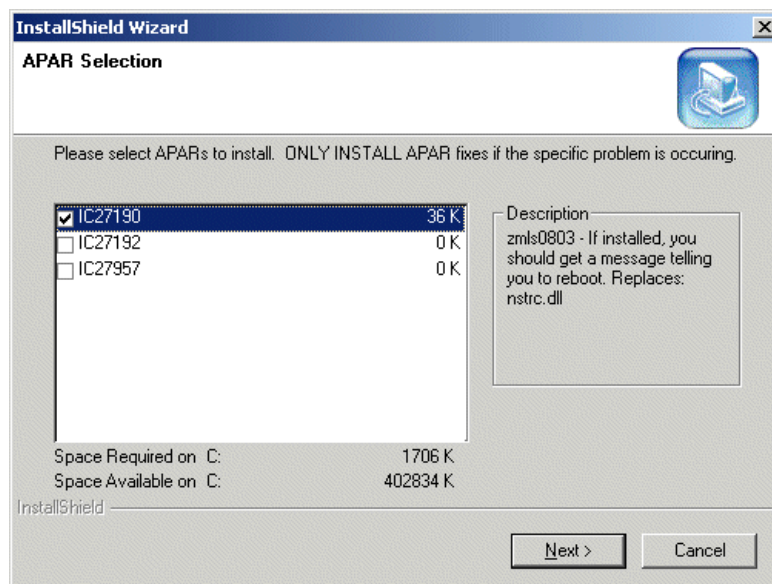


Figure 23-28 List of APARs available for download and installation

Remove an APAR

APARs that have been installed using the option **Test an APAR** can be removed using Remove an APAR option. In the Product Update Tool window, shown in Figure 23-27, click **Remove an APAR**. A list of APARS available for removal will be displayed (see Figure 23-28). Mark the APARS to be removed and click **Next** and follow the procedures to remove the APAR. The CSD and APAR Installation Utility cannot keep track of manually installed APARs; therefore, APARs that have been installed manually (by simply overriding existing product files) cannot be removed.

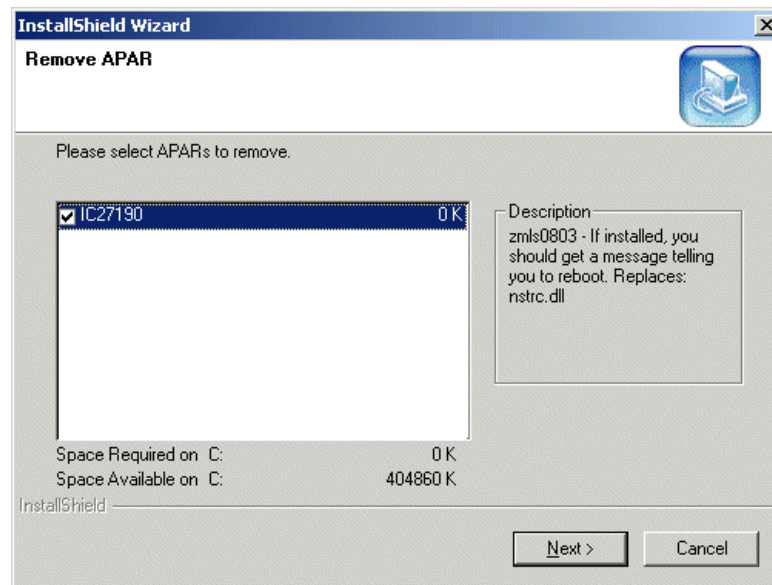


Figure 23-29 APAR removal

Committing an APAR

You do not need to test an APAR prior to it being committed. Testing and committing are independent processes. If you want to install an APAR permanently, click **Committing an APAR** in the Product Update Tool window shown in Figure 23-27 on page 839. A list with all available APARS is shown, including those installed for testing. APARS that are already committed are not listed. During the commit process, a window will appear, listing active tasks that might interfere with the commit process, for example active Personal Communications sessions. The Product Update Tool itself is one of the listed tasks. Close the listed tasks, including the Product Update Tool, and click the **Retry** button to complete the process.

After an APAR has been installed for testing or has been committed, the file `pcommaparinfo.txt` in `xxx\APARs` will be created/updated, where `xxx` represents the subdirectory you selected at installation time as your Data Application Location. That file will contain a list of the APARs in test and committed. For example:

IC27190[c]

IC27192[t]

Where [c] is committed and [t] is tested.

Do not attempt to edit this file. Its content is used by the Product Update Tool, which might not work properly if that file has been changed.

The commit process requires access to the original installation files, so if you installed Personal Communications from the CD, you must have the CD in the CD-ROM drive when committing an APAR. If you installed Personal Communications off the network, you must be connected to the network when committing an APAR. If the source is not present, you must provide the location of the source files.

Administrator information

When you install a product using Microsoft's Windows Installer, the .msi file, in this case IBM Personal Communications.msi, gets cached on the machine. This MSI file is a database containing information on all the files that were installed. When you test an APAR, the .msi file does not get updated with the updated files information. This allows the removal of the APAR if there were undesirable effects. When you commit an APAR, you are using something called a Microsoft Patch file (.msp file). The .msp file updates the product's .msi file with patch information, thus making the patch (or APAR) part of the product.

Also, the repair functionality uses the .msi file to see if files on the machine are the correct files. If a file's information varies from that in the .msi file, for example, the file version is different, the correct file, with the information matching that in the .msi file, gets placed back on the machine.

Important: Do not repair Personal Communications while testing an APAR. This causes the APAR to be removed. It is important that an APAR gets committed and is not left in test mode.

Make sure the Windows Scripting Host Service is running. The Windows Scripting Host Service runs on Windows operating systems by default; however, it can be turned off. The Product Update Tool uses .vbs files to download the correct APAR package, so if you have turned off the service, turn it back on when using the tool.

You may also obtain the test and commit packages directly from the Internet. Find these packages at:

<http://ww6.software.ibm.com/aim/pcL55MMP.exe>

Where:

pc	Personal Communications
L	Language - one of the following:
m	Multi-Language Support

s	Simplified Chinese
t	Traditional Chinese
k	Korean
55	Version 5.5
MM	Modification/CSD level; 00 is the base level for the release
P	Type of package is one of the following:
t	Test package
c	Commit package

The following illustrates the appropriate files to download for the base release of Personal Communications Version 5.6 V5.5:

- ▶ To obtain the commit APARs package for the base V5.5 release, use the appropriate link below:
 - Personal Communications 5.5 MLS Commit Package
<http://www6.software.ibm.com/aim/pcm5500c.exe>
 - Personal Communications 5.5 Simplified Chinese Commit Package
<http://www6.software.ibm.com/aim/pcs5500c.exe>
 - Personal Communications 5.5 Traditional Chinese Commit Package
<http://www6.software.ibm.com/aim/pct5500c.exe>
 - Personal Communications 5.5 Korean Commit Package
<http://www6.software.ibm.com/aim/pck5500c.exe>
- ▶ To obtain the test APARs package for the base V5.5 release, use the appropriate link below:
 - Personal Communications 5.5 MLS Test Package
<http://www6.software.ibm.com/aim/pcm5500t.exe>
 - Personal Communications 5.5 Simplified Chinese Test Package
<http://www6.software.ibm.com/aim/pcs5500t.exe>
 - Personal Communications 5.5 Traditional Chinese Test Package
<http://www6.software.ibm.com/aim/pct5500t.exe>
 - Personal Communications 5.5 Korean Test Package
<http://www6.software.ibm.com/aim/pck5500t.exe>

Once you commit an APAR, the .msp file is in the following locations:

- ▶ On Windows 9x and Windows ME, under the Windows directory in
Application Data\IBM\Personal Communications\APARs*APAR name*
- ▶ On Windows NT, under the Windows directory in
Profiles\All Users\Application Data\IBM\Personal Communications\APARs\A
PAR name
- ▶ On Windows 2000 and Windows XP, on the Windows drive in
Documents and Settings\All Users\Application Data\IBM\Personal Communi
cations\APARs*APAR name*

Once the administrator has committed an APAR to a system, the resulting .msp file can be used to commit the maintenance to other systems. We recommend that you launch the installer from a command prompt using the following syntax:

```
msiexec.exe /p [patch.msp] REINSTALLMODE=em
```

The .msp files are typically very small so the administrator could use whatever distribution mechanism at their disposal to distribute the files to end users and then have a batch file to execute the utility to apply the maintenance.

Many customers create master images of the products on a desktop, which they then replicate on new or rebuilt workstations. The installer utility may be used to apply the maintenance to such a master image by executing the installer utility using the following syntax:

```
msiexec.exe /a "Absolute Path to IBM Personal Communications.msi" /p  
"Absolute Path to .msp" REINSTALLMODE=em
```

Note: The msiexec.exe tool is found in the Windows System Directory, for example on a Windows 2000 system this would be \WINNT\system32.

Check for CSDs

CSDs are packages containing a bundle of APARs and possibly new options and improvements of the product. For Personal Communications Version 5.6, new CSDs are cumulative, and because CSDs are major recompiled parts of Personal Communications they may only be obtained if the user has purchased a full version of the product. This is validated by entering a service key. The service key is included in the original package of pcomm. After clicking the **Check for CSDs** button in the Product Update window, your Internet browser opens to a window that looks like the one shown in Figure 23-30. Here you are guided through the Web pages for entering your registration data and service key.



Figure 23-30 Start window for downloading CSD packages

Once registration is complete and the service key is entered, you are guided through the windows where you can select the product, version and language for which you need the CSD update. Finally, you reach the download page for the selected CSD. Since there were no CSDs for Personal Communications Version 5.6 as this book was written, we are using an example of Version 5.0. Please note that a CSD file is usually larger than 100 MB.

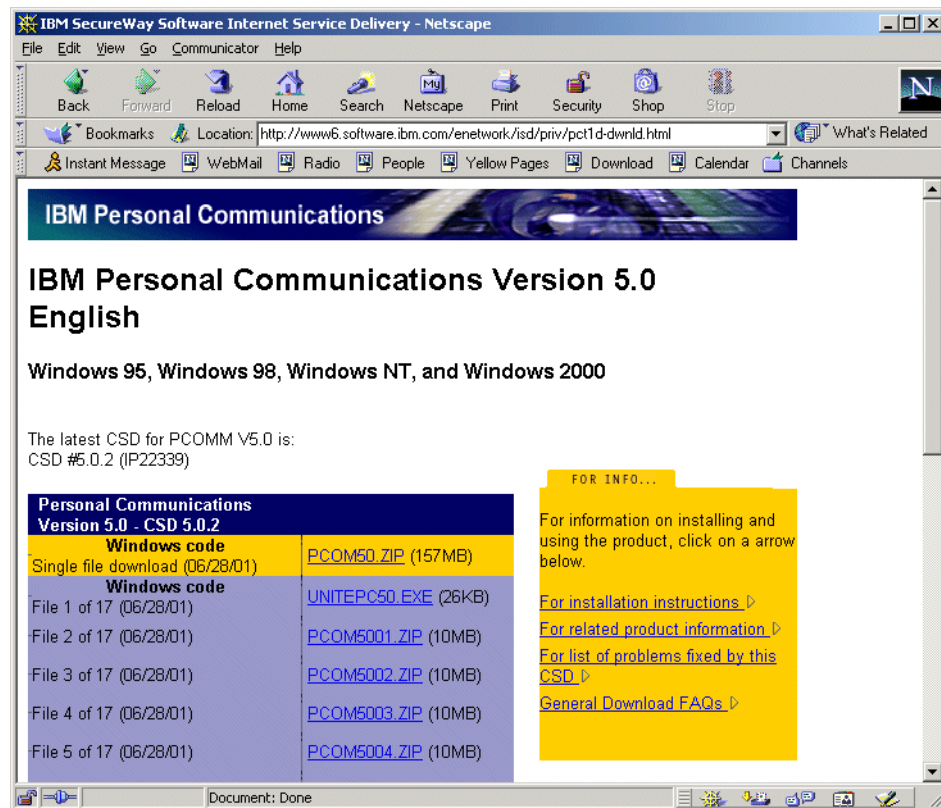


Figure 23-31 Download window for a Personal Communications Version 5.6 CSD

Note: Installing a CSD is like a re-installation of the product (removing old product files and installing the new ones); therefore, there is no test and commit available for CSDs. To remove a CSD, your only option is to uninstall the new pcomm level followed by an reinstallation of the level as it was before.

For more details, see Chapter 2 in the online documentation *Personal Communications Administrator Guide and Reference*.

23.2.4 Tivoli support

Personal Communications Version is Tivoli-Ready certified. It is integrated with Tivoli Enterprise as a desktop application, and provides a Plus Module for its application management. The Plus Module, called IBMPCOMM Plus, includes event management, software distribution, and administrative tasks. Support for problem determination is new. The Plus Module now includes tasks for remote operation of the Integrated Trace facility.

Personal Communications Version 5.5, now supports, in the IBMPCOMM Plus Module, three tasks to support problem determination. These tasks allow the administrator remotely to set up, start, stop, and format and bundle the data at the target machine. Do not use the CSTRACE via the Issue_Command Task. Use Start_Trace, Stop_Trace and Format_Trace as pointed out in Chapter 6 in the online documentation *Personal Communications Administrator's Guide and Reference*. General information about the IBMPCOMM Plus Module and its functions is contained in Chapter 8 of *Personal Communications Version 4.3 for Windows 95, 98 and NT*, SG24-4689. This chapter will cover the new functions only.

The IBMPCOMM Plus window (Figure 23-32) shows three new functions:

- ▶ Start_Trace
- ▶ Stop_Trace
- ▶ Format_Trace

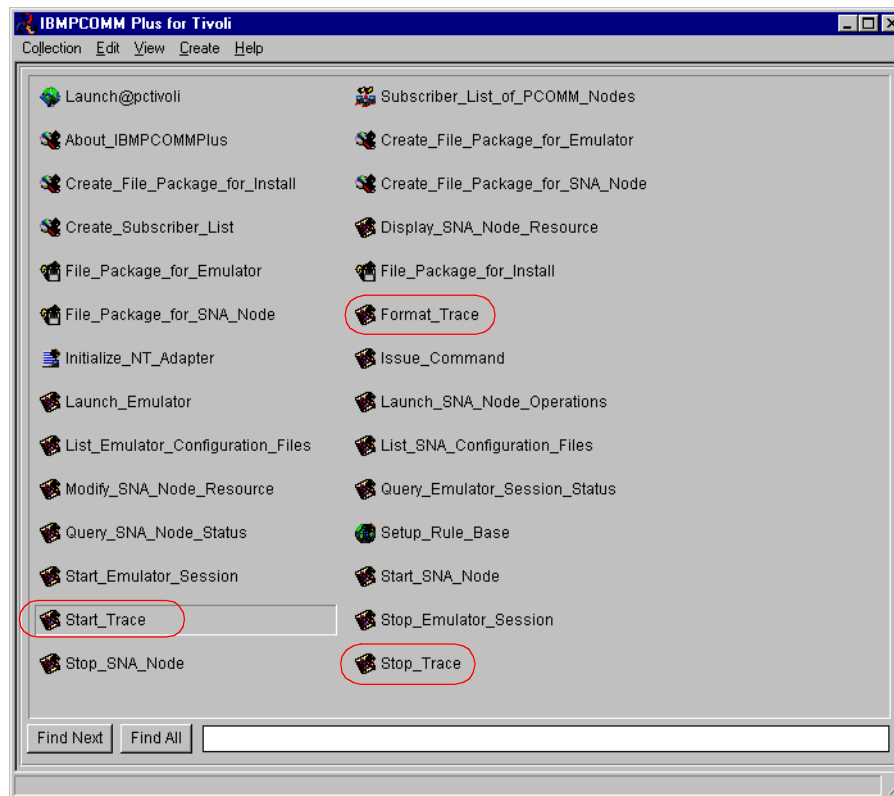


Figure 23-32 IBMPCOMM Plus window at Tivoli TMR Server

The following example shows how to use the new functions for tracing Personal Communications on a Tivoli managed node. Make sure that the node where the trace is to run is registered in the subscription list of the TMR server. On the IBMPCOMM Plus for Tivoli window, select **Start_Trace**. A window (Figure 23-33) will appear asking for an option file that contains the trace settings.

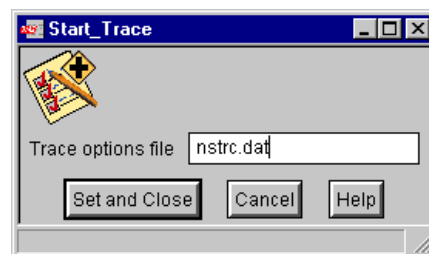


Figure 23-33 Starting a trace from the IBMPCOMM Plus Module

A sample file NSTRC.DAT containing all possible trace settings is supplied with the CD of Personal Communications Version 5.5, and is located in subdirectory \install\admin\distrib. It is an ASCII file that can be edited to suit your needs. Unwanted trace options should be flagged with a semicolon to be treated as a comment. The trace option file must be located at the node in a system class subdirectory depending of your data application location. For more details on tracing via a command line please see as well chapter “Tracing from the command line” on page 914

Note: A trace can only be started with that function from the IBMPCOMM Plus window. Do not attempt to use CSTRACE via a command-line interface to the node.

For additional information, see the help information for Start_Trace.

After the trace has been started at the node, it returns an output equivalent to the CSTRACE STATUS command, showing the applied trace options as shown in Figure 23-34.

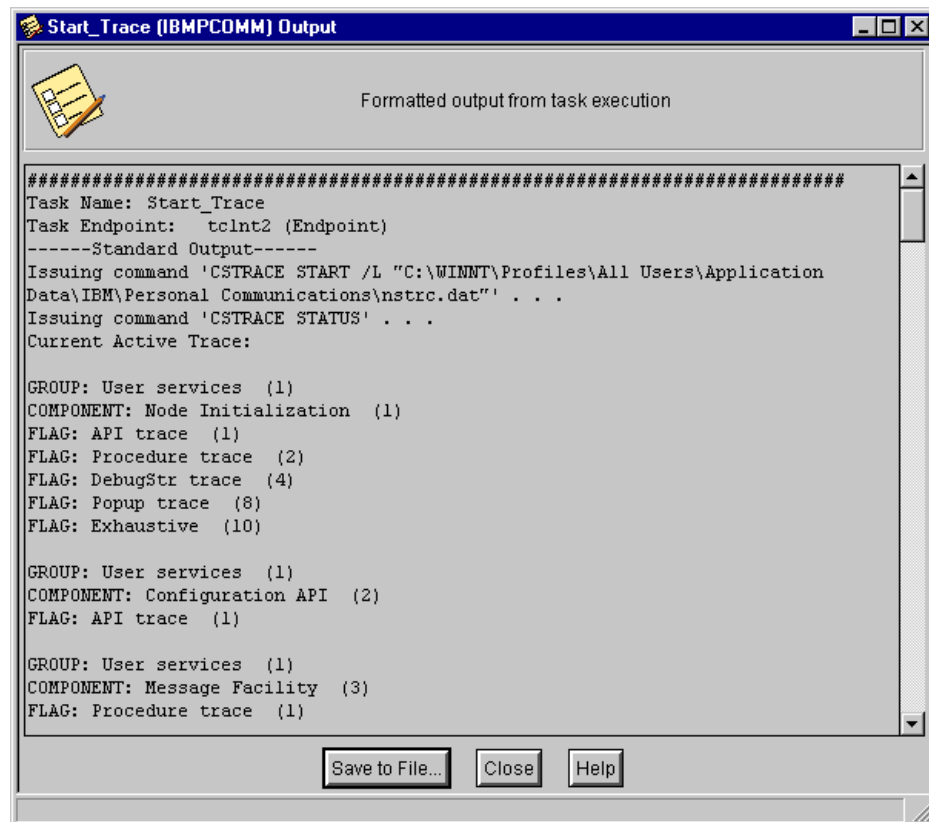


Figure 23-34 Response after starting the trace

Re-create the problem while the trace is running. After the problem has been re-created, stop and save the trace, using the **Stop_Trace** from IBMPCOMM Plus. This causes the window shown in Figure 23-35 to pop up, requesting a name for the output file for the raw unformatted trace. Remember the name of the file, since there is no Browse button in follow-up windows for you to navigate to it. The name must be retyped.

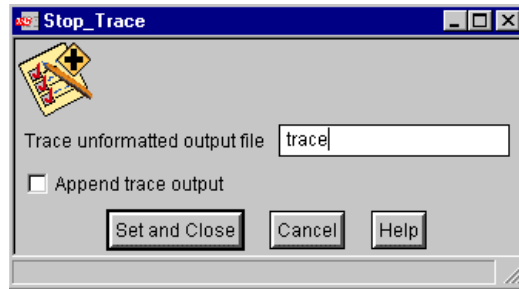


Figure 23-35 Name of raw trace to save

In most cases you do not want to append the trace to an existing trace, so do not check **Append trace output**. A status from the node will be shown verifying that the chosen action was successful. After stopping and saving the trace, a CSTRACE SHUTDOWN command is carried out at the node, clearing all trace options and buffers. The node is now ready to run a new trace command with new options.

Next, you must format the saved raw trace, save the formatted trace as an ASCII file, and collect all Personal Communications data, including the trace, into an Information Bundler file, x12345.exe. This is done by selecting **Format_Trace** from the IBMPCOMM Plus window (Figure 23-32 on page 848). This command presents a window requesting the name of the raw unformatted trace file that is to be formatted. Type in the name of the file from the Stop_Trace command, and click **Set and Close**.

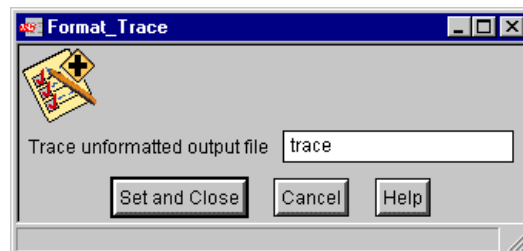


Figure 23-36 Name of formatted trace output

The node will return a window confirming the actions. The data will be stored in the file you specified in the Format Trace window (see Figure 23-36), as well as in the Information Bundler file, x12345.exe. The Information Bundler file will be needed for problem determination by IBM.

23.2.5 Defining the session view from a batch file

When starting a session from a batch file, you can define the view (fonts, window size and window position) to be used with that session.

Personal Communications Version 5.5 has a new function that allows you to capture the layout of all currently opened sessions (position and size on the screen). This is called a *view*. This view can be used to create a batch file (.bch) that will launch all these sessions and automatically position them on the screen in the same positions as when the view was captured.

To use that new function, start all desired sessions, place them in their desired locations on the screen, at the desired size, and using the desired display font. A four-session example is shown in Figure 23-37.

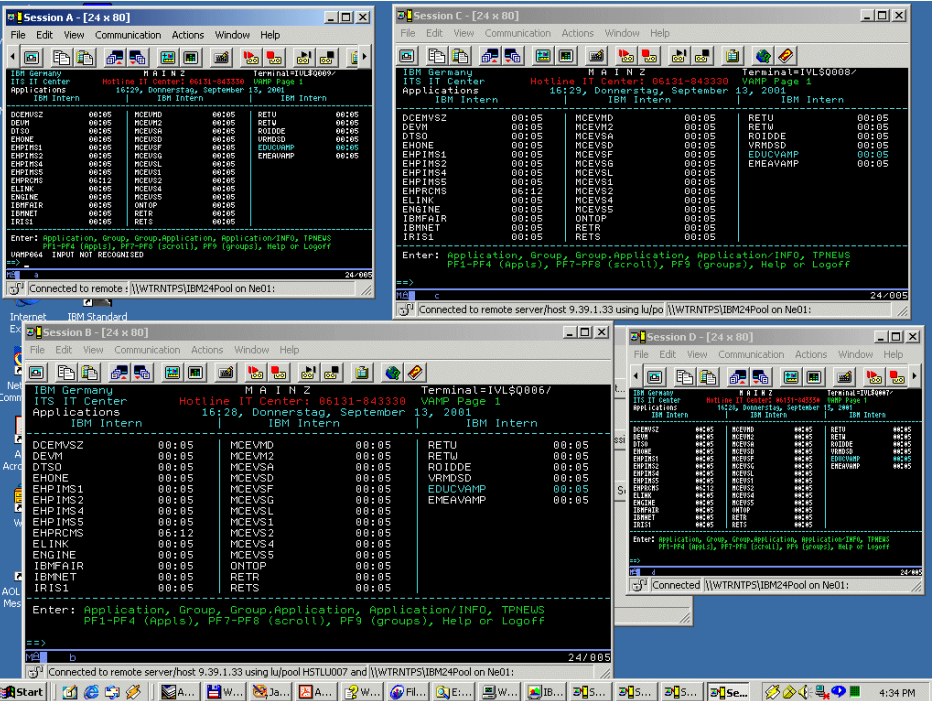


Figure 23-37 Using .bch batch file with a four-session view

Next, start the Personal Communications Session Manager and click **New Multiple Sessions....** Then click **Capture View** to add all current sessions to the right pane of the Session Manager's Create/Modify Batch File window. Save that .bch file. When starting it the next time, you will get the sessions, their positions, size and font as they had been saved.

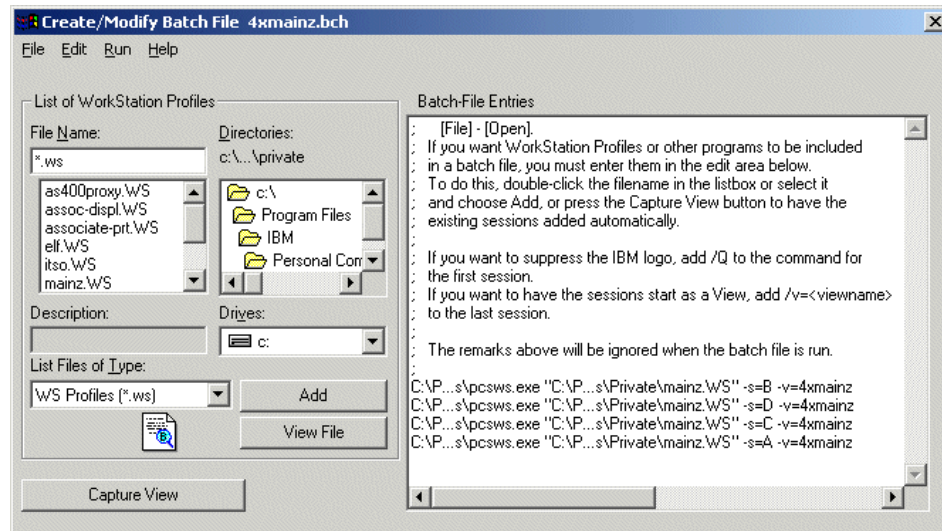


Figure 23-38 Result of Capture View button

If you wish to change any setting of that saved view, start the .bch file from the Session Manager. When the sessions appear, change them as desired, then click **View -> Save/Delete View...** on the menu bar of any of the current sessions, and save the changed view. The current appearance of all active sessions is now captured in the updated view. The next time you start your .bch file, all the sessions appear according to the new view setting.

Note: The view definition is a new part of the pcswin.ini definition file. That stores the size, position and fonts of sessions. All other settings, including color definitions, are part of the session profile in the .ws file. The definitions in the .ws file are not changed by the view function. Those are changed by clicking **Edit -> Preferences** from the Session menu bar and will be saved in each .ws file individually.

23.2.6 Convert macro to XML

Using the Convert macro to XML, you can deploy the converted macro of Personal Communications in Host On-Demand. This new function is described in 26.1, "Macros" on page 898.

23.2.7 Telnet 3270E printer association

The Telnet 3270E standard lets you specify an association between a display and a printer session. When a host is configured for printer association, the host specifies which printer is associated with which terminal. Thus, you do not need to know what LU, port number, and other parameters are needed for the printer session.

Telnet printer association maps a host-specific printer LU to a specific display session. The functionality is available only for TN3270E servers and clients. For an example of how to set up the OS/390 Telnet server to do 3270 associate printers refer to Section 12.2.4.4, "Configuration and Routing" in *IBM Communication Server for OS/390 V2R10 TCP/IP Implementation Guide Volume 1*, SG24-5227. You may also refer to RFC2355, found at:

<http://www.ietf.org/rfc/rfc2355.txt?number=2355>

For the Telnet printer association to function, two configuration files (*.ws) must be defined at the Personal Communications client, one for the printer characteristics and one for the display session. The display session will reference the printer definition file to be used for the associated printer session.

First you must define a printer session. There is no specific definition in the printer setup to make it an associated printer. Both displays and printers start their configuration with the window shown in Figure 23-39, where you provide the required host definition information.

Telnet3270

Host Definition | Automatic Host Location | Advanced Security Setup

	Host Name or IP Address	LU or Pool Name	Port Number
Primary	9.24.104.113		23
Backup 1			23
Backup 2			23

Printer Association (only valid for TN3270E Display sessions)

Associated Printer Session: Browse...

☒ Start Associated Printer Minimized

☒ Automatically close the associated printer session with this session

☐ Auto-reconnect

☐ Enable Security

OK Cancel Apply Help

Figure 23-39 Host definition for associated printer

Note that the printer session definition should have an empty LU name field. Associated printers are typically controlled by the communications server and you will typically not know the LU name, especially if you are using pools. The LU name will be determined and supplied by the communications server as part of the session negotiation process. When you provide the Session Parameters for the printer session, as shown in Figure 23-40, you select **Printer** and click **OK**. You must save the printer session definition. In our example we named it "Associated Printer", which generated a file name of ASSOCI~1.WS. Once the printer session has been saved, you can define the display session.

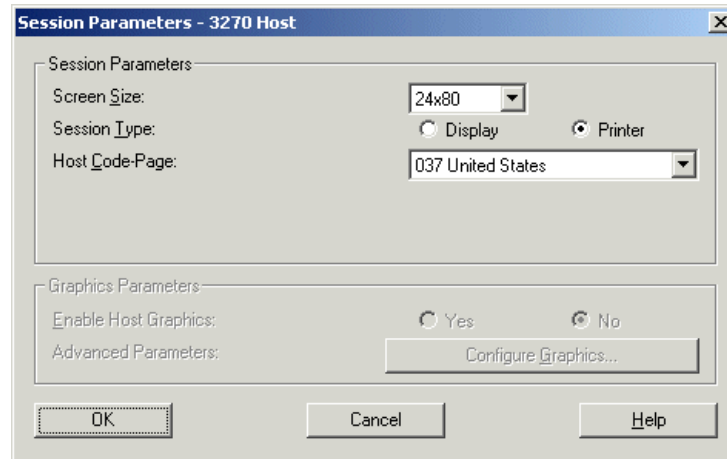


Figure 23-40 Session parameters for associated printer

When specifying the Associated Printer Session for the display session (see Figure 23-41), you must select the drop-down list and select the file that contains the appropriately defined printer, in our example ASSOCI~1.WS. It is recommended that you check **Start Associated Printer Minimized** and **Automatically close the associated printer session with this session** for a clean desktop and housekeeping. Finally, save the display session definition.

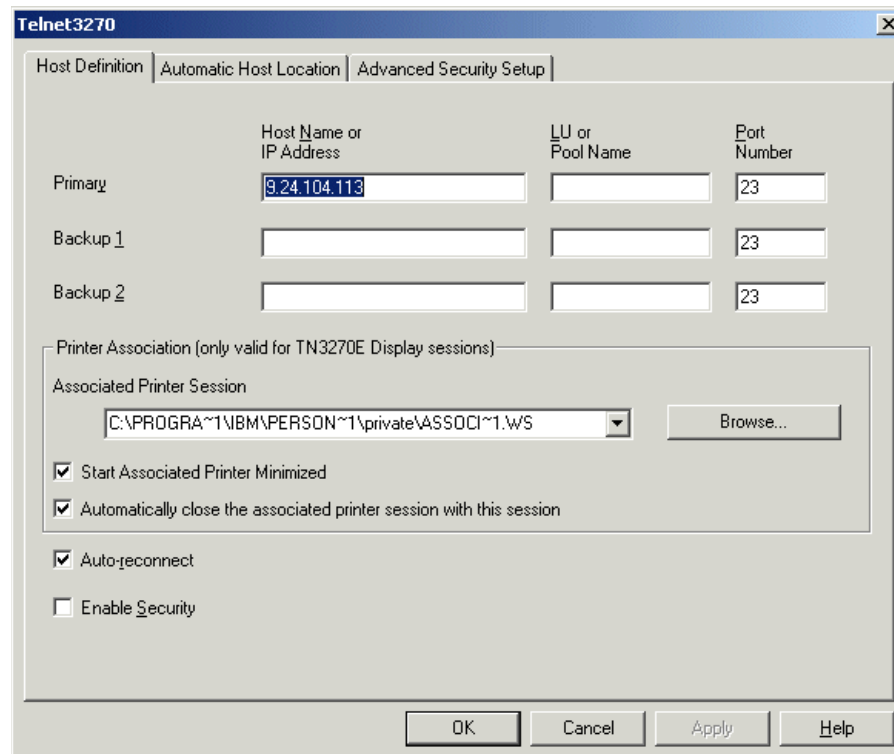


Figure 23-41 Host definition for display session of associated printer

The associated printer session is a normal printer session. At the pull-down menu of the printer session, click **File -> Printer Setup...** and select the printer of your choice.

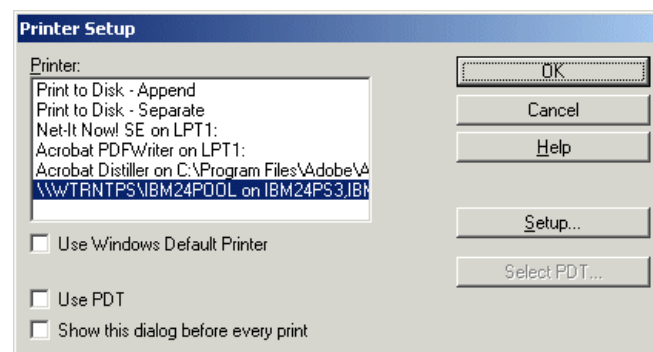


Figure 23-42 Printer Setup

When you start the display session, it will be assigned an LU name. Personal Communications will determine that there is an associated printer defined so it will automatically start that session. The LU name of the display session will be passed to the Telnet server, which will determine the appropriate printer LU to assign the printer. A more complete description of this process may be found in 19.3.2, "How an associated printer session works" on page 685. Additional documentation may be found in Chapter 5 of the online documentation *Personal Communications 3270 Emulator User's Reference*. Additional details for configuring PDT / PDF files can be obtained from the Host On-Demand online documentation.

23.2.8 Win32 cut, copy, and paste hotkeys

For 3270 and 5250 sessions, Personal Communications includes two .KMP keyboard map files that map the standard Win32 hotkeys for Cut, Copy, and Paste to Ctrl+X, Ctrl+C and Ctrl+V, respectively.

The user may utilize the new keyboard map files, or may add these new key values to an existing map file. The key values for the following 3270 functions have been changed: Page Up, Page Down, Enter, and New Line.

The key values for the following 5250 functions have also been changed: Enter and New Line.

For 3270 sessions, the new map file is pscwinkb3.kmp. For 5250 sessions, the new map file is pscwinkb5.kmp. They can be used instead of the default IBM keyboard.

In order to offer a keyboard layout that has common keys for cut, copy and paste as for other Windows applications, Personal Communications offers two keyboard definition files, one for 3270 and one for 5250, which include those changes. From the menu bar, click **Edit -> Preferences -> Keyboard**. Select **User Defined**, then click the **Browse** button and select pscwinkb3.kmp for 3270 sessions or pscwinkb5.kmp for 5250 sessions.

If personal changes were made in releases previous to Personal Communications 5.5, they cannot be migrated, but must be made again.

23.2.9 Windows 2000 Power Management

Personal Communications complies with Windows 2000 Power Management requirements for handling sleep events (standby and hibernate). This support minimizes session interruptions due to network disconnections caused by sleep on Windows 2000 and subsequent versions.

Personal Communications complies with Windows 2000 Power Management requirements for handling sleep events. This support minimizes session interruptions due to network disconnections caused by sleep on Windows 2000 and subsequent versions.

Those new function do not have parameters to configure.

In this context, the term *sleep* means that the system is on standby or is in hibernation. To applications such as Personal Communications, standby and hibernation are the same. The benefits of this power management system include reduced power consumption via the Advanced Configuration and Power Interface (ACPI). The system is able to enter a lower power state (or sleep mode) that appears to be off; however, it is still able to wake up to handle timed events or device-related needs such as receiving a fax.

- ▶ The PC is instantly available to the user because it can rapidly return from a low power state to a fully functional state.
- ▶ Customers can rely on their PCs to power down and up in a way that is easily understood and predictable.

The following Personal Communications components are affected by this Power Management arrangement:

- ▶ Emulator sessions
- ▶ Transfers that utilize an emulator session

Sleep permission

Before entering a sleep state (standby or hibernate), Windows 2000 normally requests permission from the applications that are running. When one or more emulator sessions are connected and Windows signals that the user is available for interaction, Personal Communications asks the user to grant or deny sleep permission. If the user grants permission, Personal Communications logs the event and then notifies Windows. When user interaction is not possible, sleep permission is denied. When Personal Communications is not in the connected state, Windows 2000 may automatically go to sleep without prompting the user for permission.

Critical sleep

In cases of critical sleep, Personal Communications will not be notified by Windows 2000 of the impending sleep event. Personal Communications will be suspended without warning. After the computer wakes up from the critical sleep, Win2000 is supposed to send Personal Communications a PBT_APMRESUMECRITICAL notification.

Each connected Personal Communications emulator session will log message PCSWS046 and display a similar message on the status bar if and when the "critical sleep event occurred" message arrives.

How to initiate a sleep

- ▶ System-initiated sleep - set the system standby in the Power Options control panel to "After 1 min" and wait a minute.
- ▶ User-initiated sleep - Choose **Stand by** in the system Shut Down window.
- ▶ Critical sleep - Choose **Stand by** in the system Shut Down window, then hold down the Ctrl key while clicking **OK** or pressing the Enter key.

Possible problems

Refer to Table 23-5 for a list of possible sleep problems and probable causes.

Table 23-5 Sleep problems

Problem	Probable Cause
Windows 2000 won't go to sleep	A Personal Communications session is connected
Windows 2000 goes to sleep even when sessions are connected	In the event of a critical sleep, Windows 2000 doesn't ask applications for permission
After a critical sleep, message PCSWS046 is not displayed on the status bar of connected sessions	Personal Communications has not received a notification from WIN2000 after a wake from critical sleep

23.2.10 1390/1399 code page support

Personal Communications supports 1390 and 1399 code pages in Windows NT and 2000; this allows Japanese character processing. Refer to the *Administrator's Guide and Reference* for more information about 1390/1399 code page support.

New 1390/1399 code page support in Windows 2000 and Windows NT allows Japanese character processing. For details, see Chapter 16 in the online *Personal Communications Administrator Guide and Reference*.

23.2.11 Hindi support

Personal Communications supports 1137 Hindi code page use for Windows NT and 2000.

Details of the new Hindi support are located in Chapter 3 of the online documentation *Personal Communications Quick Beginnings*, and Chapter 15 of *Personal Communications Administrator Guide and Reference*.

Highlights:

- ▶ Hindi - National Language of India
- ▶ Hindi is also Phonetic Language
- ▶ As per IS 13194 : 1991 Hindi includes:
 - Consonants
 - Vowels
 - Matras
 - Vowel Modifiers
 - Halant & Nukta
 - Punctuation similar to English
 - 2 Avagrahs and 3 extra vowels in Sanskrit
 - Numerals

Hindi support is installed as follows:

- Select **India** for the language settings and make Hindi as Your Locale (from the Control Panel for Win2000)
- Make sure to select **1137** as the host code page while configuring the Personal Communications session
- Select Keyboard Layout of Personal Communications session: **Edit -> Preferences -> Keyboard -> India-Hindi**
- Select **Edit -> Preferences -> Appearance -> Font -> Devanagari MT Narrow**

23.2.12 Edit wrap pasted text

Personal Communications allows the pasting of copied text across fields and lines without breaking in the middle of a word, or ending a line with an invalid word. Cut/Copy and Trim options have also been added to the Edit menu. Personal Communications Version 5.5 offers an increased set of edit functions. The trim rectangle can be changed in size by using *size handles*. Most enhancements have been added to the Paste function. From the pull-down menu, click **Edit -> Preferences -> Edit** to set up the desired options as shown in Figure 23-43.

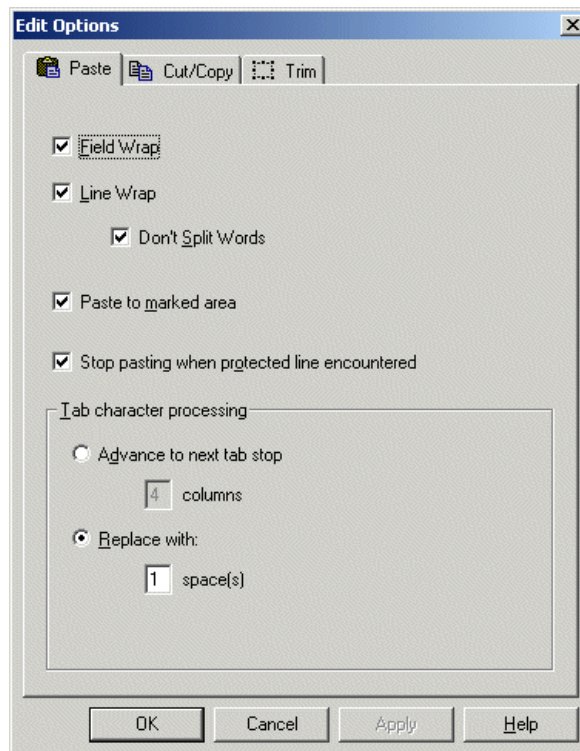


Figure 23-43 Paste option tab

You can control how text is pasted before and after protected fields, and how tabulated text appears after it is pasted.

► Field Wrap

Check **Field Wrap** if you want pasted data that falls onto a protected field to move to the next unprotected field. If you do not check this box, any data that falls onto an unprotected field is lost.

► Line Wrap

Check **Line Wrap** to allow pasting of copied text across lines.

► Don't Split Words

Check **Don't Split Words** to avoid words being split across fields and lines. The text being pasted into fields is split on word boundaries, which breaks the text and starts the new word in the next field. If one word is being pasted into a field, but the field is not long enough to hold the word, then as much of the word as possible is put into the field, and the rest of the word is carried on to the next field.

Note: The word break option is not available if the Field Wrap and Line Wrap options are both disabled.

- ▶ **Paste to marked area**
Check this box to restrict pasting to a marked area, if it exists. If the marked area does not exist, it will paste at the current location.
- ▶ **Stop pasting when protected line encountered**
Check this box to have the pasted text stop when it comes to a protected line on the emulator window. If you do not check this box, the paste continues.
- ▶ **Advance to next tab stop**
You can choose to align tabulated text at specified tab stops. For example, if you set the advance to 4 column(s), your tabulated text is advanced to the column position that is the next multiple of 4.
- ▶ **Replace with n space(s)**
You can choose to replace tab stops with a certain number of spaces. For example, if you set the replace to 3 spaces, each tab stop in your original text becomes 3 spaces.

The default setting is to replace each tab character with one space.
- ▶ **Paste data to fields**
This selection is available for 5250 sessions only.

You can choose to have tabulated text placed in subsequent unprotected fields. With this option, when a tab character is encountered, the following text data will be pasted into the next unprotected field of the emulator session.

23.2.13 Express Logon Feature

The Express Logon Feature allows a Personal Communications TN3270E user to log on to a host application without sending a user ID and password.

Traditionally, users are authenticated by host applications with a user ID and password. Administrators want to reduce the number of application-unique IDs and passwords to increase usability and security, and to reduce costs. The Express Logon Feature can assist in this objective by providing a mechanism to establish a secure TN3270 session and log the user onto a host application without the requirement of sending the host a user ID and password pair.

For complete details, see 25.3, “Express Logon Feature” on page 894

23.2.14 Smart card support

Personal Communications supports X.509 certificates in a dedicated security device.

Personal Communications has added support for smart cards, small electronic devices that contains electronic memory and may be used to store a single personal X.509 digital certificate. For complete details see 25.2, "Smart card support" on page 890

23.2.15 Scaling

Personal Communications uses BestFit to map the window size (display resolution) to the printed page size (printer resolution). BestFit scales the image so that it fills one printer page.

Personal Communications Version 5.5 has introduced the BestFit parameter to print graphics at the correct size at printers with 600 DPI and above. From the menu bar click **File -> Page Setup** and in the appearing window the **Graphics** tab.

The screen image is printed at the size of an output page. This is called BestFit (BF) and is the default. You can select a division factor to divide the size of the output:

BestFit is the default and best fits the output page.

- /2 divides the output page by one half.
- /3 divides the output page by one third.
- /4 divides the output page by one fourth.

If the output becomes too small, depending upon the printer, division selections /3 and/or /4 may only reduce the size the same as /2 division. BestFit will be recalculated each time **Print Graphics** is selected, in case printer properties change.

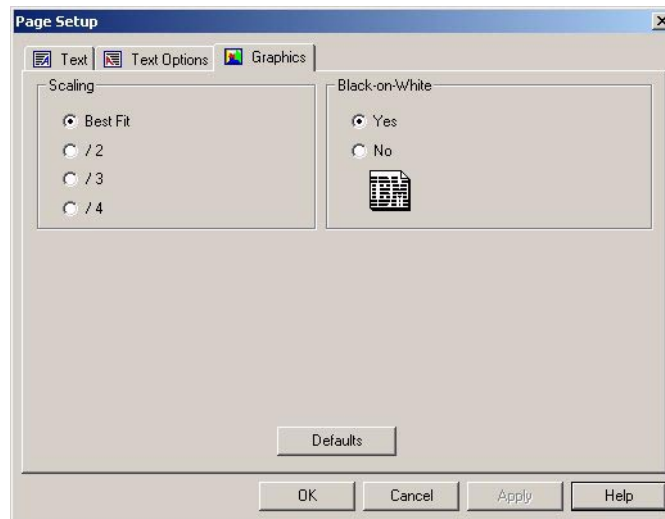


Figure 23-44 Scaling printing graphics

23.2.16 Higher DPI printer support

Personal Communications allows more scaling options so that graphic print jobs can be printed at the correct size on printers at 600 DPI and above. Personal Communications Version 5.5. changed the way graphics are printed. This new support incorporates the BestFit printing.

The old way was to take the graphic on the screen with the screen resolution and print it with the printer resolution. Usually the printer resolution was less and the printout of the graph was small. So the options while printing have been a multiplication of the print size.

Now the screen image is multiplied automatically with a factor determined by the printer resolution that best fits the page. The calculation is done every time before printing takes place so that the graphic will always fit on the page. The used DPI of the printer has no impact on the size of the graph when printed. That way, virtually any DPI can be used - including higher than 600 DPI. The user still can change the size of the printed output by scaling it down, as described in Figure 23-44 on page 865.

23.2.17 Capture view

This function allows you to add all open sessions with the click of a button to a batch file for later use. This will include the view (such as color settings) and the session short IDs. You can add up to 26 sessions in one view; you can save as many as eight views in a batch file.

23.2.18 NWSAA and ActiveX Support

Personal Communications Version 5.6 does not continue to offer the installation of support for NWSAA and ActiveX. This is due to the fact that the underlying services which are outside the scope of Personal Communications and IBM are no longer supported. However Personal Communications Version 5.6 still includes the executeables on the product CD on a “as is” basis.



Migration

Personal Communications Version 5.6 can be installed over all previous versions beginning with V4.3. During the installation process the older version will be detected and deleted before the installation of V5.6 begins. The uninstall process only uninstalls the product itself. Therefore, the user data and configuration files e.g. the \private subdirectory, will not be deleted.

In releases of Personal Communications prior to V5.5, all configuration files, macros and other client customization files were located in the \private subdirectory.

Personal Communications V5.5 allowed to store the users application data either in the classic \private subdirectory or in subdirectories as shown in Table 24-1 on page 868. When installing Personal Communications Version 5.6 it will detect that location of the data for the previous version. It will copy that data to the new Application Data Location as specified for the new installation of Personal Communications Version 5.6

24.1 Migration during installation

Personal Communications Version 5.6 allows you to customize the automatic migration process when updating from previous versions of Personal Communications. Migration is optional, but when selected, all profile references are updated to the current path for profiles selected at installation time that are moved during automatic migration. For more details on locations of application data of Personal Communications please refer to table in online documentation *Personal Communications for Windows, Version 5.6: CD-ROM Guide to Installation*, Chapter 3.

If the **User's Application Data Folder** is selected, the profile paths shown in Table 24-1 are used.

Table 24-1 Application data location, UserProfile

Operating System	User Class Directory (Current User)	System Class Directory
Windows 95/98/Me	C:\Windows\Application Data\IBM\Personal Communications	C:\Windows\All Users\Application Data\IBM\Personal Communications
Windows 95/98/Me (user profiles enabled)	C:\Windows\Profiles\%USERNAME%\Application Data\IBM\Personal Communications	C:\Windows\All Users\Application Data\IBM\Personal Communications
Windows NT 4.0	C:\Winnt\Profiles\%USERNAME%\Application Data\IBM\Personal Communications	C:\Winnt\Profiles\All Users\Application Data\IBM\Personal Communications
Windows 2000 / XP	C:\Documents and Settings\%USERNAME%\Application Data\IBM\Personal Communications	C:\Documents and Settings\All Users\Application Data\IBM\Personal Communications

If the **All Users Common Application Data Folder** is selected, the profile paths shown in Table 24-2 are used.

Table 24-2 Application data location, AllUsers

Operating System	User Class Directory (Current User)	System Class Directory
Windows 95/98/Me	C:\Windows\All Users\Application Data\IBM\Personal Communications	C:\Windows\All Users\Application Data\IBM\Personal Communications
Windows 95/98/Me (user profiles enabled)	C:\Windows\All Users\Application Data\IBM\Personal Communications	C:\Windows\All Users\Application Data\IBM\Personal Communications
Windows NT 4.0	C:\Winnt\Profiles\All Users\Application Data\IBM\Personal Communications	C:\Winnt\Profiles\All Users\Application Data\IBM\Personal Communications
Windows 2000 / XP	C:\Documents and Settings\All Users\Application Data\IBM\Personal Communications	C:\Documents and Settings\All Users\Application Data\IBM\Personal Communications

Note: In Windows 95, 98 and Me, you have the option of enabling user profiles. A user profile is an account maintained by the operating system that keeps track of a particular user's files and system configuration. When a user logs on to the system, Windows is loaded with the logged-on user's files and system configuration settings in place.

In Windows NT, 2000, and XP, user profiles are always enabled.

If **Classic Private Directory** is selected, the profile paths shown in Table 24-3 are used.

Table 24-3 Application data location, Private directory

Operating System	User Class Directory (Current User). Notes:1,2	System Class Directory
Windows 95/98/Me	C:\Program Files\IBM\Personal Communications\Private	C:\Program Files\IBM\Personal Communications\Private
Windows 95/98/Me (user profiles enabled)	C:\Program Files\IBM\Personal Communications\Private	C:\Program Files\IBM\Personal Communications\Private
Windows NT 4.0	C:\Program Files\IBM\Personal Communications\Private	C:\Program Files\IBM\Personal Communications\Private
Windows 2000 / XP	C:\ Program Files\IBM\Personal Communications\Private	C:\ Program Files\IBM\Personal Communications\Private
<p>Note1: If the User Preference Manager (UPM) was set to a directory other than the default directory, Personal Communications will utilize that directory to store the user-class files. System-class files are always stored in the Private directory.</p> <p>Note2: For the classic Private directory locations, C:\Program Files\IBM\ Personal Communications is the drive where Personal Communications is installed</p>		

If the user is doing a custom installation, the window shown in Figure 24-1 will be provided to change the level of migration performed, ranging from no migration to full migration. A typical installation will result in a full migration.

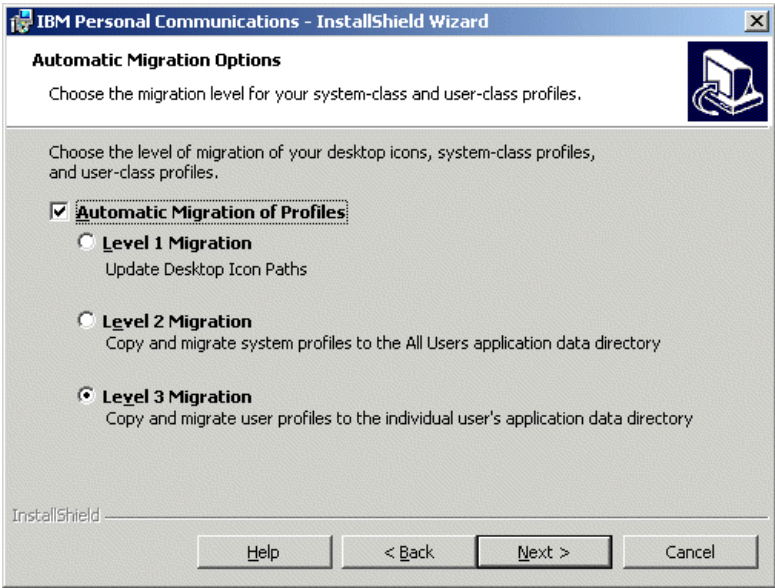


Figure 24-1 Migration levels

The levels of migration vary based on the location of the application data that the installer has chosen. Refer to [“Application Data Location” on page 570](#).

- ▶ Level 1 Migration
Only desktop icons are migrated.
- ▶ Level 2 Migration
This is considered system-level migration. It includes desktop icons and adds system-class profiles. Table 24-4 describes the system-class files that will be migrated.

Table 24-4 System-class profile file extensions

File extension	File type
.acg	SNA configuration
.mlg	Default message log
.trc	Unformatted trace
.tlg	Formatted trace

Level 2 also migrates user-class profiles when migrating profiles to either All Users Common Application Data Folder or the Classic Private Directory. For a list of user-class profile files, see Table 24-5.

Table 24-5 User-class profile file extensions

File extension	File type
.ws	Workstation profile
.bch	Multiple sessions
.ini	Session size and location
.pmp	Popup-keypad configuration
.kmp	Keyboard configuration
.srl	File transfer list
.ndc	AS/400 connection configuration
.upr	AS/400 user profile
.tto	AS/400 data transfer request (receive)
.tfr	AS/400 data transfer request (send)
.bar	Toolbar setup
.mac	Macro
.mmp	Mouse setup
.xlt	Translation table
.xld	DBCS translation table
.cert	Certificate
.sth	Password stack
.adu	Automatic dial utility
.kbd	Certificate Management database
.der	Binary DER

► Level 3 Migration:

This is considered a full migration. It includes desktop icons, system-class profiles (Table 24-4) and user-class profiles (Table 24-5).

It is recommended that an individual user accept the automatic migration option selected, and that only Personal Communications Version 5.6 administrators use the different levels of migration.

In previous releases of Personal Communications, all of the user data migration occurred during the reboot of the machine after the install was run. In this release the system-class profiles are moved and migrated during the first reboot after Personal Communications Version 5.6 is installed. The desktop icons and user-class profiles are moved and migrated the first time a user logs on after Personal Communications Version 5.6 is installed.

A log is created during the migrations. This log file is named pcsmig.log and is located in the System Level profile directory. This log file will contain a history of what was migrated and what files were moved, including their original and new locations. This file is essential in finding any problems that occurred during the migration. If a user has migration problems, or any problems with his profiles after Personal Communications Version 5.6 was upgraded, this pcsmig.log file may be required for problem determination. The pcsmig.log file is also written during a manual migration.

24.2 Migration Utility

This migration process can be run manually after Personal Communications Version 5.6 has been installed. The Migration Utility is used to copy the configuration files for Personal Communications V5.6 by reading the information stored in configuration files from previous versions of Personal Communications. This process allows you to avoid having to reenter all of your configuration data when you upgrade to Personal Communications V5.6.

Start the Migration Utility by clicking **Start -> Programs -> IBM Personal Communications -> Administration and PD -> Migration Utility**. The resulting window is shown in Figure 24-2.

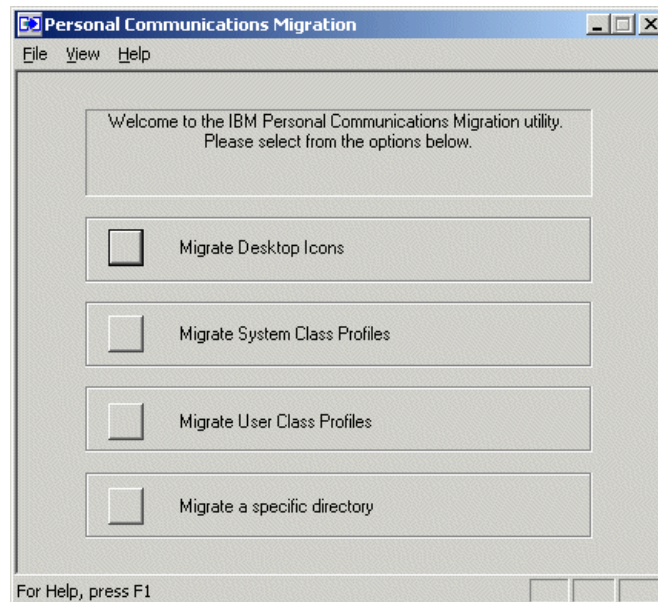


Figure 24-2 Migration Utility

If you do not see your session profiles in the session manager window after migration your old session definitions etc. might be in a subdirectory which the new Personal Communications has not recognized. In that case click the button **Migrate a specific directory**. You get a panel to choose the subdirectory from:

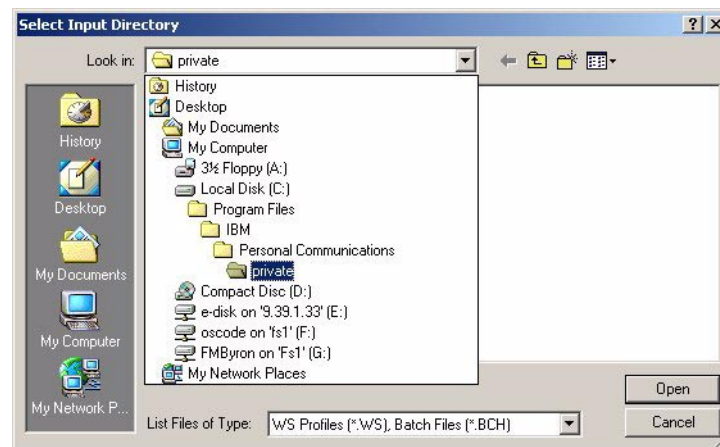


Figure 24-3 Select specific directory for migration

In previous releases of Personal Communications there have been small changes in the workstation profiles (.ws) from one release to another. Personal Communications just executes, expecting the profile contains the keywords as recognized by this latest version of Personal Communications.

For the future, Personal Communications has added a version number to the Personal Communications profile to allow it to differentiate among different versions of the profiles. The keyword is in the [Profile] section and is named Version. If this keyword does not exist in a profile, the profile is from Personal Communications 5.0 or earlier.

All profiles from Personal Communications V5.5 have a version number of 5. The profile version number does not correlate to the Personal Communications version number.

Example 24-1 Version number in ws file

```
[Profile]
ID=WS
Version=5
[Telnet3270]
HostName=9.39.1.37
HostPortNumber=23
Security=N
[Communication]
Link=telnet3270
[3270]
QueryReplyMode=Auto
HostCodePage=1047-U
```

If a user starts a Personal Communications profile from a previous release of Personal Communications, the user will be prompted to migrate the profile to the current version:

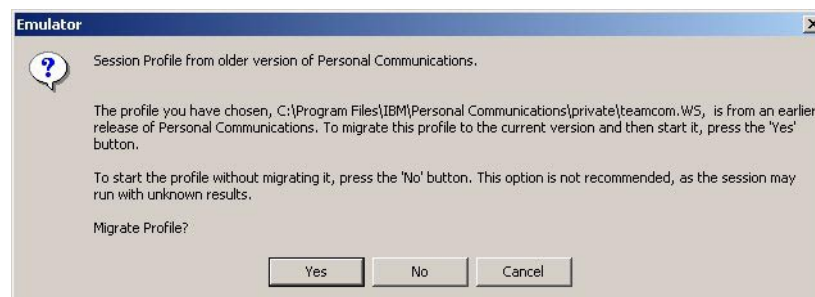


Figure 24-4 Prompt for migrating down level profile

Whenever any profile gets modified, its version number will be incremented by one.

Example 24-2 Version number in migrated ws file

```
[Profile]
ID=WS
Version=6
[Te1net3270
....
```

There is also an option letting the user run the profile as is; however, it is not recommended to run the profile as is.

If a user starts a profile created in a newer version of Personal Communications than he has installed on his machine Personal Communications will display a pop-up window telling the user that the profile is from a new release and that he must upgrade his version of Personal Communications if he wants to run the profile.



Security

Personal Communications Version 5.6 supports the industry-standard Secure Sockets Layer (SSL) protocol to insure privacy of data transmission. For an overview of the operation of SSL, refer to “Secure Sockets Layer” on page 1018.

25.1 Enhancements in certificate management

Personal Communications Version 5.6 introduces Gskit6 (Global Security Tool Kit) for certificate management.

You can use it via the command line interface Ikeyman, the certificate wizard or the certificate management.

To start the certificate management click **Start - Programs - Personal Communications, Utilities - Certificate Management**. Figure 25-1 on page 878 shows the version of the key management which appears when you click at **Help - About**.

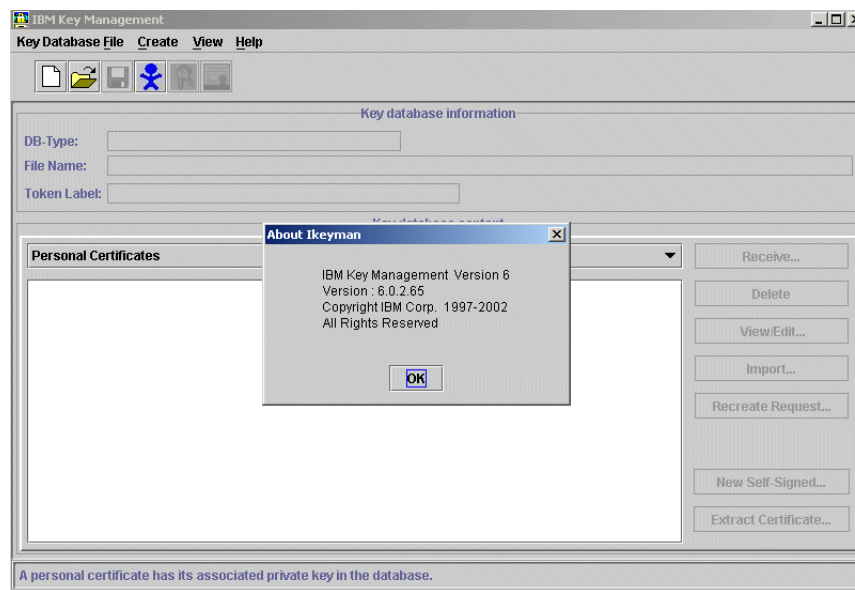


Figure 25-1 Certificate management with key management version

The enhancements over previous versions as contained in Personal Communications and HOD are:

- Wider range of formats for certificate file as shown in Figure 25-2 on page 879 (Cryptographic Token is a device or smart card holding the certificate).

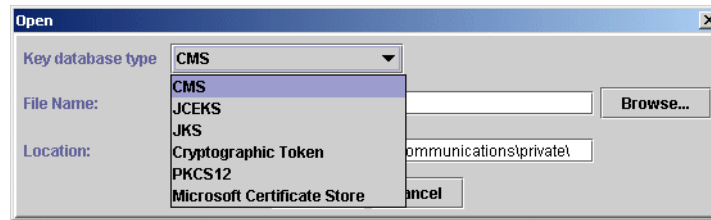


Figure 25-2 Formats of certificate files

- Extend supported types of key data bases

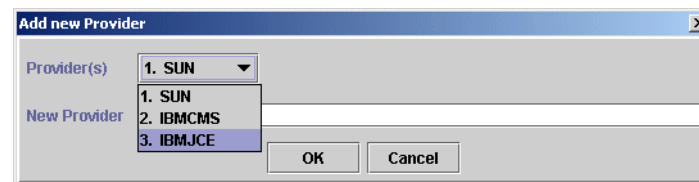


Figure 25-3 Selection of providers for key database file types

25.1.1 Example of certificate management

The use of IKEYMANN is well described in the Online documentation. Experience shows that for Personal Communication the GUI versions (Wizard and Management) are preferred. So we will use the GUI of the Certificate Management for our example. To show a general use of the Certificate Management we will setup a scenario where we have the HOD redirector being used as proxy for a Personal Communications Version 5.6 secure telnet session.

The procedure to install a certificate would be the same as far as Personal Communication and its Certificate Management concerns if the certificate was created at another source (for example, at a zSeries Telnet server).

Also the procedure of creating a self signed certificate using HOD Certificate Management applies as well to create a self signed certificate using the Gskit6 Certificate Management of Personal Communications Version 5.6.

Here are the steps for our example:

- Use HOD Certificate Management to issue a self signed certificate which is added into HOD key data base.
- Extract certificate from HOD key data base as ARM file for use in Personal Communications Version 5.6

- ▶ Setup Redirector of HOD with client side security and restart it to pick up the configuration change and the certificate.
- ▶ Import certificate into key data base of Personal Communications Version 5.6
- ▶ Setup a 3270 Telnet session with Personal Communications Version 5.6 to use SSL with certificate

Creating a certificate using HOD Certificate Management.

We use the HOD certificate management to create a new self signed certificate as follows:

Open HOD Key Management and select the **Personal Certificates** from the drop down box. You will see Figure 25-4 on page 880. You will notice that the Key Management of HOD has a slightly different view than that of Personal Communications Version 5.6. This is because HOD is using Gskit Version 5 and Personal Communications Version 5.6 is using Gskit 6.

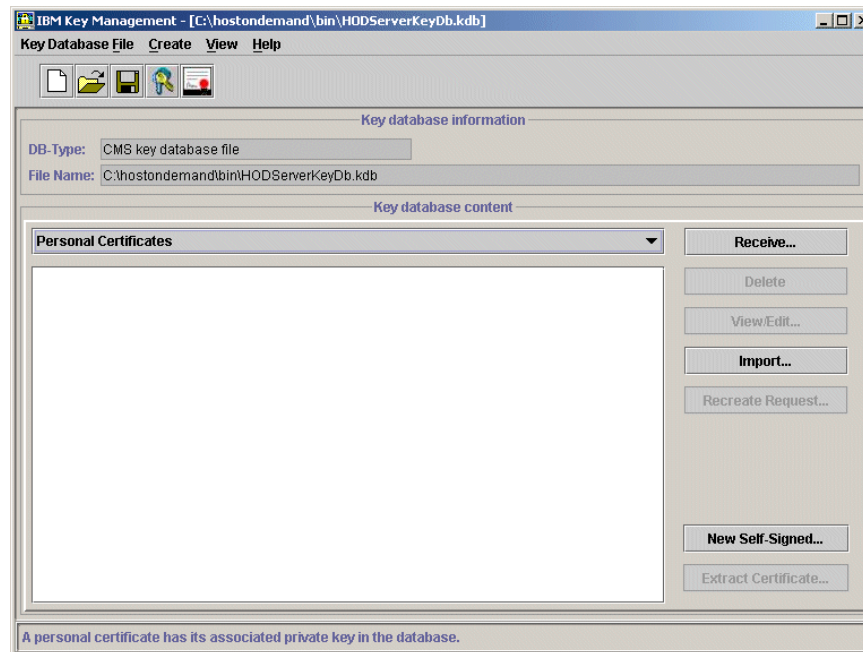


Figure 25-4 Key Management of HOD 7

Click **New Self Signed**. You will get the window as shown in Figure 25-5 on page 881.

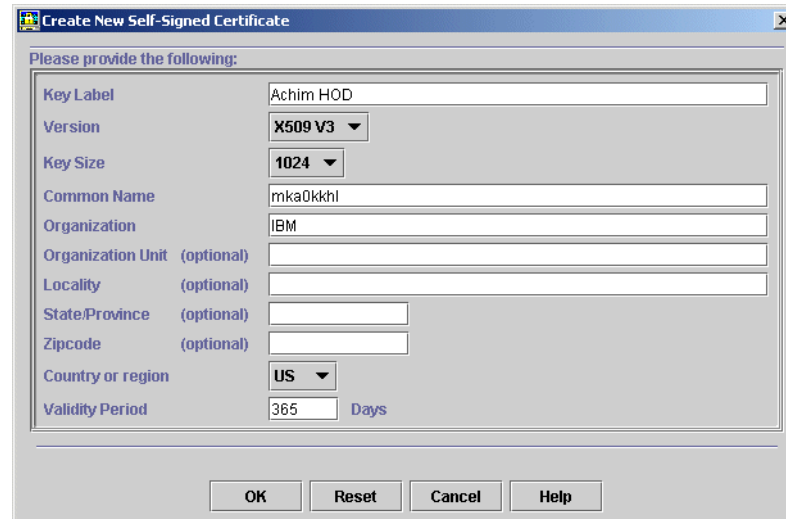


Figure 25-5 Create new self signed certificate

Fill in a key label and the organization. The rest of the field have been left to their default values. This includes the Common Name which is the same as found in My Computer - Properties - Network Identification - Full Computer Name.

After clicking **OK** this certificate is added to your Personal Certificates.

Extracting a Certificate

We now need to extract the certificate which is currently contained in our key data base file. For Personal Communications Version 5.6 we need to extract it as Base 64 encoded ASCII file. To create that file we click **Extract** and receive the panel as shown in Figure 25-6 on page 881. Using **Browse** we select the subdirectory to save the certificate.

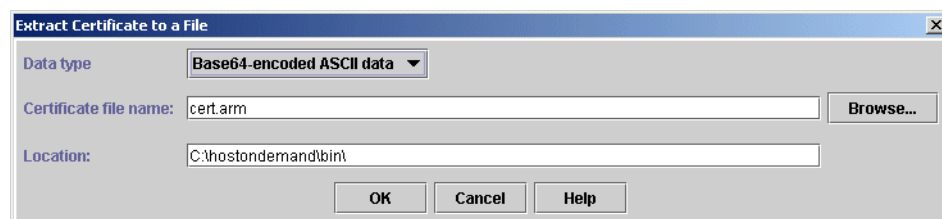


Figure 25-6 Extracting certitude from key data base

Click **OK** and we are done, extracting a self signed certificate from our HOD key data base so that it can be used by Personal Communications Version 5.6. We end the HOD key data base management utility.

Setting up the HOD redirector for a secure session with telnet client

See Chapter 7.3.1, “Configuring the Redirector” on page 351 for the general procedure how to setup the redirector. For our example here we use a client side security:

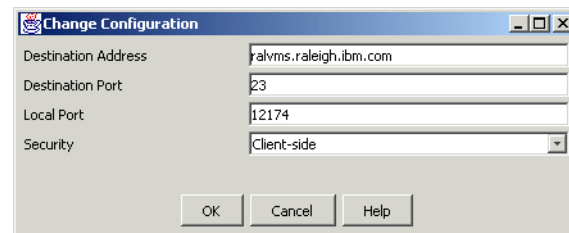


Figure 25-7 Setting up redirector for client side security

Note: Be sure to stop and restart the redirector so it picks up the new definitions.

You need to restart it also when changing certificates!

Setting up Personal Communications Version 5.6 for secure session

We use the Key data base management of Personal Communications Version 5.6 to implement the HOD certificate into the key data base of Personal Communication. Click **Start - Programs - IBM Personal Communications - Utilities - Certificate Management**.

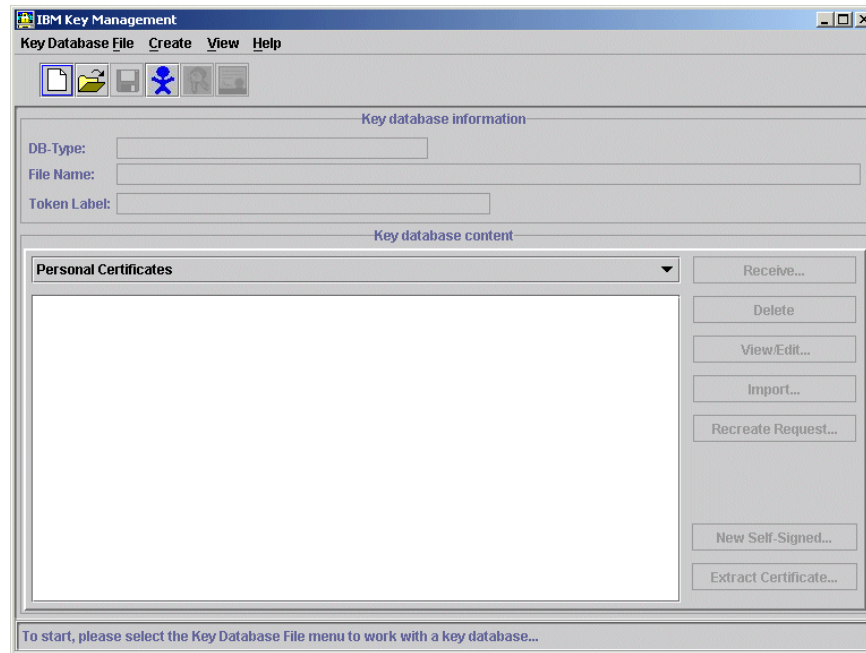


Figure 25-8 Key Management of Personal Communications Version 5.6

In Key Management of Personal Communications Version 5.6 as shown in Figure 25-8 on page 883, we click **Key Data Base File - New** to create a new key ring data base file. The type has to be CMS as shown in example of Figure 25-9 on page 883.

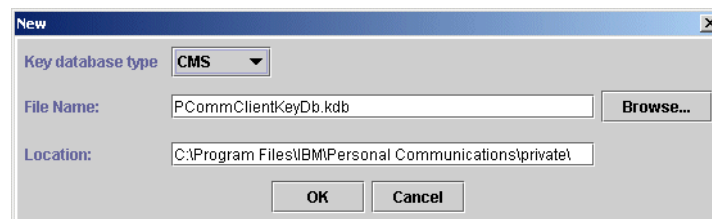


Figure 25-9 Opening key data base file and select type

We use the default password pcomm and select Stash password to a file as shown in Figure 25-10 on page 884.

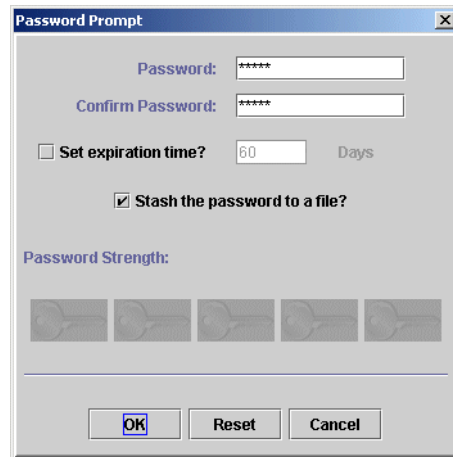


Figure 25-10 Password prompt and stash password to file

We are now seeing the Signer Certificates as shown in Figure 25-11 on page 884.

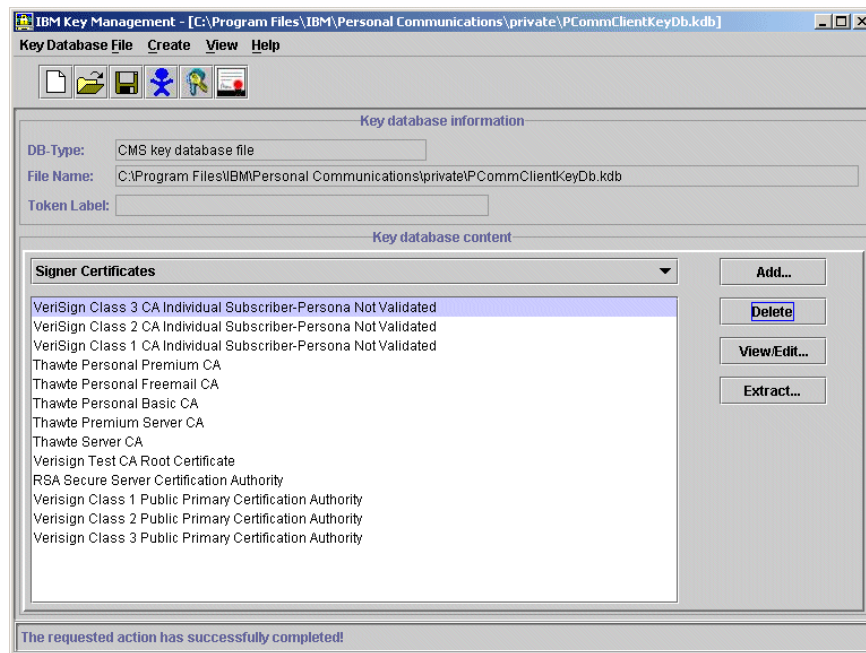


Figure 25-11 Signers Certifications pane in Gskit6 Window

We now **Add** to add the certificate which we had created before at the HOD server. At HOD we had extracted the ARM file, Base64 encoded ASCII, which we need now for Personal Communications Version 5.6. Use the browse button to point to the location of that file. In the example we had for taking our screen copies HOD and Personal Communications Version 5.6 on the same machine - so we find the certificate on the default HOD subdirectory as shown in Figure 25-12 on page 885.

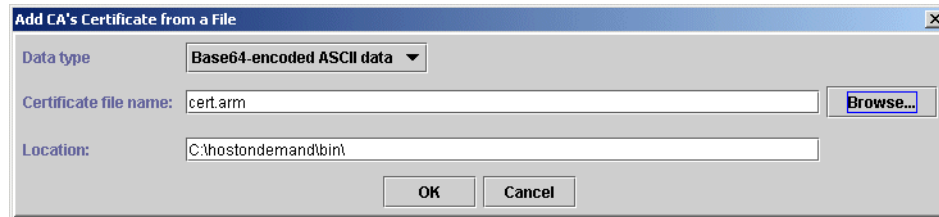


Figure 25-12 Adding a certificate for Personal Communications

After clicking **OK** we are prompted to enter a label for the new certificate. Choose any which you can remember. Click **OK** and the new certificate is included in the list of Signers Certificates.

Now we can close the certificate management. We open the session manger of Personal Communications Version 5.6 by clicking **Start - Programs - Personal Communications - Start or configure Sessions**. We click **New** and get the session configuration window of Personal Communications Version 5.6.

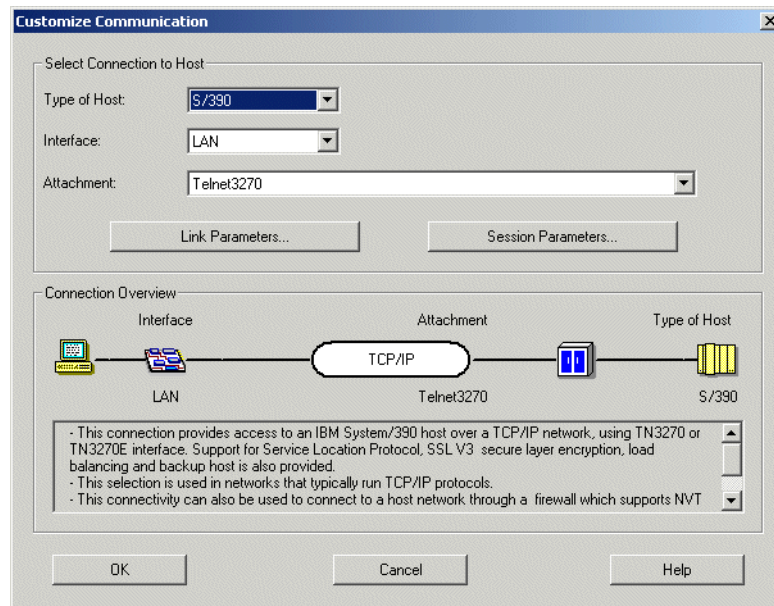


Figure 25-13 Telnet configuration

The window defaults to a Telnet configuration. We click at **Link Parameters** and get the window to enter the IP address and port and to select as well the secure check box as shown in Figure 25-14 on page 887. Note that we have entered the IP address and port not from the telnet server but from our HOD redirector.

Telnet3270

Host Definition

Automatic Host Location

Advanced Security Setup

	Host Name or IP Address	LU or Pool Name	Port Number
Primary	9.24.104.186		12174
Backup 1			23
Backup 2			23

Printer Association (only valid for TN3270E Display sessions)

Associated Printer Session

Browse...

☒ Start Associated Printer Minimized

☒ Automatically close the associated printer session with this session

☐ Auto-reconnect

☒ Enable Security

OK

Cancel

Apply

Help

Figure 25-14 Parameters for Telnet session

This is the only window in which we made entries. As reference we show in Figure 25-15 on page 888 the tab Advanced Security Setup with its unchanged default values.

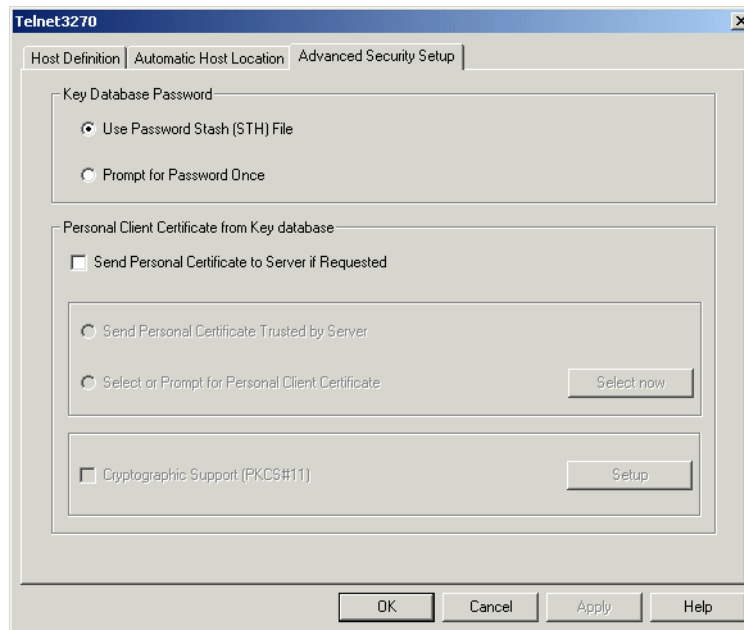


Figure 25-15 Advanced tab of secure tn3270 session

Clicking **OK** through the panels back out we will get the resulting TN3270 session from Personal Communications Version 5.6 via our HOD redirector to the mainframe in which the telnet server runs.

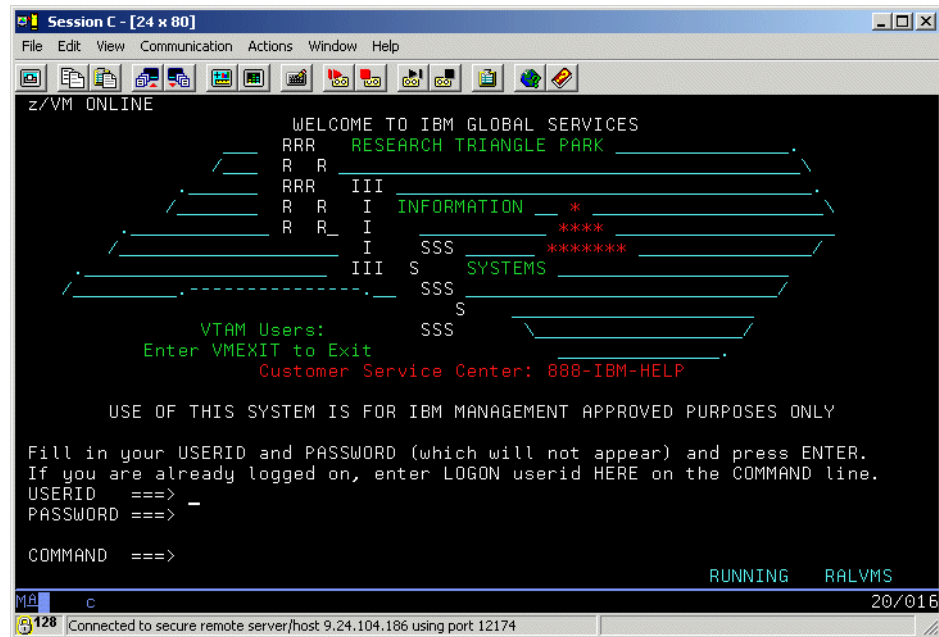


Figure 25-16 Secure TN3270 session with PCOMM

In the status bar we see the address and 5 digit port number of the redirector. The TN server address is not known by the emulator and is not displayed. However we see at the lower left corner a closed lock with a number. This shows us that we have a secure session with 128 bit encryption.

If your session fails to connect: Did you stop and restart your redirector after the certificate was created at the HOD?

25.2 Smart card support

A *smart card* is a small electronic device that contains electronic memory and may be used to store a single personal X.509 digital certificate. It does not hold the signer certificates. The signer certificate or the root and any intermediate certificate of the personal certificate on the smart card should be added in the PCommClientKeyDb.kdb file.

Some advantages of using a smart card are:

- ▶ Support for Netscape PKCS#11 cryptographic devices
- ▶ Allows you to store the Personal Certificates on the cryptographic devices. (for example, smart card, IBM embedded chips, and others)
- ▶ Provides additional security since the certificate is stored on an external physical device.
- ▶ It takes two things to make a client-authenticated connection: something you have (the smart card) and something you know (the PIN password). This is considered safer than storing the certificate in the key database, PCommClientKeyDb.kdb file, on disk.

25.2.1 Enabling smart card support

To enable smart card support when configuring a session, perform the following steps:

1. Select **Enable Security** in the Host Definitions tab, then select the **Advanced Security Setup** tab. See Figure 25-17.
2. On the Advance Security Setup tab, select **Send Personal Certificate to Server if Requested**.

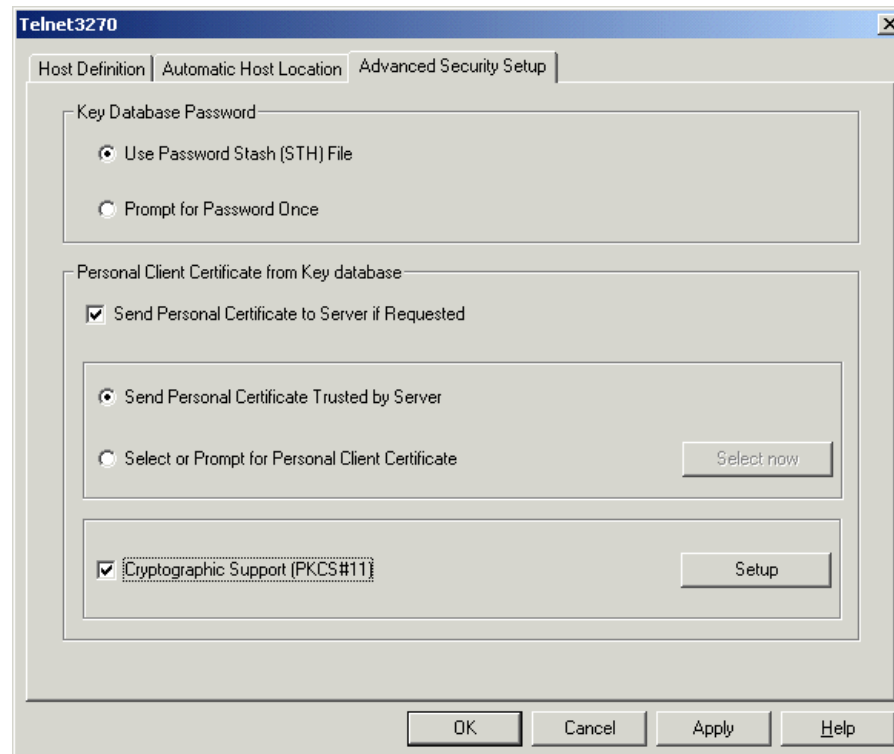


Figure 25-17 Enable Security

3. To obtain the certificate from the smart card, select **Cryptographic Support (PKCS#11)**, then click **Setup** to display the window shown in Figure 25-18.

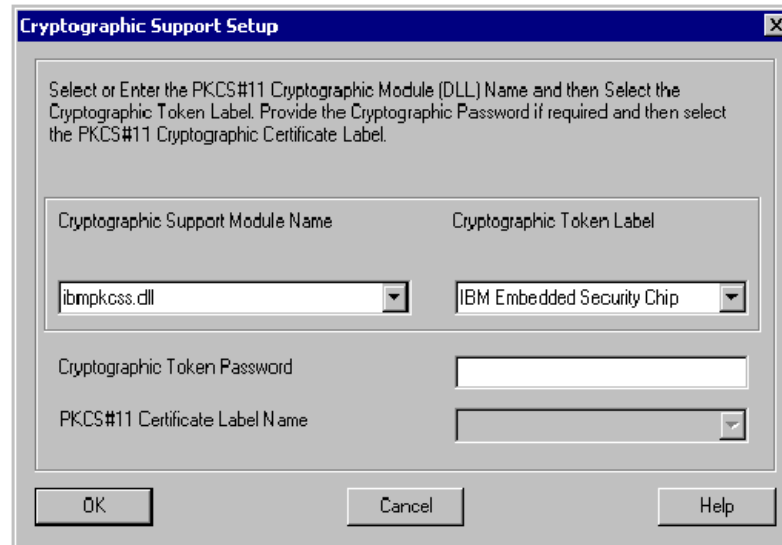


Figure 25-18 Smart card setup in Personal Communications (continued)

4. In the Cryptographic Support Setup window, choose the Netscape compatible PKCS#11 driver name from the drop-down list. If your provider's driver is not found by Personal Communications Version 5.6, then you must manually enter a smart card driver name. Table 25-1 is a list of supported smart card drivers and their file names.

Table 25-1 Supported smart card drivers

Smart card Drivers	File names
IBM SecureWay Smartcard	w32pk2ig.dll
GemPlus/GemSoft Smartcard	w32pk2ig.dll
IBM Netfinity PSG Chip ¹	ibmpkcss.dll
Rainbow Ikey 1000	Cryptoki22.dll
Schlumberger Cryptoflex	acpkcs.dll or slbck.dll
SCW PKCS 3GI 3-G International	3gp11csp.dll
Data Key	Dkck232.dll
Fortezza Module	fort32.dll
¹ The system boards in some IBM systems are preconfigured with a Promise of Value (POV) card, a 256-bit encrypted security chip daughtercard attached to the motherboard. If an attempt is made to remove the POV card from the board and install it into another system, the cryptographic key material will be erased, rendering it unusable. This security feature is by design and prevents the cryptographic key migration from one system to another. Therefore, moving this security chip from one board to another is not a supported option. If an attempt is made to move the POV card from one system to another, it may hang on boot and display an error message referring to an invalid machine type and serial number.	

5. If the driver loaded successfully, the Cryptographic Token Label list is displayed. Now enter the Cryptographic Token Password (PIN). This provides access to the PKCS#11 cryptographic device and displays the PKCS#11 Certificate Label Name. If PKCS#11 cryptographic support is enabled and a password for the PKCS#11 cryptographic module is not defined during configuration, the user will be prompted to provide the password.

When you install support for a smart card, drivers are installed that allow you to store and retrieve your certificate from your smart card. Drivers are provided that allow both Netscape and Internet Explorer to access the certificate in the smart card. If your certificate was not preinstalled in the smart card for you by your administrator, you may use your browser to insert your certificate directly into the card.

If you have your certificate in a P12 file, you can use the Certificate Management Utility to store the certificate on the PKCS#11 device if the ikmuser.properties file is updated with the right PKCS#11 module name.

Please refer to Chapter 8 of the *Personal Communications Version 5.6 Administrator's Guide and Reference* for further details.

25.3 Express Logon Feature

Personal Communications has been enhanced to support Express Logon. For a discussion of the concepts and operations of Express Logon, refer to 11.8, “Express Logon Feature” on page 455. The remainder of this section will discuss the Personal Communication-specific implementations and operations regarding the Express Logon Feature.

25.3.1 Client setup

The Express Logon Feature requires an SSL session with client authentication; therefore, you first you must have Personal Communications configured for SSL and client authentication. Once the client-authenticated session is established, you may record the ELF macro.

Recording the ELF macro

From the menu bar of the emulator session, click **Actions -> Start Recording a Macro**. This results in the window shown in Figure 25-19 on page 895. In this window you must select the radio button **Macro File**, since ELF macros can only be recorded in the native macro language of Personal Communications Version 5.6. Next, select **Enable** in the Express Logon for Macro field. In the Application ID field, fill in the Host Access Application ID that RACF uses to identify the desired application. This is *not* the VTAM APPLID of your host application. The Host Access Application ID is an ID that matches the RACF PTKTDATA profile configured at your OS/390 or z/OS host. Your RACF administrator will provide you with this ID, if they did not record and distribute the macro. Enter the name of the macro in the File Name field, and you may optionally provide a description of the macro in the Description field. Click **OK** to start the macro recording. In the OIA you will see the capital letter “R”, indicating that the recording process is taking place.

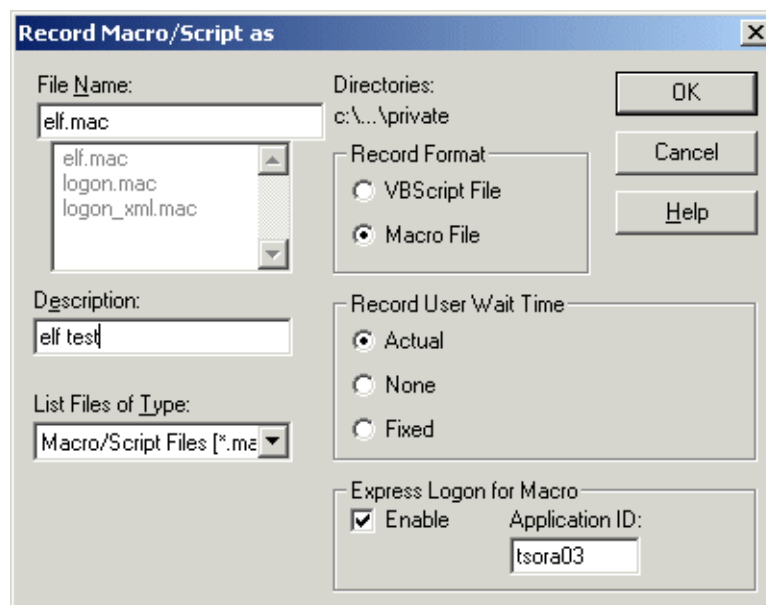


Figure 25-19 Record Macro window for ELF

With the macro recording, return to the Personal Communications session and log on to your host application by typing in your host application name (logon applid(application_name)), your user ID and your password. Note that in the logon statement the application_name entered *will be* the VTAM APPLID. The windows you see might vary, depending on your host systems. Enter your user ID and password when prompted.

Example 25-1 shows the results of the macro created for this chapter.

Example 25-1 Express Logon Feature macro

```

Description =elf test
[wait sys]
"logon applid(ra03t)
[enter]
[wait inp inh]
wait 10 sec until FieldAttribute 0000 at (1,26)
wait 10 sec until cursor at (2,1)
[wait app]
elf applid tsora03
")USR.ID(
[enter]
[wait inp inh]
wait 10 sec until FieldAttribute 000C at (8,19)
wait 10 sec until cursor at (8,20)

```

```
[wait app]
")PSS.WD(
[enter]
```

Notice the elf line after [wait app] and just prior to the user ID placeholder)USR.ID(. This line:

- ▶ Identifies the macro as being ELF enabled
- ▶ Saves the host access application ID entered on the record window in the application ID field

The real host user ID and password are replaced in the macro with special placeholder strings:

- ▶ The ELF user ID placeholder is)USR.ID(
- ▶ The ELF password placeholder is)PSS.WD(

To make the macro an autologon macro, click **Edit -> Preferences -> Macro/Script**, select your ELF macro from the Macro/Script drop-down list as shown in Figure 25-20, and click **OK**.

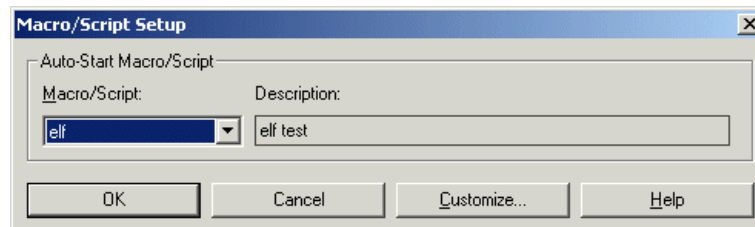


Figure 25-20 Create an autologon macro

The next time you click that session in the Session Manager, it will open the session window, connect and run the ELF macro automatically.



Programming interfaces

Personal Communications Version 5.6 has several programming interfaces that may be used to extend functionality or more commonly to automate operations. The following APIs (application program interfaces) are available for use with Personal Communications Version 5.6:

- ▶ Protocol stack interfaces

- SNA API

- For details see the online documentation *Client/Server Communications Programming*, pccsp.pdf.

- ▶ Emulator interfaces

- Macros

- EHLLAPI

- For details, see the online documentation *Emulator Programming*, pcep.pdf.

- HACL

- For details, see the online documentation *Client/Server Communications Programming*, pccsp.pdf, Part 5 and the online documentation *Host Access Class Library*, pcecl.pdf, and as well the online documentation in the CD subdirectory \publications/en_US/doc/hacl/.

- OLE

26.1 Macros

This chapter focuses only on the following:

- ▶ Converting macros to XML
- ▶ Import macros into Host On-Demand
- ▶ Hiding logon passwords

26.1.1 Converting macros to XML

Personal Communications Version 5.6 records keyboard macros either in its native language or in VBSCRIPT. However, the Convert Macro utility will not convert VBSCRIPT macros; it will only convert macros recorded and stored in the native Personal Communications language (Record Format = Macro File).

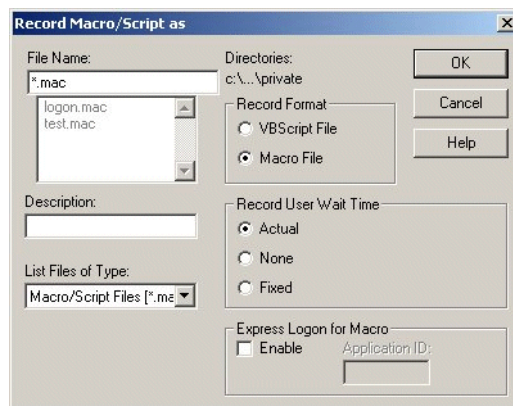


Figure 26-1 Recording a macro as native macro file

Personal Communications Version 5.6 provides a utility that is intended for use by customers who are migrating from Personal Communications to Host On-Demand. This utility will convert Personal Communications macros into an XML format that may be imported into Host On-Demand. Note that there is no utility available to migrate from XML or VBscript to the Personal Communications macro file format.

To convert a macro to XML, click **Start -> Programs -> IBM Personal Communications -> Utilities -> Convert Macro**.

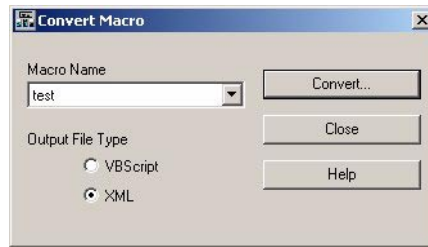


Figure 26-2 Convert Macro utility

If you cannot find your recorded macro you possibly did not record it in the native macro file.

Attention: Always check output for comments (unconverted statements). See the example in 26.1.3, "Hiding logon passwords" on page 902.

See as well Chapter 23.1.11, "Enhanced macro conversion utility" on page 825 for changes in saving converted macros.

26.1.2 Import macros into Host On-Demand

A macro, converted to XML using the Convert Macro utility, can be imported into Host On-Demand as shown in the following example. It is not required to change the extension from .mac to .xml to import the macro to Host On-Demand.

To import the macro into Host On-Demand, you must enable your Host On-Demand session so that you show the toolbar with the Macro Manager portion enabled as shown in Figure 26-3. This may be done by clicking **View -> Macro Manager** from the menu bar.

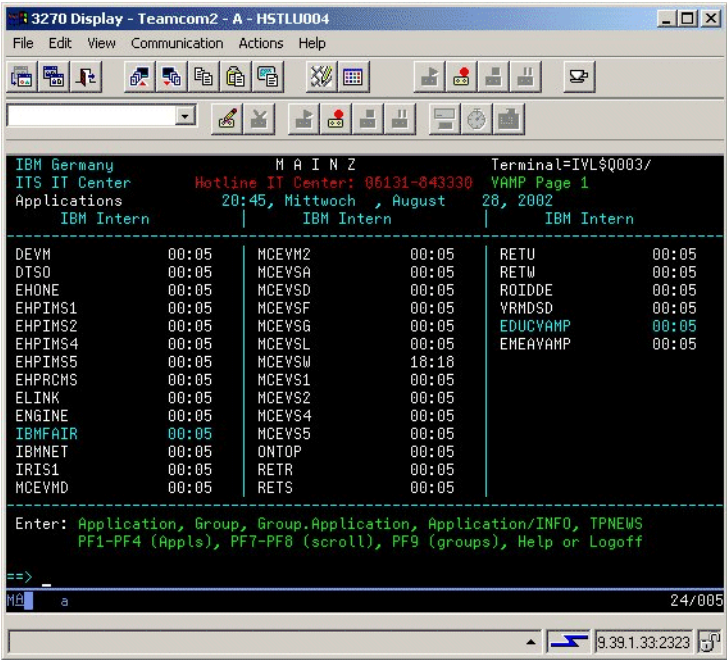


Figure 26-3 Host On-Demand session with toolbar for managing macros

Next, click the edit icon from the Macro Manager toolbar to open the window for importing the XML macro from Personal Communications Version 5.6, see Figure 26-4. Click **Import** to display the window in which you may select the files available for import. Navigate if necessary to the location containing the macro and select it. When the macro is read the fields in the window are completed using the information contained in the headers of the imported macro. You can update the fields manually by overtyping.

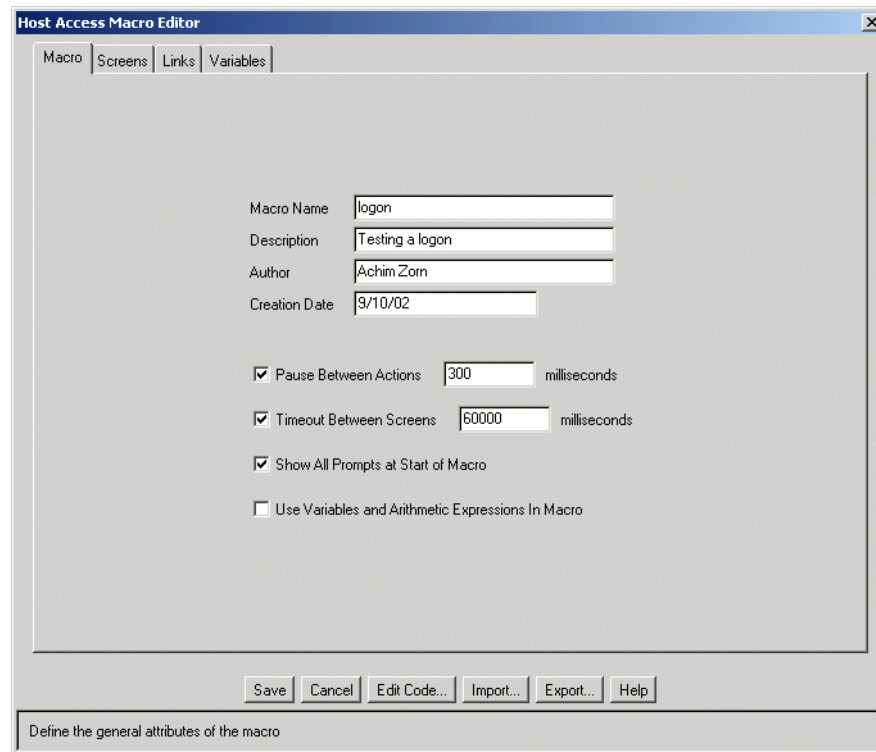


Figure 26-4 Fill in the fields and use Import

After the file is imported you can change the macro if desired, then click **Save** to store the macro with the session properties.

ELF macros can also be imported from Personal Communications Version 5.6 to Host On-Demand and used by Host On-Demand.

The user should look at the converted file to make sure all functions were converted correctly. If there is an unsupported macro function, it is put into the output file as a “xml comment” line.

26.1.3 Hiding logon passwords

One of the most popular uses for macros is a logon macro. By default, macro recording will capture and store passwords in the clear even though they are not displayed on the host screen. The contents of the host screen contains the password but the display attribute of that field is *Hidden* so that the data cannot be seen. Administrators can modify the client workstation profile (*.WS) in the [keyboard] stanza with the following option:

```
HideNonDisplayDataOnRecord=Y
```

With this in the *.ws file, the macro recording process will place [input nd] into the macro instead of the actual password as shown in Example 26-1. When the macro is replayed in Personal Communications Version 5.6, a Windows prompt for the user's password will be issued, and the password will not be displayed while typing.

Example 26-1 Workstation profile

```
[Profile]
```

```
ID=WS
```

```
Version=6
```

```
[Telnet3270]
```

```
HostName=9.39.1.11
```

```
Security=N
```

```
AutoReconnect=Y
```

```
HostPortNumber=23
```

```
[Communication]
```

```
Link=telnet3270
```

```
[3270]
```

```
QueryReplyMode=Auto
```

```
HostCodePage=037-U
```

```
[Keyboard]
```

```
HideNonDisplayDataOnRecord=Y
```

```
CuaKeyboard=1
```

```
Language=United-States
```

```
DefaultKeyboard=$$BLANK$$
```

Example for the resulting native macro as recorded:

```
Description =
```

```
[wait app]
```

```
"devm zorn
```

```
[enter]
```

```
[wait inp inh]
```

```
wait 10 sec until FieldAttribute 000C at (22,80)
```

```
wait 10 sec until cursor at (23,1)
```

```
[wait app]
```

[input nd]
[enter]

A translation of that macro into VBSCRIPT will also contain the equivalent command for [input nd].

When converting this macro into XML for Host On-Demand the [input nd] line is not recognized; therefore, it will be commented within the XML macro during translation

```
</comment> ***** The following line is not translatable to HOD/XML---  
[input nd]
```

In such cases where there is no equivalent for a Personal Communications macro command in XML; you must build your own work-around with the available set of XML commands. Available XML macro commands and the descriptions are located in the Host Access Toolkit online documentation *Host Access Beans for Java Reference*. Specifically, for entering a password the following command should be inserted at the point where the password should be entered:

```
<prompt name="password" description="" row="29" col="17" len="8" default=""  
clearfield="true" encrypted="true" movecursor="false" xlatehostkeys="false"  
>
```

Where:

row	The row to place the prompt. The value must be a number. This is a required element.
col	The column to place the prompt. The value must be a number. This is a required element.
len	The length of the prompt. The value must be a number. This is a required element.
name	The name of the prompt. This can be any valid unicode character. This element is optional.
description	The description of the prompt. This can be any valid unicode character. This element is optional.
default	The prompt's default value. This can be any valid unicode character. This element is optional.
clearfield	This clears the host field on placement of prompt text. The value must be true or false. This element is optional. The default is false.
encrypted	This element is optional, and must be either true or false. If the value is true a password echo character will be displayed (*) as the password is typed. The default is false.

`xlatehostkeys` If true, host key mnemonics (for example, [enter]) will be translated. For a list of key mnemonics, see Appendix A, “SendKeys Mnemonic Keywords” in the Host Access Class Library document. The value must be true or false. This attribute is optional. The default is false. If you do not have this value set to true, which is normal because you wouldn't ask users to type key mnemonics, don't forget to code an input element after the prompt(s) for the current actions to get the prompt data entered onto the host.

26.2 EHLLAPI

HLLAPI (High Level Application Program Interface) and EHLLAPI (Enhanced HLLAPI) are standard program interfaces to many emulator types. They are simple to use and can be used by various programming languages. There are no new functions in Personal Communications Version 5.6, but we wanted to add some hints in this book because all other interfaces to the presentation space are all bridged via EHLLAPI.

Personal Communication provides a tool (`vbhllapi.exe`) for testing single EHLLAPI commands. It is located at the CD of the product in the subdirectory

`\install\pcomm\program files\Ibm\Personal\Communications\samples\vbhllapi`.

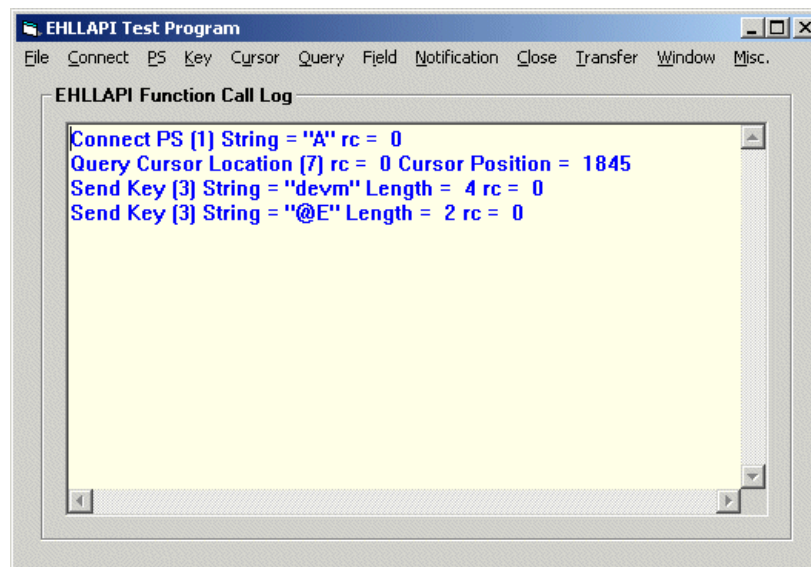


Figure 26-5 EHLLAPI test program

The following sequence was tested with the EHLLAPI test tool and recorded in Figure 26-5. For details on EHLLAPI commands and their return codes, refer to Chapter 3 of the online documentation *Emulator Programming*, pcep.pdf.

The sequence connects to the presentation space of display session A, queries the cursor position, and sends a string and the Enter key to that session.

1. Start a display session A. Now use the EHLLAPI test tool as follows:
2. Click **Connect** and **Connect PS**

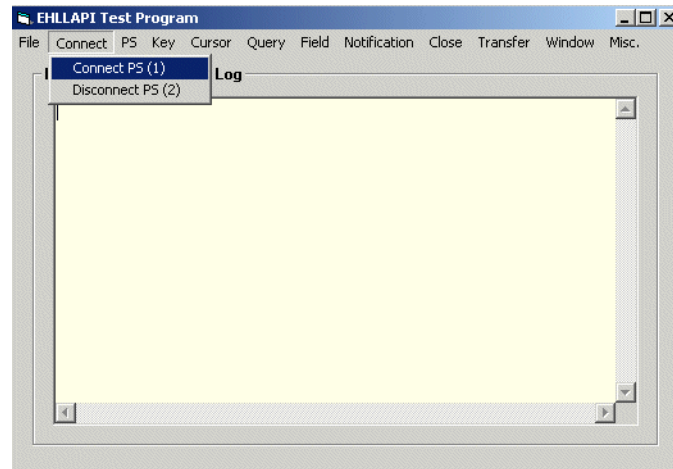


Figure 26-6 Starting EHLLAPI test tool and connecting a presentation space

3. Enter the short session ID to which you want to connect and then click **Execute**. Note that the number in parentheses after the command is the command number as referred to in the online documentation *Emulator Programming*. After the command is executed, it returns a return code, which is zero for successful completion. After that, click **Exit** to leave the window.

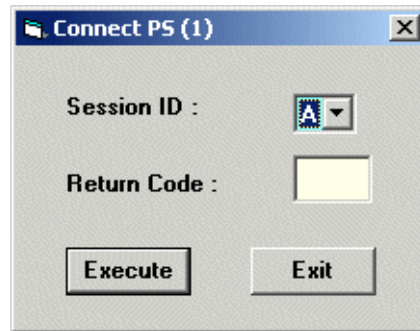


Figure 26-7 Connect Presentation Space (PS) window

4. In the HLLAPI test program panel click **Cursor** and select **Query**.
5. Click **Execute** and then **Exit**.

Note that the window in the EHLLAPI test program gets filled with the commands, the corresponding parameters and return values and return codes.

According to the previous commands, we used subsequently the Key -> Send Key once for sending the string "devm" to the presentation space and afterwards the same command to send the Enter key.

The Personal Communications session window showed the appropriate responses - just as if someone has typed in "devm" and pressed the Enter key.

All actions that have been recorded in the EHLLAPI Test Program can be saved by clicking **File -> Save Log As...** from the menu bar.

The equivalent test tool (vbdde.exe) is available for DDE (Dynamic Data Exchange) in the subdirectory \install\program files\Ibm\Personal Communications\samples\vbdde.

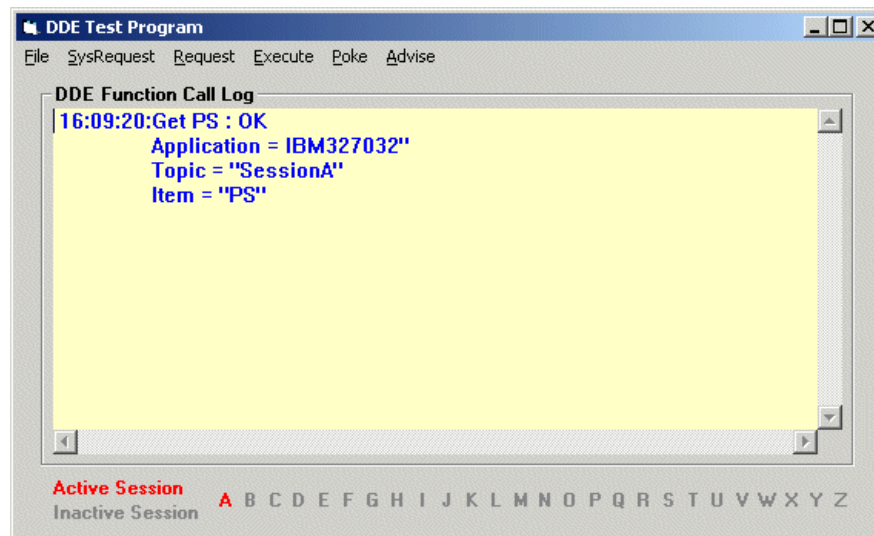


Figure 26-8 DDE Test Program

Figure 26-8 shows the content of the DDE test program after clicking **Request** -> **Get PS** and selecting session **A**:

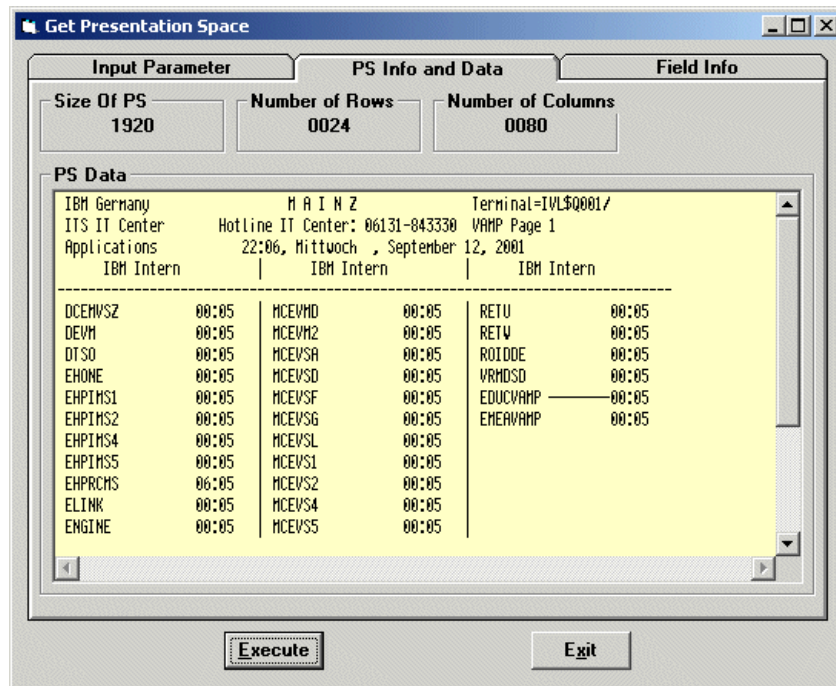


Figure 26-9 Request Get PS with DDE Test Tool

26.3 HACL

In Personal Communications Version 5.6 no new Host Access Class Library (HACL) functions have been added. For details on how to write applications that utilize HACL, please refer to the online documentation *Host Access Class Library* (pcecl.pdf) and Chapter 9 in the redbook *Personal Communications Version 4.3 for Windows 95, 98 and NT*, SG24-4689.



Problem determination

When you experience a problem, a number of methods and tools are available to be used to find its cause. In many cases, problems occur due to configuration errors somewhere at and between the endpoints. This chapter covers some of the most important tools in Personal Communications Version 5.6 to help resolve problems.

27.1 Operator information area (OIA)

The operator information area is the text row below the blue separator line at the bottom of the Personal Communications window. This OIA is common to most hardware terminal types and emulators. Its content is controlled by the host and controllers.

The left part of the OIA shows general information about the current state of the connection and host type, and at the right part of the OIA shows the current cursor position in the line/column format. For details and an explanation of messages from the emulator menu bar, click **Help -> Help Contents -> The Operator information area messages** to display the OIA help window shown in Figure 27-1.

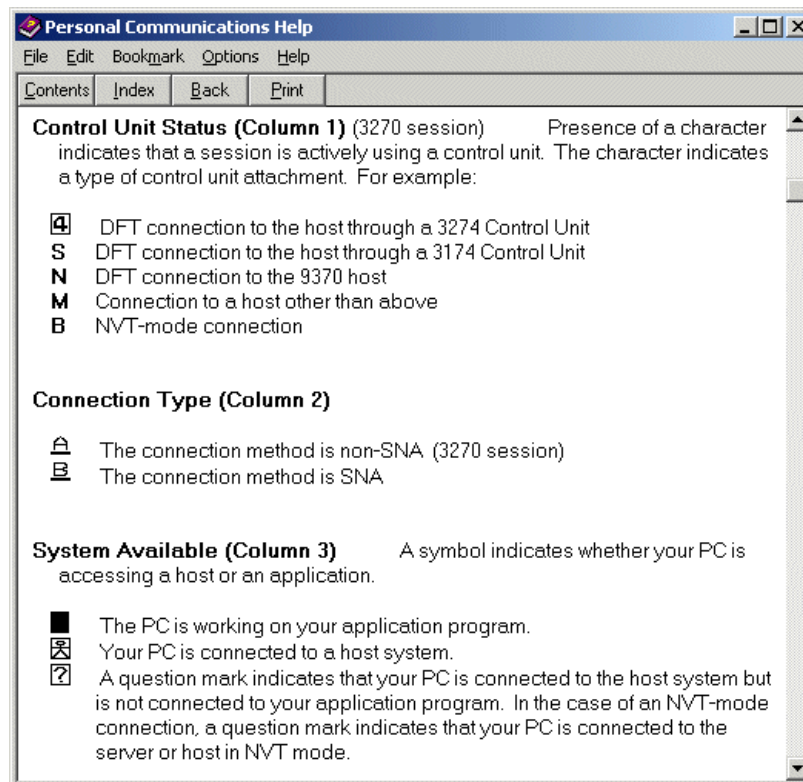


Figure 27-1 Help window for OIA on a 3270 session



Figure 27-2 TN3270 session using 168-bit encryption

Using Figure 27-2, we interpret the OIA:

- ▶ The M in position 1 says that the host connection is not a DFT or NVT connection.
- ▶ The A indicates a non-SNA connection.
- ▶ The stick-man in the box indicates you are connected to the host VTAM system. When the stick-man turns to a solid box, you will be connected to your application.
- ▶ The 24/001 at the far right of the OIA states that the cursor is at row 24, column 1.

In case of a problem during a printer or display session, the status of the OIA will provide important information. For detailed information, use the Help from the menu bar. For the explanation of error codes, click **Help -> Contents -> When you encounter a Problem** from the menu bar.

For help with error messages that are not displayed in the OIA, but that pop up in an error message window, click **Help -> Help Contents -> Contents** from the menu bar and then use the tabulator **Find**.

27.2 Status bar

The grey field below the OIA is the status bar (see Figure 27-2). It contains additional details about the session status, and its content is generated by Personal Communications.

The left-most section indicates the security status:

- ▶ A non-secure session is indicated with an open padlock.
- ▶ A secure session is indicated with a locked padlock and the level of encryption will be indicated with a number. In Figure 27-2, the session is shown secure using 168-bit SSL encryption.

The next section of the status bar is used to contain the current status of the connection. A history of the messages that appear here may be viewed by clicking from the menu bar **View -> Status Bar History**. A window like the one shown in Figure 27-3 will appear.

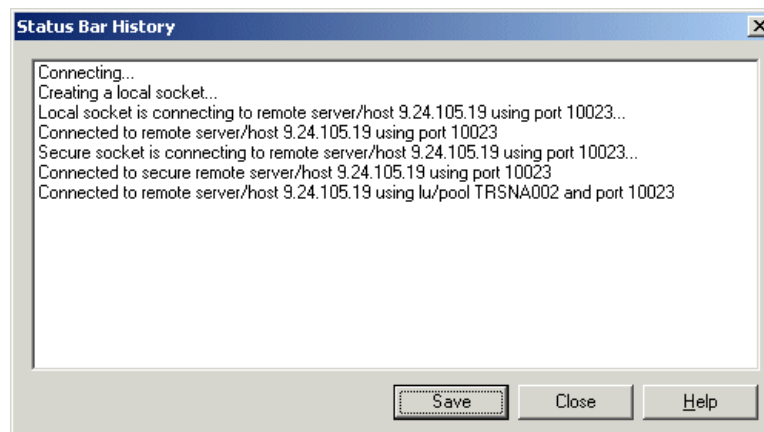


Figure 27-3 Status Bar History window

Telnet sessions do not automatically place entries into the PCWMSG.MLG file; however, from this window, you may save the contents of the history window into the PCWMSG.MLG file, where you can then use the Log Viewer (see Figure 27-4) to examine the entries.

Note: The Status Bar History entries are not logged separately in the PCSWMSG.MLG file. The are merged into the existing PCWMSG.MLG with the remark Emulator in the Component column (see Figure 27-4).

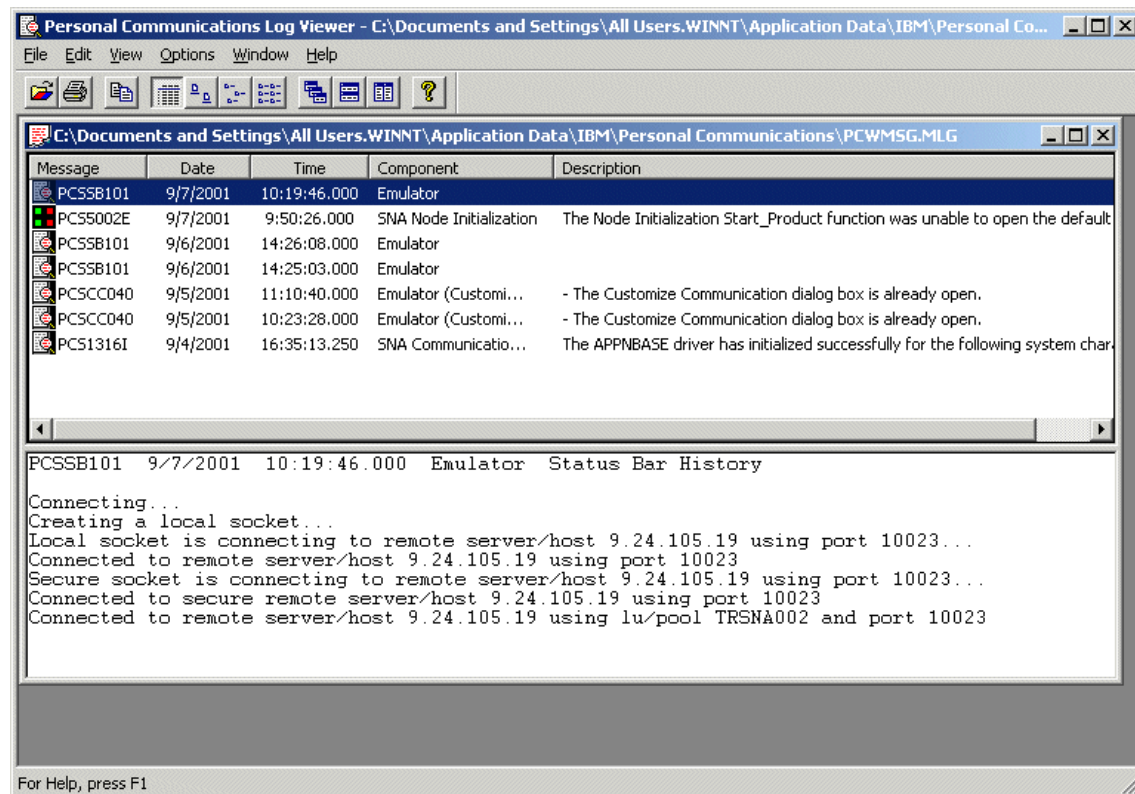


Figure 27-4 Log Viewer displaying Status Bar History saved data

27.3 Tracing / bundling problem determination data

The Trace Facility is able to monitor and record a number of parameters. It can be started in one of the following ways:

- ▶ Via the Windows Start menu. Click **Start -> Programs -> IBM Personal Communications -> Administrative and PD Aids -> Trace Facility**.
- ▶ Via the menu bar of a Personal Communications Version 5.6 Session. Click **Actions -> Launch -> Trace Facility**.

- Via command line issue the command **cstrace**.

Before a trace is started, the session and the node (if SNA is used) should be stopped first so that the start of link and of the session will be captured in the trace. Then follow these steps:

1. Start the trace.
2. Start the session.
3. Recreate the problem.
4. At the point where the problem occurs, take a copy of the window or screen by one of the following methods:
 - Alt+Print Screen for the active window
 - Ctl+Print Screen for the complete screen content
5. Stop the trace.
6. Record the steps that were taken to recreate the problem. Use WordPad to record the steps.
7. Paste the content of the clipboard, which contains the screen copy of the problem, into the readme file.
8. Store the readme file in your Application Data Location along with your configuration files so that it will be saved by the Information Bundler.
9. After the trace has been taken, save and format it in your Application Data Location.
10. Use the Information Bundler to collect the trace and relevant data for Personal Communications Version 5.6 from your Application Data Location.
 - Start the Information Bundler from the menu bar of a session by clicking **Actions -> Launch -> Information Bundler**, or by clicking **Start -> Programs -> IBM Personal Communications -> Administrative and PD Aids -> Information Bundler**.
 - A self-extracting file named x12345.exe will be created and placed in the subdirectory where all configuration files of Personal Communications Version 5.6 reside (which depends on options set during installation).
11. Send the information to IBM support.

27.4 Tracing from the command line

CSTRACE command options

In some cases, it is helpful to start the trace via the command line. Examples are:

- ▶ When only a command prompt is available, such as when doing a remote control of a workstation.
- ▶ If it is easier to send a batch file with the trace commands to users instead of guiding them through the graphical interface.

Tracing execution and formatting is performed using the **cstrace** command. The various options and syntax are listed here.

- ▶ APPLY [-f function_ID -c component_ID -o trace_options] [-r] [-t trunc_length]

This command affects dynamic changes to the currently running options where:

-f function_id	Specifies the function (group) to trace, where function_id is an integer. If you specify the -f flag, you must also specify the -c and -o flags.
-c component_ID	Specifies the component to trace, where component_id is an integer.
-o trace_options	Specifies the trace options to use, where trace_options is a hex value. Where the value has leading zeros, those zeros are optional.
-r	Clears the trace buffer.
-t trunc_length	Specifies the maximum trace data length, where trunc_length is an integer between 992 and 131072. The default is 16352.

- ▶ FORMAT[filename]

This command converts the trace data to a human-readable log. The default file is nstrc.trc. If you specify [filename], the file must specify the extension .trc.

- ▶ RESET

Use this command when you wish to discard the current trace data.

- ▶ SAVE [-a] [filename]

Issue this command to save the current trace data to a file. If you specify the -a flag, the data is appended to the file. The default is to overwrite the current trace data. [filename] is the name of the file to save.

- ▶ SHUTDOWN

This shuts down the trace facility, and exits the program.

- ▶ START [-f function_ID -c component_ID -o trace_options] [-r] [-t trunc_length] [-s]

Use this command to start the trace facility. You may optionally specify the following options at startup, or later use the APPLY option to change them.

-f function_id	Specifies the function (group) to trace, where function_id is an integer. If you specify the -f flag, you must also specify the -c and -o flags.
-c component_ID	Specifies the component to trace, where component_id is an integer.
-o trace_options	Specifies the trace options to use, where trace_options is a hex value. Where the value has leading zeros, those zeros are optional.
-r	Reset the trace buffer.
-t trunc_length	Specifies the maximum trace data length, where trunc_length is an integer between 992 and 131072. The default is 16352.
-s storage_number	Specifies the number of blocks in the trace buffer.
-b block size	Specifies the size of a block in the trace buffer.
-l list file	Specifies a file containing a list with the required trace options in a .dat file.

► STATUS

Displays the current active trace and all its options.

► STOP [-f function_ID -c component_ID -o trace_options]

Suspends one or more active trace options. If you do not specify an option, all active traces are suspended.

-f function_id	Specifies the function (group) to trace, where function_id is an integer. If you specify the -f flag, you must also specify the -c and -o flags.
-c component_ID	Specifies the component to trace, where component_id is an integer.

Note: Please look up the parameter values for the functions, components and options in the file nstrc.cfg located in the subdirectory \en_US\ of your installation path of Personal Communications Version 5.6. Use only a text browser to view that file. Changes to that file will cause the trace function to produce unexpected results.

27.4.1 Example TCP/IP trace from command line

When traces are taken using the graphical interface, users must select the options as shown in Figure 27-5.

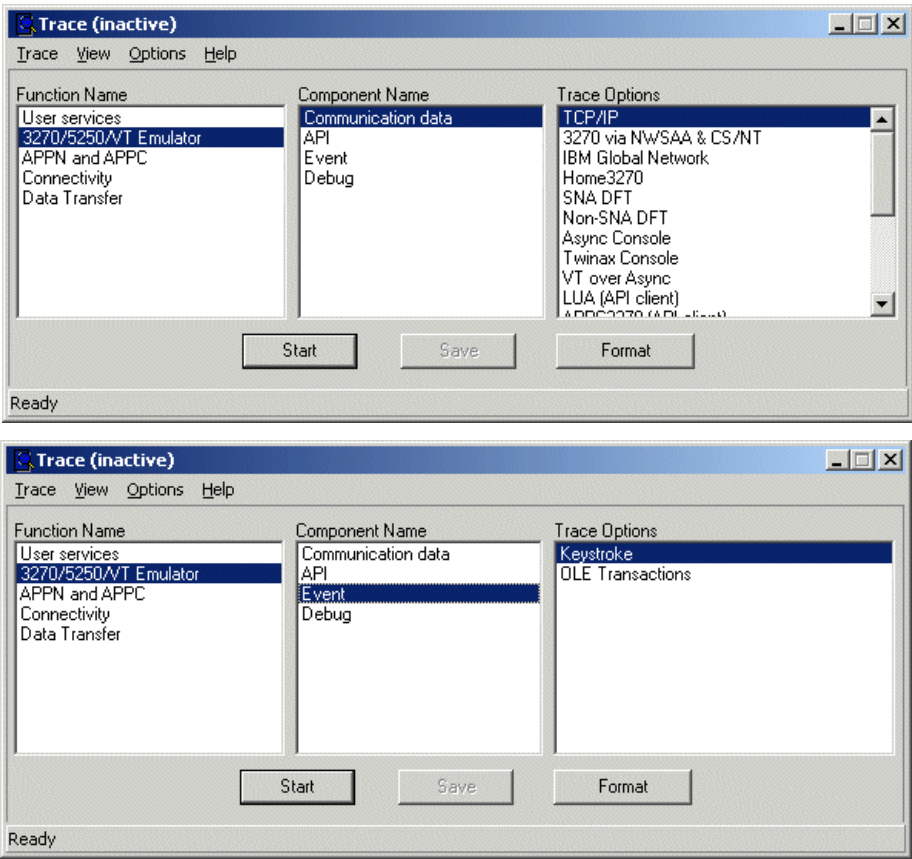


Figure 27-5 Graphical trace selections

You can issue following commands in a batch file to set the same trace parameters as shown in Figure 27-5:

```

cstrace stop
cstrace start
cstrace apply /f 2 /c 1 /o 1
cstrace apply /f 2 /c 3 /o 10000
cstrace status

```

You have to apply the different trace functions in separate **apply cstrace** commands. The response on the screen to the cstrace status is:

```

C:\>cstrace status
Current Active Trace:

GROUP: 3270/5250/VT Emulator (2)
COMPONENT: Communication data (1)
FLAG: TCP/IP (1)

GROUP: 3270/5250/VT Emulator (2)
COMPONENT: Event (3)
FLAG: Keystroke (10000)

```

To stop and save the trace we used the following sequence in a second batch file

```

cstrace stop
cstrace save nstrc.trc
cstrace format nstrc.trc
copy *.nstrc "c:\program files\ibm\personal communications\private"
cd "c:\program files\ibm\personal communications\private"
dir nstrc*.*

```

Screen response:

```

Volume in drive C has no label.
Volume Serial Number is F48E-B3E4

Directory of c:\Program Files\IBM\Personal Communications\private

08/28/2002  03:46p                6,296 nstrc.tlg
08/28/2002  03:46p                2,038 nstrc.trc
08/28/2002  03:46p                  0 nstrcips.trc
               3 File(s)            8,334 bytes
               0 Dir(s)  14,090,272,768 bytes free

```

In that example we copied the resulting trace files to the subdirectory where we keep our configuration files for Personal Communications. From there all data will be copied when the information bundler will be used to collect data for problem determination.

Instead of typing all trace options into the command line or using a batch file you can you use a trace option file. For usage and contents of the trace option file, refer to "" on page 919. A command using a trace option file would be as follows:

```
cstrace start /l c:\pcomm_path\nstrc.dat
```

A sample file NSTRC.DAT containing all possible trace settings is supplied with the CD of Personal Communications Version 5.6, and is located in subdirectory \install\admin\distrib. It is an ASCII file that can be edited to suit your needs. Unwanted trace options should be flagged with a semicolon to be treated as a comment

To verify if the correct options have been applied and are active, you may issue the status option as follows:

```
C:\>cstrace status
Current Active Trace:
GROUP: 3270/5250/VT Emulator (2)
COMPONENT: Communication data (1)
FLAG: TCP/IP (1)

GROUP: 3270/5250/VT Emulator (2)
COMPONENT: Event (3)
FLAG: Keystroke (10000)
```

The **cstrace save** and **cstrace format** options must include a full file specification to the Application Data Location specified for your installation. It is the directory where the configurations files are located.

To find out a parameter for a command-line trace, you can use the GUI to set up the trace, start the trace from GUI with all parameters selected as needed, then issue **cstrace status** from a command line. The output shows the equivalent command-line parameters. Apply the reading of the output to the command-line parameters of **cstrace**.

You may also invoke the Information Bundler via the command-line interface:

```
pcspd /q
```

Where /q suppresses the pop-up window asking for the registry keys to be included.



Part 3

Screen Customizer

IBM Screen Customizer V2.0.70 is the component of Host Access Client Package V3 that provides the ability to present host screens as a graphical user interface when used with Host On-Demand. This part discusses the installation and deployment of Screen Customizer and how to use it to customize host application screens to improve usability for casual users.



Screen Customizer

IBM Screen Customizer Version 2.0.60 is a Java client for Host On-Demand that provides a graphical user interface alternative to the host application “green screens”. It has the ability to change a standard 3270 or 5250 emulator application into something that’s comparable to a typical Web application, making 3270 and 5250 host applications easier to use. Screen Customizer can combine data from multiple screens, hide unneeded information from the user, and change cryptic mainframe input fields into more friendly forms, such as radio buttons, check boxes, drop-down lists or valid value lists depending on the input required.

In short, Screen Customizer can provide a face-lift to applications, providing new life for 3270 and 5250 applications whose only problem is their user interface. Screen Customizer can extend the life of the mainframe’s terminal-based applications while a Web-based replacement is being built, or even serve as an end-of-life substitute for little-used applications. The following are some of the immediate benefits that can be derived upon deploying Screen Customizer.

- ▶ Users unfamiliar with traditional host application green screens typically find interfaces similar to Web pages easier to use.
- ▶ Customization can be established for individual users or groups, so that the graphical screens provide controlled application access and flow accordingly.
- ▶ Customization can be done offline by a graphics artist.
- ▶ Host On-Demand provides all Telnet connectivity and security, enabling its use in most TCP/IP network environments.

28.1 Screen Customizer overview

IBM Screen Customizer Version 2.0.60 is deployed as an applet to be used with the Host On-Demand Client. Once enabled in the session configuration parameters, Screen Customizer displays all screens in a graphical fashion. It interprets the host data stream then provides either a default graphical representation of the host screen, or a customized graphical representation of the host screen, that was created by means of the Screen Customizer Customization Studio.

Screen Customizer can impose a large degree of consistency on mainframe applications that have evolved over a number of years (often decades). The result of this evolution is often a set of very stable applications with an inconsistent or command-driven user interface. It's not uncommon to see an application where the same function key means many different things depending on the application and context. This increases the difficulty of using the application and the time it takes to train new users. Screen Customizer can remap function keys based on individual screens or use graphical controls such as buttons to provide navigation that is consistent regardless of what application is being used.

In Screen Customizer applications are captured screen by screen, then customized to become graphical user interfaces. The Screen Customizer Administrator examines every screen it encounters and collects the number, length and relative position of fields on the screen and stores this information, along with the screen ID assigned by the administrator, in a database. This screen ID is referred to as a screen map. At runtime, Screen Customizer again collects the screen information and uses it to search the database to determine if the screen has been customized. If a match is found, the screen ID is extracted from the database and is used to request the associated screen map from the Web server. The Web server returns the screen map, which Screen Customizer uses to build and display the customized screen in place of the default character-based user interface screen. If a screen ID is not found, the default graphical representation is shown.

The transition from the normal application green screen appearance to a fully customized graphical interface is depicted in the three figures that follow. The steps taken to effect this transition are found in 28.7, "The Screen Customizer development cycle" on page 942.

Figure 28-1 depicts the appearance of the traditional green screen for the application.

The screenshot shows a window titled "DALVM1 - A" with a menu bar (File, Edit, Transfer, Appearance, Communication, Assist, Print, Help) and a toolbar with icons for Jump, Same, Exit, Send, Recv, Copy, Paste, PtfSom, Remap, Color, Play, Record, Stop, Pause, and Macro. The main display area contains the following text:

```

100                               Search Request

To request a search, type the search data and press Enter.
                                           Lines __ to 22 of 32

Corporate and Personal Directories
  Name . . . . .
  Search Locs/Nodes/Dirs .
  Node . . . . .
  User ID . . . . .
  Extension . . . . .
  Job Responsibilities .
  Department . . . . .
  *any field (Name Value) .
  *any field (Name Value) .
                                           ex: DIV 1*
                                           ex: MGR Y

Departments Directory
  Department Title/Number .
  Search Locs/Nodes/Dirs .

Distribution Lists
  List Name . . . . .

Services Directory
  Service Keyword(s) .
  Search Locs/Nodes/Dirs .
  Facility (F1 gives help)
  City . . . . .
  State . . . . .
                                           ex: =IBMUS
                                           ex: MKTG
                                           ex: Dallas
                                           ex: CT

Command ==>
F1=Help F2=Set 2 F3=Exit F4=Profile F5=Refresh F6=Fuzzy search
F7=Backward F8=Forward F9=Command F10=Actions F11=Dist list F12=Cancel
  
```

At the bottom of the window is a keyboard layout with buttons for PF1 through PF12, Enter, PA1, Attn, Insert, NewLine, PF7, PF8, PF9, PF10, PF11, PF12, Clear, PA2, SysReq, Delete, and NextPad. A small icon and the text "Signed by: International Business Machines" are visible in the bottom left corner.

Figure 28-1 Traditional emulator screen

The default graphical interface is achieved by merely enabling Screen Customizer in Host On-Demand session properties as shown in Figure 28-2.

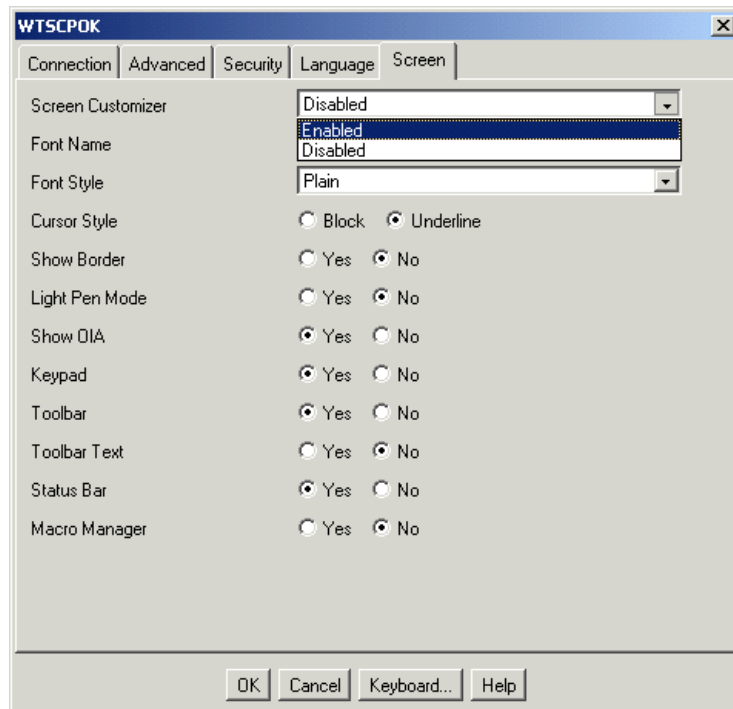


Figure 28-2 Enabling Screen Customizer

Figure 28-3 shows the appearance of the same green screen being displayed using the default graphical interface.

DALVM1 - A.

File Edit Transfer Appearance Communication Assist Service Print Help

100 Search Request

To request a search, type the search data and press Enter. Lines to 22 of 32

Corporate and Personal Directories

Name

Search Locs/Nodes/Dirs

Node

User ID

Extension

Job Responsibilities

Department

any field (Name Value) ex: DIV 1

*any field (Name Value) ex: MGR Y

Departments Directory

Department Title/Number

Search Locs/Nodes/Dirs

Distribution Lists

List Name

Services Directory

Service Keyword(s)

Search Locs/Nodes/Dirs ex: =IBMUS

Facility (F1 gives help) ex: MKTG

City ex: Dallas

State ex: CT

(C) Copyright IBM Corporation 1988, 1992. All rights reserved.

Command ==>

F1=Help F2=Set 2 F3=Exit F4=Profile F5=Refresh F6=Fuzzy search
F7=Backward F8=Forward F9=Command F10=Actions F11=Dist list F12=Cancel

F1	F2	F3	F4	F5	F6	Enter	Reset	Clear	Insert	NewLine
F7	F8	F9	F10	F11	F12	Erlnp	ErFld	ErEOF	Delete	NextPad

Signed by: International Business Machines

Figure 28-3 Default graphical interface

As you can see, by simply using the default graphical interface available with Host On-Demand you can easily identify all the input fields and function keys.

The simple process of customization can further transform the appearance of this application to a sophisticated graphical interface. Figure 28-4 shows the appearance of the same screen after customization.

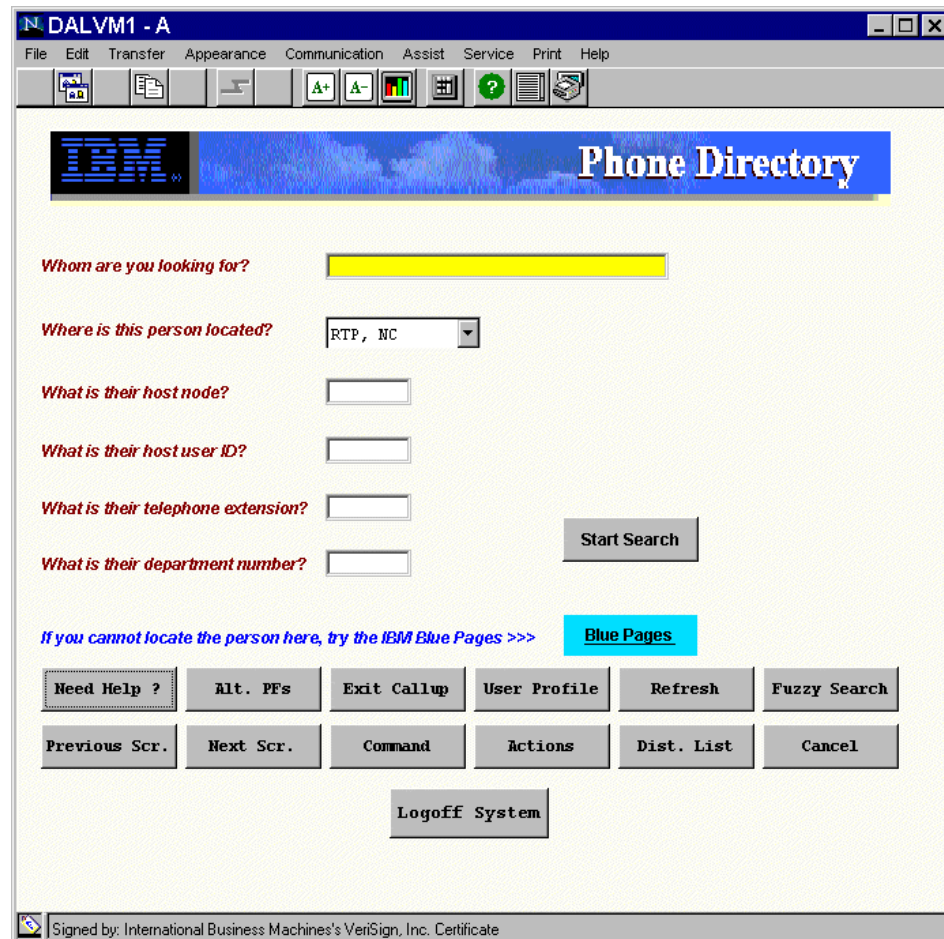


Figure 28-4 Customized graphical interface

Customization for 3270, 5250, and CICS Gateway client sessions is supported, while VT display client customization is not. The creation of a customized graphical interface involves no changes to the host application nor any programming on the workstation.

28.2 Screen Customizer features

The following is a list of features introduced with Screen Customizer Version 2.0.

- ▶ Light pen support (3270 only).
- ▶ An SSL indicator (formerly available only for Host On-Demand users).

- ▶ A Service Bundler tool that creates a package of files for IBM service.
- ▶ AS/400 subfiles and table support.
- ▶ The addition of an API that allows a Java programmer to interact with Screen Customizer host sessions and objects within those sessions. This allows a custom-written Java applet or application to dynamically change values, settings or the appearance of the current Screen Customizer application.

With the introduction of Screen Customizer Version 2.0.60, the following features were added:

- ▶ Support for Host On-Demand Version 6.0

Note: Screen Customizer V2.0.60 does not support Personal Communications Version 5.6 or any release of Host On-Demand prior to Version 6.0.

- ▶ Netscape 6.0 Support for Screen Customizer Version 2.0.60
Support for Netscape 6.0 (Java 2) is available with Screen Customizer/LE, also referred to as the default graphical interface. The Customization Studio, Administrator, SCCI and Screen Customizer Beans are only supported in a Java virtual machine 1.1.8 environment.
- ▶ Support for Java 2 enabled browsers
Clients are supported on Java 2-enabled Web browsers, such as Netscape 6.x and Mozilla. The Java 2 Plug-in for use with Netscape 4.x and Microsoft Internet Explorer is also supported.
- ▶ Greek Translation
Screen Customizer has been translated into Greek.

28.3 Screen Customizer components

Screen Customizer has three individual components, each having a specific functionality, all of which are Java applets.

Administrator

This applet is used to capture host screens and assign them IDs. It is also used to set global defaults for screen fonts, colors, button styles, and other attributes, and saves the defaults in a profile. The Administrator requires a live host session; therefore, it requires Host On-Demand to provide the Telnet session.

Customization Studio

This applet allows you to create customized versions, called maps, of the host screens captured by the Administrator component. No programming is involved. Customization does not require a host connection, so it could be done by people working offline, for example at home.

Runtime Client

The runtime is downloaded from a Web server along with the Host On-Demand applet or installed locally on a client workstation. Once the data stream is received by Host On-Demand, control is passed to Screen Customizer which uses its screen definition database to determine whether to present the default graphical interface or the customized version of the screen sent by the host. If a screen ID is found in the database for the screen, the map is downloaded from the Web server and the graphical representation rendered. If a screen ID is not found, the default graphical interface is rendered.

28.4 Planning for Screen Customizer

Screen Customizer Version 2.0.60 Client or Administrator requires that Host On-Demand Version 6.0 be installed. The Customization Studio is an offline utility: therefore, it does not require Host On-Demand to operate.

Screen Customizer only manipulates the appearance of the 3270 or 5250 application data, relying on Host On-Demand to provide the basic Telnet connection and transport. Host On-Demand must be installed before installing Screen Customizer. If, for any reason, you reinstall Host On-Demand you must also reinstall Screen Customizer, because the default Screen Customizer runtime and the full version of Screen Customizer share the same file names. Failure to do so will result in the inability of the Administrator to function properly and the runtime client unable to display customized maps.

You should not install a language for Screen Customizer if it has not been installed for Host On-Demand. If you install a language that was not installed for the base product, Screen Customizer will not work in that language.

If you are installing the full version of Screen Customizer on a Host On-Demand server, the Screen Customizer Runtime client replaces the default graphical user interface provided with Host On-Demand while maintaining that functionality.

28.5 Installation of Screen Customizer

Screen Customizer Version 2.0.60 operates only with Host On-Demand Version 6.0 and above. If Host On-Demand Version 6.0 is not installed, you can only install the Customization Studio, since it does not require a host connection.

Screen Customizer includes:

- ▶ Runtime Client for Host On-Demand (all platforms)
- ▶ Administrator (Windows only)
- ▶ Customization Studio (Windows only)

28.5.1 Administrator and Customization Studio Installation

The Screen Customizer application development environment, administrator and Studio, must be installed on a Windows 32-bit platform. It has been tested and certified on Windows 95, Windows 98, Windows ME, Windows NT with SP 5 (or higher) and Windows 2000. Before installing Screen Customizer Version 2.0.60, you must install a local copy of Host On-Demand Version 6.0 or above.

There are two installation options of Screen Customizer on a Windows platform as shown in Figure 28-5. Selecting **Full** will install all three components, while selecting **Custom**, as shown in Figure 28-6 on page 932, will allow you to tailor your installation further.

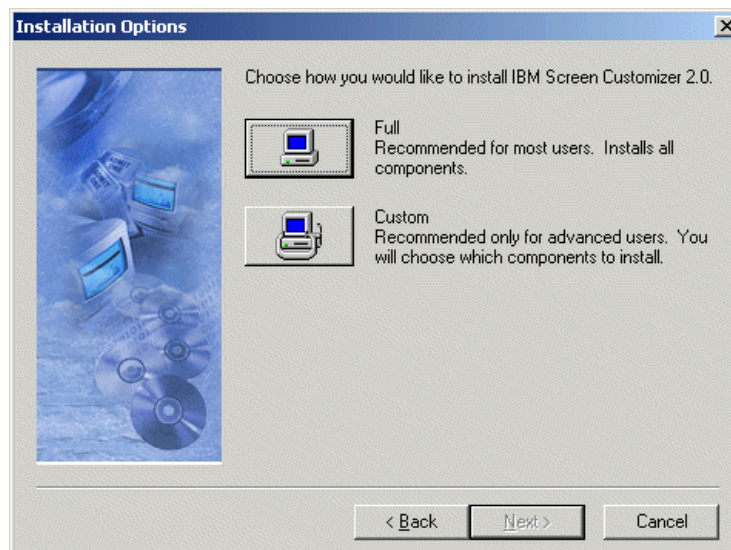


Figure 28-5 Screen Customizer installation

If you want a complete installation, you must select the **Full** option. If you choose not install all the components, choose **Custom** install. The following information should provide the necessary guidelines:

1. Selecting **All Components** is the same as the default or full installation. Use this installation choice if your job will include that of administrator and customizer.
2. Selecting the **Client** installation option installs only the Screen Customizer runtime code. Use this installation if you are installing Screen Customizer on a workstation that will be used for testing or local deployment, or you are installing only the runtime components on a Windows Host On-Demand server.
3. Select **Customization Studio** if you will be doing customization of previously captured screens. The Customization Studio can be used in a completely offline environment. Use this type of installation if you are doing screen customization only. In this environment the administrator must capture the screens and move them to a file server, NFS drive, or other device where the Customization Studio operator may access and tailor them.

Note: It is not necessary to have a locally installed copy of Host On-Demand to install just the Customization Studio.

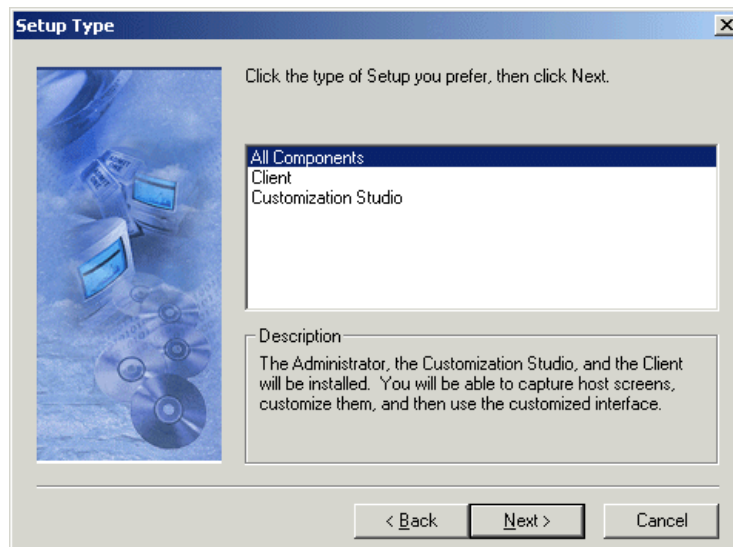


Figure 28-6 Screen Customizer custom installation

Please refer to the online installation guide and *Getting Started Guide* for more information on the custom installation of these products.

28.5.2 Runtime installation

The IBM Screen Customizer runtime is supported on any server that supports Host On-Demand. For the complete list of servers and platforms where support is available, please refer to the installation guide of Host On-Demand.

Windows NT/2000 server

The installation on a Windows NT or Windows 2000 server-based platform is controlled by a graphical installation program. It can either be done in normal mode or in silent mode. For silent mode installation, please refer to 28.5.4, “Silent installation” on page 940.

To install the client, please follow these steps:

- ▶ Insert the installation CD into the CD-ROM drive.
- ▶ If your desktop is configured for auto-start, it would automatically start the Screen Customizer installation software. Otherwise run the **Setup.exe** present in the win32 directory of the installation CD.
- ▶ Choose the **Full/Complete** installation unless and until you wish not to install all of the components of Screen Customizer.

AIX

The installation on AIX also provides a graphical interface similar to the Windows interface as well as the silent mode installation.

To install the client using the graphical interface:

- ▶ Mount the CD-ROM drive and insert the CD.
- ▶ Change to the root directory of the CD and enter **setupaix.sh**.
- ▶ Click **Install Product**.
- ▶ Follow the directions in the installation windows.

UNIX

Screen Customizer runtime client must be installed in the Host On-Demand server publish directory so that it is available to client workstations.

- ▶ Insert the CD and mount it.
- ▶ Change to the Host On-Demand publish directory.
- ▶ Untar the client.tar file to install the base files into the HOD directory. Support for English is installed by default. Additional languages must be installed separately. This step assumes that the tar files are in the /cdrom/tar directory.
- ▶ Run the following command from the publish directory:

```
tar -xf /cdrom/tar/client.tar
```

- ▶ For each additional language that you want to install, run:

```
tar -xf /cdrom/tar/sc_lang.tar
```

For example, to install support for the Korean language, run:

```
tar -xf /cdrom/tar/sc_ko.tar
```

To extract the documentation file (includes all languages), run:

```
tar -xf /cdrom/tar/doc.tar
```

iSeries

To install IBM Screen Customizer on an iSeries:

- ▶ Shut down the Host On-Demand by issuing the command ENDH0DSVM.
- ▶ Insert the Screen Customizer CD.
- ▶ Run the following command:

```
RSTLICPGM LICPGM(5648D76)DEV(OPT01)
```

Where:

- RSTLICPGM starts the OS/400 installation program
- LICPGM(5648D76) is the Screen Customizer program number to install
- DEV(OPT01) is the source device for the CD

- ▶ Start Host On-Demand by issuing the command STRH0DSVM.

Novell NetWare

Screen Customizer must be installed in the Host On-Demand server publish directory so that it is available to client workstations.

To install the client:

- ▶ Insert the CD.
- ▶ Change to the Host On-Demand publish directory.
- ▶ To extract the files, run the following command from the Host On-Demand publish directory:

```
unzip -d [cd_rom]:\zip\client.zip
```

- ▶ To extract the documentation file, run:

```
unzip -d [cd_rom]:\zip\doc.zip
```

Where:

- unzip is your unpacking program. It must support long filenames.
- -d is the parameter that recreates the zipped directory structure.

- [cd_rom] is the CD-ROM drive letter.
- zip is the directory on the CD.

OS/2

In OS/2, Screen Customizer must be installed in the Host On-Demand server publish directory so that it is available to client workstations.

To install the client:

- ▶ Insert the CD.
- ▶ Change to the Host On-Demand publish directory.
- ▶ To extract the client files, run the following command:

```
unzip -d [cd_rom]:\zip\client.zip
```

- ▶ To extract the documentation file, run:

```
unzip -d [cd_rom]:\zip\doc.zip
```

Where:

- unzip is your unpacking program. It must support long filenames.
- -d is the parameter that recreates the zipped directory structure.
- [cd_rom] is the CD-ROM drive letter.
- zip is the directory on the CD.

28.5.3 Screen Customizer considerations for OS/390 and z/OS

Screen Customizer V2.0.60 can be installed on an OS/390 server. The Administrator and Customization Studio components are not installed with Screen Customizer for OS/390; they can run only on Windows 95, 98, NT or 2000. Both the Administrator and Customization Studio require that you first install a local copy of Host On-Demand. Refer to 2.4.10, “Installing the locally installed client” on page 62.

Once screens are customized, they can then be uploaded to the OS/390 server to be served through Host On-Demand. This section discusses the OS/390 Screen Customizer installation and deployment considerations.

Planning

Screen Customizer consists of only one FMID, HCUT206, and must be installed in the same SMP/E environment as Host On-Demand V6. Screen Customizer V2.0.60 works only with Host On-Demand V6; it will not work with any previous release of Host On-Demand.

The installation of Screen Customizer is similar to the Host On-Demand installation using SMP/E. The package is provided in three formats - 6250 tape, 3480 cartridge, and 4mm cartridge. A program directory is supplied with the package; this provides the information necessary to install Screen Customizer using SMP/E and activating Screen Customizer.

It is recommended that you review the current support, product information, and hints and tips found on the following Web sites:

- ▶ Support site:
<http://www.ibm.com/software/network/screencustomizer/support/>
- ▶ Product information site:
<http://www.ibm.com/software/network/screencustomizer/>
- ▶ Program directory softcopy (latest copy):
<http://www.ibm.com/software/network/screencustomizer/library/>

Maintenance for Screen Customizer V2 for OS/390 can be received in one of two ways; both are SMP/E installed:

- ▶ Go to the support Web site and select **Support Downloads**. You must be registered with the IBM Software Internet Service Delivery site. Refer to 2.7, “Service updates” on page 77.
- ▶ Order the PTF tape via IBM support.

Screen Customizer requires that Host On-Demand be installed first. The product will not install successfully if Host On-Demand is not present. The SMP/E processing will complete, but a post-SMP/E step will fail.

Since Host On-Demand is required, the Screen Customizer installation instructions assume the UNIX System Services environment is set up for the PATH, LIBPATH, and, if using Java 1.1.8, CLASSPATH variables. Refer to 3.3.1, “UNIX System Services environment” on page 86.

Important: Depending on when the product was ordered, you may receive PTF tapes in addition to the base product tapes. You must install the base product before installing the PTF tapes.

DASD storage requirements

DASD storage is required for the target and distribution libraries and for the HFS (Hierarchical File System). Work space is also needed during the SMP/E installation. The program directory outlines the storage requirements for Screen Customizer and for SMP/E. We found that the number of tracks outlined in the program directory for distribution library ACUSHFS1 needed to be increased to 1700 tracks.

The base Screen Customizer, prior to running the mvsccli.sh shell script, required 60 cylinders on a 3390. After the mvsccli.sh shell script completed, 145 cylinders were in use before any customization. We allocated an HFS of 250 cylinders on a 3390. After the apply step, the HFS was 24% full. After running the mvsccli.sh shell script, the HFS was 58% full with one extent. This leaves 42% of 250 cylinders for customization and maintenance.

Installation

In this section we will detail the installation of Screen Customizer on OS/390. We discuss the installation jobs and instructions and the post-SMP/E processing.

SMP/E processing

Screen Customizer ships with sample jobs that can be copied from the product tape. The program directory provides the JCL to copy the sample jobs.

CUSRECVE	Sample RECEIVE job
CUSALLOC	Sample job to allocate target and distribution libraries
CUSDDDEF	Sample job to define SMP/E DDDEFs
CUSHFS	Sample job to define Screen Customizer HFS data set (optional)
CUSISMKD	Sample job to invoke the supplied cusmksdir REXX EXEC to define the HFS paths
CUSAPPLY	Sample APPLY job
CUSACCPY	Sample ACCEPT job

The sample jobs should be updated to reflect the CSI, target zone, and distribution zone names used in the installation.

Installation instructions

Follow the instructions in the program directory. If allocating a new HFS using CUSHFS, we recommend increasing the space allocation to accommodate future service updates.

Create the mount point and make sure it has permissions of 755. For example:

```
TSO MKDIR '[PATHPREFIX]/usr/lpp/customizer' MODE (7,5,5)
```

where [PATHPREFIX] is the appropriate high-level directory name. For users installing in the default path, this would be null. For others, the high-level directory may be something like /service/ or some meaningful name for your installation.

Mount the HFS to the system; it must be mounted in read and write access. You can omit the mode parameter on the mount command; the default is RDWR.

```
TSO MOUNT FILESYSTEM('hfsprfx.hfs')  
MOUNTPPOINT ('[PATHPREFIX]/usr/lpp/customizer') TYPE(HFS)
```

Regardless of whether a new or existing HFS is used, you must run **CUSISMKD** to allocate the HFS paths for the Screen Customizer product.

Post SMP/E processing

Once the SMP/E processing is complete, the mvsccli.sh shell script must be run to configure Screen Customizer with Host on-Demand. The script resides in /usr/lpp/customizer directory unless the default path was changed during the SMP/E installation. If the production path is different from the installation path, you must mount the HFS on the production mount point before running the shell script. For example, if Screen Customizer is SMP/E installed on mount point /service/usr/lpp/customizer but the production mount point will be /usr/lpp/customizer, you must mount the Screen Customizer HFS on the production mount point (/usr/lpp/customizer) before running mvsccli.sh shell script. The Host On-Demand HFS must also be mounted on its production mount point.

The mvsccli.sh shell script untars both the Screen Customizer code and documentation to the created HFS directories. The shell script also creates other HFS directories and calls the HODSEDLink-UNIX shell script. This script creates the symbolic links to Host On-Demand. You should not call HODSEDLink-UNIX manually. During the execution of mvsccli.sh, a HODLink-UNIX-errors file is created. If the installation runs without errors, the last two lines in the file should be:

```
Linking HOD files  
Done.
```

If any errors occur during the execution of the shell script, the errors will be listed in the HODLink-UNIX-errors file and displayed on the UNIX System Services console. Correct the error and restart the mvsccli.sh shell script.

The mvsccli.sh shell script takes several minutes to run. Do not stop the script. Upon completion, the resulting HODLink-UNIX-errors file will be very large.

Java 1.3 consideration

If you are using Java 1.3, you may not have the CLASSPATH variable set, since it is not necessary for Java information. The mvsccli.sh shell script requires the CLASSPATH variable to not be null. If it is null and you try to run the mvsccli.sh shell script, you will receive the following error:

```
*** This program uses Java. You must set the CLASSPATH to run it.
```

To avoid getting the error, you can verify the CLASSPATH is null by issuing the following command:

```
echo $CLASSPATH
```

If it is null, set the CLASSPATH to something. We set the variable with the following export command:

```
export CLASSPATH=.
```

If you had previously tried to run the mvsccli.sh shell script, you will get the following error, which can be ignored:

```
mkdir: FSUM6404 directory "customizer": EDC5117I File exists.
```

The shell script will continue to run for several minutes. After it has completed, verify no other errors are in the HODLink-UNIX-errors file.

Deployment considerations

Since there is no Customization Studio on the OS/390 system, you must create your customized screen on a Windows platform as described in 28.7.2, “Using the Customization Studio” on page 953. Once the screens are customized, you can transfer the customization to an OS/390 Host On-Demand server with Screen Customizer installed.

The customization will be in multiple folders on the Windows system where the Customization Studio was used. These folders, described in Table 28-1 will need to be transferred to the OS/390 server into the corresponding directories.

Table 28-1 Directories for customized screens on OS/390

Folder on Windows	Directory on OS/390	Contents
\map	customizer/custom/map	Contains the screen.db (screen database) file, all screen maps (.scm files) and template files (.tpl files)
\img	customizer/custom/img	Contains all graphics used by the application,restricted to GIF (.gif) and JPEG (.jpg) files
\lst	customizer/custom/lst	Valid values list
\ps	customizer/custom/ps	Base screen data
\ref	customizer/custom/ref	Reference files for field help and valid-value list
\wsp	customizer/custom/wsp	Global customizations

Folder on Windows	Directory on OS/390	Contents
\en\help or \lang\help	customizer/custom/en/help	Help information

On the OS/390 Screen Customizer install, the subdirectories are not automatically created except for the custom/img directory. We created the subdirectories in the custom directory using the following commands. Our installation is in the /usr/lpp/customizer/customizer/custom directory path.

1. **cd** /usr/lpp/customizer/customizer/custom
2. **mkdir** map
3. **mkdir** lst
4. **mkdir** ps
5. **mkdir** ref
6. **mkdir** wsp
7. **mkdir** en
8. **mkdir** en/help

Using FTP, transfer the contents of the folders from Windows to the OS/390 in binary format to the respective directories. Depending on your customization, some of the subdirectories could be empty. We used the FTP client function of Host On-Demand to transfer the files to the host. With the FTP client, we could list the contents of the source folder and by marking all the files within a folder, all were transferred at once.

On the OS/390, verify the permissions of the files transferred are 755. If the Host On-Demand server is already active, you do not need to recycle the server to activate the customized screens. For the sessions you wish to use Screen Customizer, you should enable Screen Customizer from the Screen tab of the session properties window as shown in Figure 28-2 on page 926.

If you do not see your customized screens, verify the files were transferred in binary. You may also need to get a Web server trace to determine what might be in error.

28.5.4 Silent installation

A silent mode installation via a response file is available. The silent installation installs Screen Customizer without displaying any windows or asking for input. To perform a silent installation of Screen Customizer, you must first create a response file that contains the information required on the installation windows. The installation CD of Screen Customizer comes with sample response files, which are located in the \INSTMGR directory. The Windows sample response file

is server1.iss, and the AIX sample response file is install.script. These samples contain the default installation options. You can use those or create your own. Once a response file is created, start the silent installation. It is recommended that you create your own response file.

Note: Complete instructions on silent mode installation are documented in the *Getting Started Guide*, which is found on the IBM Screen Customizer Web site at the following URL:

<http://www.ibm.com/software/network/screencustomizer/library/>

When you install in silent mode, there is no indication that installation is in progress or that it is complete.

A silent installation may prove to be very useful for large enterprises and such other locations where you must install Screen Customizer over a large number of workstations across the network.

28.6 Migration

If you are installing onto a Windows system and if you are migrating from IBM Screen Customizer Version 1, migration will occur automatically. The following sections describe the migration process in more detail, and the procedures for migration when installing on a non-Windows platform.

In Screen Customizer Version 2, profiles are no longer supported and have been replaced with global templates. When installing on Windows, profiles are automatically migrated from the default custom/wsp directory to the equivalent template name (with a .tpl extension) in the custom/map directory. Profiles that are not located in the default directory, for example, a user-defined directory mycustom/wsp, are not migrated automatically during installation. A profile migration utility is provided that migrates those profiles to the new template format. On Windows systems, the migration utility is started by clicking **Start -> IBM Screen Customizer**. On non-Windows systems, the utility can be started manually.

To start the migration utility manually, enter the following command (on one line):

```
java -classpath publish_dir\lib\scmigr.jar;publish_dir\lib\rt.jar;  
publish_dir\lib\i18n.jar com.ibm.hi.customizer.util.profile.ProfileMigrator  
ProfileDir=profile_dir TemplateDir=template_dir
```

28.7 The Screen Customizer development cycle

Developing a Screen Customizer application falls into five basic steps:

1. Administration: Screens within an application are identified and saved as Screen Customizer maps.
2. Screen Customization: This step is where a screen gets a face lift.
3. Template Development: This activity can take place in parallel with screen customization or afterwards. Templates allow the developer to provide a more uniform look and feel for all screens in an application (even screens that have not been customized) without working on individual screens.
4. Testing: Once the application has been built, it should be moved to a stand-alone client or a test server where the application can be exercised to ensure it is fully functional.
5. Deployment - once tested, the application can be moved to a production server or servers.

28.7.1 Screen Customizer Administrator

Screen Customizer Administrator allows you to capture and customize the host application screens by substituting Web page-like objects for application fields to create a true graphical user interface. Some of the objects that are available for customizing your host screen are:

- ▶ Images
- ▶ Image buttons
- ▶ Frames
- ▶ Buttons
- ▶ Logos
- ▶ Check boxes
- ▶ Choice boxes
- ▶ Radio buttons
- ▶ Lists
- ▶ Valid value buttons
- ▶ Web link buttons

The Administrator applet configures and runs live sessions to host applications. Screens are identified in these sessions that are to be saved for customization. A screen ID must be assigned to each screen, and these IDs are stored in a screen database file on the server called screen.db. Subsequently, host screens are captured and saved as Screen Customizer maps that serve as the starting content for customization using the objects mentioned above. Both the screen database and the screen maps are saved in the \custom\maps directory.

The Screen Customizer Administrator can be run on several machines to capture screens for transforming an application. If users customize applications from more than one workstation, then the contents of the screen databases from these Administrator systems would have to be merged. A utility is provided to perform this function; see “Merging screen databases” on page 951.

The Administrator is also used to change global settings for graphical screen characteristics such as fonts, available window features, and other graphical characteristics. This global customization is saved in a user profile in the \wsp directory; by convention the default user profile is named default.wsp. As many user profiles as needed can be created in the same manner.

The Administrator is launched by the HODCustomAdmin.html file, which resides in the Host On-Demand publish directory.

The Administrator must be run only on a Windows 95, Windows 98, Windows ME, Windows NT or Windows 2000 system that has Host On-Demand installed. A user named customadmin is set up by default during the Screen Customizer installation, and that user ID is enabled to perform all of the tasks involving the identification and capturing of screen maps.

Also, the Administrator and Customization Studio should not be run with the Host On-Demand cached client whenever possible. Many of the functions in both are nonfunctional within the cached client. To remove the cached client, follow the appropriate procedure in the online Host On-Demand *Getting Started Guide*.

Screen capture and global customization with Administrator

The sections that follow are intended to familiarize you with some of the features of the Screen Customizer Administrator, Customization Studio, and Client. They are not intended to serve as a replacement or substitute for the *Screen Customizer Administrator's Guide* and online help, to which you should refer for more information.

These sections assume that Screen Customizer has already been installed correctly.

Starting the Administrator

To start the Screen Customizer Administrator:

- a. Open the file HODCustomAdmin.html in a browser.

If Screen Customizer is installed on a Windows system, click **Start -> Programs -> IBM Screen Customizer -> Administrator**; or you can click **HODCustomAdmin.html** in Windows Explorer. The Screen Customizer session configuration window appears as shown in Figure 28-7.

- b. Click **Add Sessions**. Configure the connection properties for the session and enable the Screen Customizer option for the Administrator operation on the Screen page.

Note that there is another option, Import a session. If a Host On-Demand session has been previously exported, the session definitions can be imported as a session object here. Personal Communications Version 4.1 and later workstation profiles can be imported as well.

Customization that existed in the imported session does not become part of the new session object. It must be re-established.

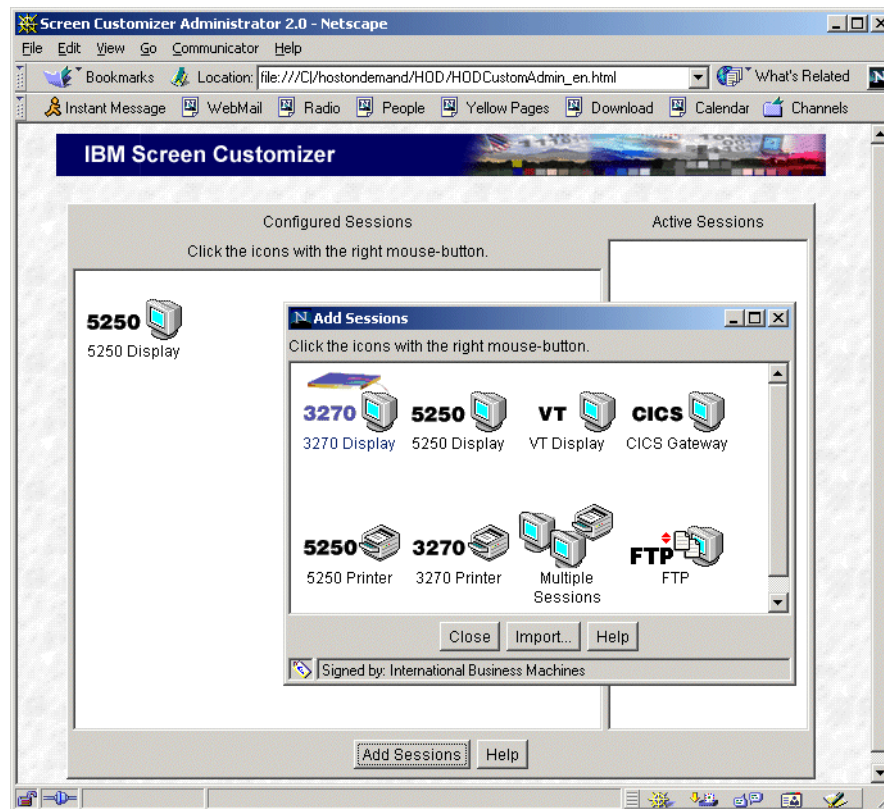


Figure 28-7 Defining session in Screen Customizer Administrator

Note: If Host On-Demand is using an LDAP directory for user management, you must create the following user when you install Screen Customizer; otherwise, configuration information cannot be written to or read from the LDAP directory:

user ID: customadmin
password: password

Alternatively, you can specify a different user ID and password by adding the following parameters to HODCustomAdmin(_[language]).html:

```
<Param Name=User          Value="[userid]">  
<Param Name=Password      Value="[password]">
```

Capturing a screen for customization

One of the primary purposes of the Screen Customizer Administrator is to capture screens and turn them into maps for the Customization Studio. Screen Customizer creates a screen signature based on the screen structure, allowing you to set up a unique ID for each host screen. Without a unique ID, you cannot customize a screen.

Screen Customizer recognizes host screens by the following methods.

Screen structure

- ▶ Size and positions of all fields on the screen
- ▶ Total number of fields on the screen
- ▶ Additional criteria used to build the screen ID

Tags

- ▶ Differentiates between screens that have the identical structure by interrogating the contents of a screen field that is designated as the tag
- ▶ Allows unique screen IDs to be assigned for different modes of the same screen

Built-In screen ID

- ▶ Some host applications have a system-assigned screen ID found within the presentation area that can be designated as the identifier.

To capture a screen for customization, do the following:

1. Load HODCustomAdmin.html from the file system. Alternatively, this can be done by clicking **Start -> Programs -> IBM Screen Customizer -> Administrator**.
2. Complete the Screen Customizer session configuration window with the appropriate host information, then launch the host session.

3. Log on to your host and go to the screen you want to customize.
4. In the session window, click **File -> Screen Properties**.
5. In the Setup Screen ID window, click the **Screen ID** tab, then type in a 4-character ID to identify the host screen. In the example shown in Figure 28-8, we used a screen ID of ca11. Optionally, you can also give the ID an 18-character name, which can be viewed in the View Screen IDs tab; you can also type an extended description of the ID which can only be viewed from the Screen ID tab. Other tabs available are:
 - Tag
Allows you to identify a screen by using a specific field and its contents on the screen. Tags are needed when creating screen IDs for different modes of the same screen.
 - Statistics
General information about the screen, including protected and unprotected fields.
 - View Screen IDs
You can browse through screen IDs that had been previously created, or delete them from the database.
6. Click **Apply** to apply the ID to your screen.
7. Click **OK** to exit the Setup Screen ID window and return to your host screen. Notice, in the lower-right corner of your screen, that your screen ID now appears indicating recognition of the screen.
8. With a screen ID assigned, the last task is to save the screen's presentation space content. Click **File -> Save Screen** if you want the name of the map to match the screen ID. Otherwise, click **File -> Save Screen As...** and assign a unique name.

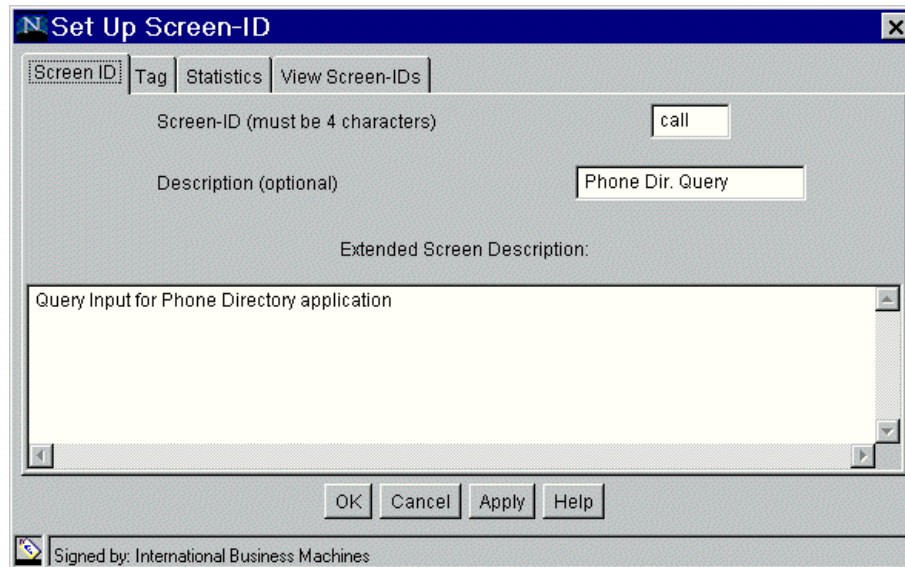


Figure 28-8 Screen Customizer Screen ID window

You are now ready to customize your screen by means of the Customization Studio, so click **File -> Customization**. This launches the Customization Studio displaying the screen you want to customize; however, it is better to save each as a map file and customize them later.

Tip: In certain instances, tags are used in a reciprocal context from that described in the steps above. Two of these instances are common in 5250 display sessions: error message handling and the opening of AS/400 application subfiles.

In these cases, the appearance of an error message or the opening of subfiles causes detection of a change in the appearance of the screen by the recognition class of Screen Customizer. Therefore, to completely customize the application, each of these screens must be assigned a screen ID, and then be customized. This is obviously undesirable, given the number of screens involved.

Instead, tags for these screens should identify a screen element that remains unchanged by the appearance of these items. Consequently, when the new screen items are sent, the customized map would just have the text of the error message or subfile appear as part of the custom screen.

Screen Customizer: a closer look

To illustrate some of the features in Screen Customizer Version 2, let's use an IBM business system login screen, shown in Figure 28-9, in its original emulator format.

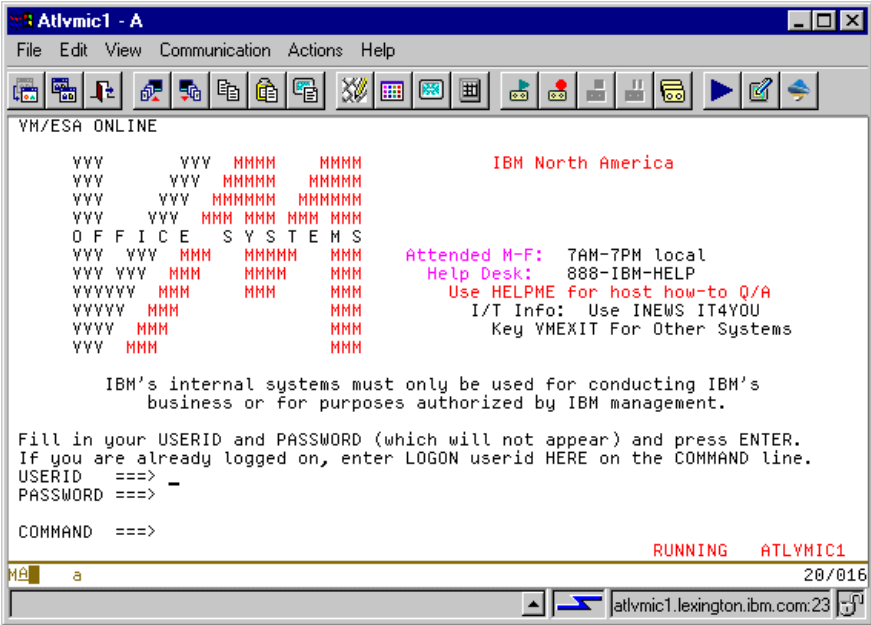


Figure 28-9 3270 screen in native mode

When this same session is loaded via Screen Customizer Administrator, it will first appear much like the Host On-Demand default graphical interface. However, the Administrator has an additional toolbar, as shown in Figure 28-10.

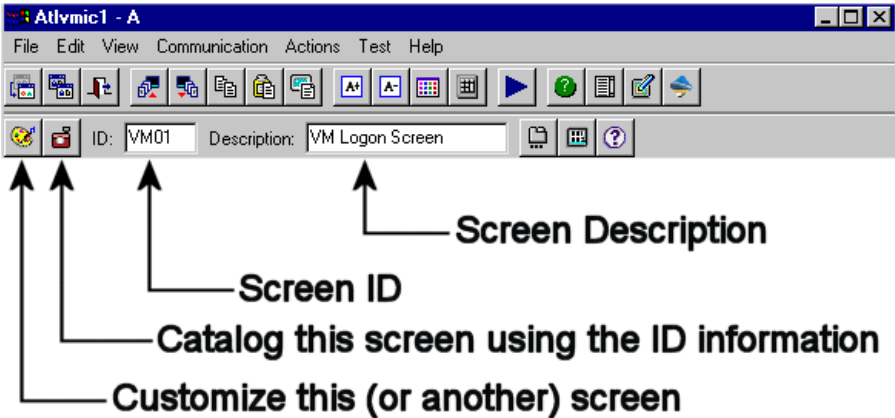


Figure 28-10 Screen Customizer Administrative toolbar

Notice the addition of the administrator's toolbar beneath the regular Screen Customizer toolbar. When working on a Screen Customizer application, the first tools we'll be interested in are the first four items (from left to right):

1. The Customize the current screen tool, represented by the icon that looks like an artist's palette in Figure 28-10.
2. The Capture the current screen tool, represented by the icon that looks like a camera in Figure 28-10.
3. The four-character Screen ID entry field, which should be familiar to Screen Customizer Version 1 users.
4. A Screen description, a field which should also be familiar to Screen Customizer Version users. Just as with the previous products, this field is still optional but nevertheless recommended.

Cataloging a screen is much easier with the new toolbar. As long as a screen ID has been entered, the administrator has three options to catalog a screen:

1. The Capture tool (the camera icon). If this option is used, then the screen is simply cataloged and the administrator can continue with the host application.
2. The Customize tool (the palette icon). If this tool is used, then the screen is cataloged and Screen Customizer Studio is launched to tailor the screen.
3. The Screen ID properties tool (the icon to the right of the screen description field). If this tool is used, Screen Customizer brings up a window (similar to that in Version 1) used to alter the way the screen is recognized. A screen tag can be used in place of the default screen recognition mechanism.

Note: With any of these tools, the optional description can be entered, as shown in Figure 28-10, and will be cataloged with the screen ID. It is recommended that you enter a screen description.

Once the administrator has cataloged several screens, there are other operations that can be performed from the new toolbar.

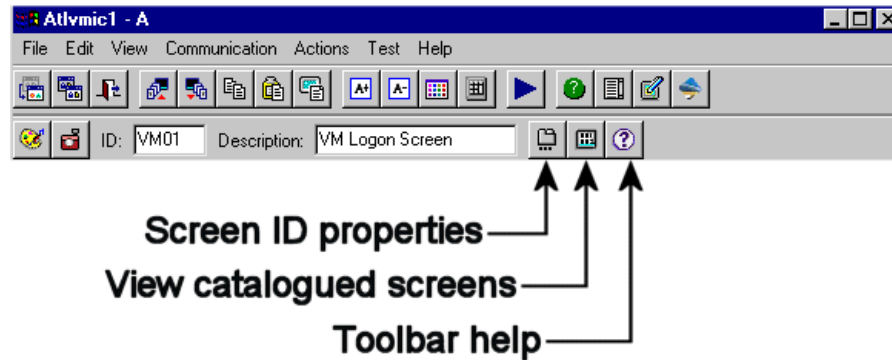


Figure 28-11 Working with cataloged screens

As we can see from Figure 28-11, the new administrator's toolbar offers more "one-click" ways to get to common activities:

- ▶ Setting up the screen ID
- ▶ Viewing all the screen IDs that have been cataloged
- ▶ Getting help for the administrative toolbar
- ▶ Bringing up the Customization Studio for other work (for example, creating/altering a template)

Merging screen databases

Typically there will be more than one person involved in the capturing of screens for customization. This could take place on one or several machines.

Regardless of whether screens have been captured from one or more applications, there can only be one screen database file in the maps directory that is downloaded to clients to enable presentation of the customized maps. A merge utility is provided as part of the Administrator to control this process.

The following explains the process of merging screen databases.

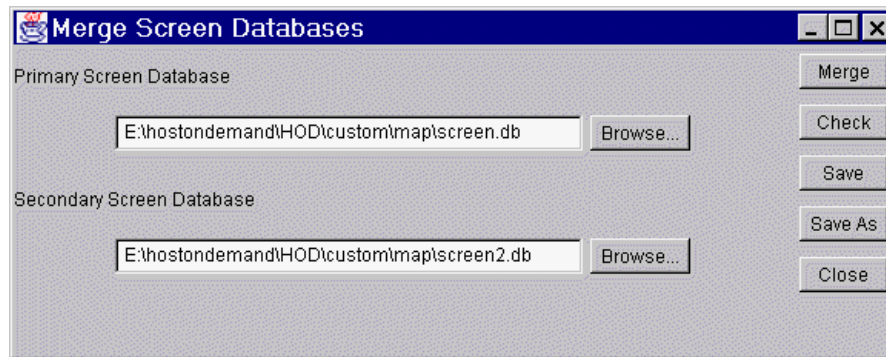


Figure 28-12 Selecting the screen database to merge

Figure 28-12 shows the selection of databases to be merged. In addition to the screen database files themselves, the maps that are referenced by screen ID within them must also be present in a common directory.

When the two files are merged, comparisons are made to ensure that maps will not be duplicated in the merged version. To see the results of the comparison, you would click **Check** on the Merge Screen Databases window.

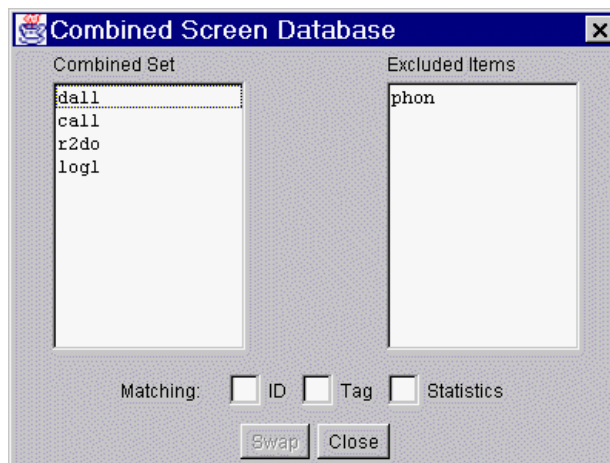


Figure 28-13 Working with the merged screen database

Figure 28-13 shows the comparison results for the two database files. There was a duplicate map named phon found in the screen2.db file; by default, duplicates that are members of the designated primary screen database take precedence over those of the secondary during the merge process.

Selecting the map phon in the Excluded Items pane allows you to see which map of the primary database it duplicates, and the items (ID, Tag, or Statistics) that matched. You can choose to swap the maps for the combined database if desired. Once the merged contents are finalized, the database is saved in the directory of the primary.

28.7.2 Using the Customization Studio

The Studio must be installed and operated locally on a Win32 workstation; it cannot be downloaded from a Web server. It does not require a host connection, unless the person using the Studio would like to test his or her screen in real time using an Administrator mode session.

With the Customization Studio, you can perform the following tasks:

- ▶ Save a screen as a map
- ▶ Work with fields - hide, modify, move
- ▶ Change the screen colors
- ▶ Convert a text field to a choice field
- ▶ Modify host function keys (F1 - F12)
- ▶ Add an image to the screen
- ▶ Create a button
- ▶ Create a Web link button
- ▶ Create a label field
- ▶ Fine-tune the screen so everything is precisely positioned
- ▶ Automate screen inputs to control application navigation

Starting the Studio

You can start the Studio by either of the following methods:

- ▶ Open CustomStudio.html from the file system by clicking **Start -> Programs -> IBM Screen Customizer -> Customization Studio**; or click **CustomStudio.html** in the Windows Explorer.
- ▶ Click **File -> Customization** on the screen menu while the screen is displayed in an Administrator mode session.

Saving a screen as a map

When the Customization Studio appears, with your host screen ready to modify, you will see at the top of the screen, above the menu bar the suggested path and file name for the map file of the screen ID that you saved with the Administrator. If you have not yet saved the map file, it is a good idea to do so before customizing

the screen because this lets you come back to it later. For example, the screen ID used in the Administrator was call, so to save the map in the Customization Studio, click **File -> Save As**, ensure that the path is correct, then save the file as call.scm. In this case, the correct path is:

```
d:\hostondemand\HOD\custom\maps\call.scm
```

Tip: If you are running with live sessions, keep the Administrator mode session running while you use the Customization Studio. In the Customization Studio, first save the modifications to the screen you are working on, then switch over to the Administrator mode session (with the same screen displayed) and click **Actions -> Refresh** to see your resulting changes.

Keep in mind that certain customization objects are not visible using this technique; for example, images can only be seen in live client sessions.

Working with fields

To display all fields on the host screen, click **View -> Fields**. At this point, you can click any field that you want to work with; a left-click selects the field, a right-click displays its properties.

Hiding fields

To hide a field, left-click to select it, then click **Edit -> Hide** (or press the spacebar). With the empty fields out of the way, you can use the same process to hide any other fields that your users don't really need or that you do not want them to see. Notice that when a field is selected, its properties are displayed at the bottom of the Customization Studio screen: the type of field (for example, text, button, label), its X and Y coordinates on the screen, and its width and height in pixels.

Note: You can hide but not delete text fields, buttons, or other objects that come from the host. You can delete objects that you have created in Screen Customizer.

Modifying fields

Refer to Figure 28-14 as we describe how to modify a field.

100 Search Request

To request a search, type the search data and press Enter.

Lines to 22 of 32

Corporate and Personal Directories

Name

Search Locs/Nodes/Dirs .

Node

User ID

Extension

Job Responsibilities . .

Department

any field (Name Value) . ex: DIV 1

*any field (Name Value) . ex: MGR Y

Departments Directory

Department Title/Number .

Search Locs/Nodes/Dirs .

Distribution Lists

List Name

Services Directory

Service Keyword(s) . . . ex: =IBMUS

Search Locs/Nodes/Dirs . ex: MKTG

Facility (F1 gives help) ex: Dallas

City ex: CT

State

(C) Copyright IBM Corporation 1988, 1992. All rights reserved.

Command ==>

F1=Help F2=Set 2 F3=Exit F4=Profile F5=Refresh F6=Fuzzy search

F7=Backward F8=Forward F9=Command F10=Actions F11=Dist list F12=Cancel

F1	F2	F3	F4	F5	F6	Enter	Reset	Clear	Insert	NewLine
F7	F8	F9	F10	F11	F12	Erlnp	ErFld	ErEOF	Delete	NextPad

Signed by: International Business Machines

Figure 28-14 Callup main screen

1. Select the field shown as **Name....**, then right-click it to bring up its properties window.
2. From the Set Label Properties window (see Figure 28-15) you can change the following properties:

Caption

To change the caption of a label, or to add a caption, type the new caption in the New field. In this case, we will replace Name..... with Whom are you looking for?

Font	To change the font, select the font, then its style and size. In this case, we will use the Helvetica font, with a Bold Italic style and a size of 12.
Color	To change colors, select foreground or background, then select a color from the palette. In this case, we will use a red foreground with a white background.
Appearance	To change the appearance, type a different value (in pixels) for the horizontal position, vertical position, width, or height. In this case, we want the width to be 175 and the height to be 22.
Read/Write	Allows a field to get or send its current value (in this case its caption) from or to any other field of any screen by using variables.

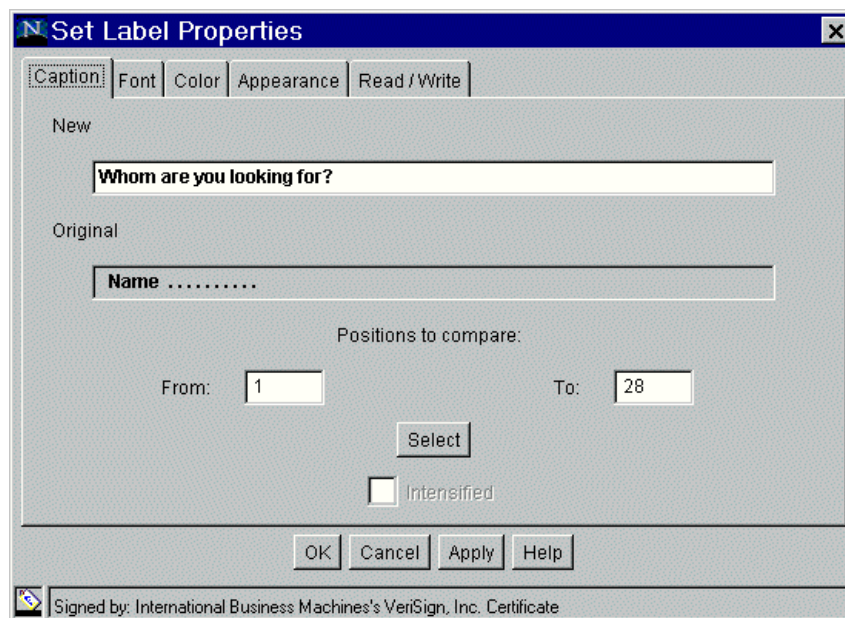


Figure 28-15 Set label properties window

Click **Apply** or **OK**.

Continue this process for the remaining label fields (Location, Node, Extension, Job, and Department) along with their corresponding input text fields (see Figure 28-19 on page 967).

Tip: When setting the caption for a label, if the content's dimensions exceed the dimensions specified on the Appearance tab page due to string length or the font in use, the background of the label will appear red until the label is resized to contain it.

If you move the Set Properties window so that the screen you are customizing is visible behind the properties window, you can click **Apply** and observe the changes before you close the window.

Moving fields

You can move a field by any of the following methods:

- ▶ Click the field you want to move.
- ▶ Press and hold Shift and use the arrow keys to move the field.

or

- ▶ Click the field you want to move.
- ▶ Move the mouse pointer to where you want the field to be.
- ▶ Press the M key.

or

- ▶ Click the field you want to move.
- ▶ Hold the left mouse button down while you move the mouse pointer to a new position on the screen.
- ▶ Let go of the mouse button to land the field.

Another tool that is useful for moving multiple objects is found by clicking **View -> Object** from the session window menu. The List of Objects window will show all objects on the screen, including those previously hidden. Multiple fields can be selected by holding down the Shift key while selecting them. They can be moved in increments as small as one pixel in any direction on the screen.

Tip: The properties of objects can be accessed in the List of Objects window by clicking the **Properties** button. If multiple objects are selected, the properties that are available to change are only those that are common to all the selected objects.

In addition, objects that are hidden on a screen map will have their coordinates and dimensions appear in parentheses in the List of Objects window. This makes it easier to identify them quickly when there are many objects present.

Changing the color of the screen

To change the color of the screen, click **Screen -> Colors** or click the color object (rainbow) on the toolbar. In this case, we have decided to use a white background color for the screen.

Converting a text field to a choice box

The text field is the default object that Screen Customizer assigns to an unprotected field from the host. If there is a text field in which users really only have a limited number of input choices, it is a good idea to replace it with a choice box. In this case, we want to change the phone directory application's Location field to a choice box. To do this, follow these steps:

1. Left-click the field you want to convert.
2. Click **Edit -> Convert -> to Choice**.
3. When the Choice field appears, click the right mouse button to bring up the Set Choice Properties window.
4. Click the **Settings** tab, then add the items you want to have in the Choice object. In this case, the locations we want the users to choose from are: RTP, USA, UK, GERMANY, EUROPE, JAPAN, ASIA, and AFRICA. After you have typed each entry in the Set Item field, click **Add**.
5. Click the **Font** tab and select the font you want. In this case, we chose the Courier font, with a style of Bold, and a size of 12.
6. Click the **Host Link** tab and ensure that input from your choice box is linked to the appropriate host input field; the correct link should already be highlighted. If you click other values, you will see the focus (blinking field) change on the customized screen behind the properties window.

7. Click the **Appearance** tab and set the appropriate appearance attributes. In this case, we want a width of 90 and a height of 23.

Note: The correct width and height for the box should be based on the width of the longest selectable entry and the font size in use by the object. If the width is set too small, a horizontal scroll bar will appear at the bottom of the box. The simplest means of setting these values correctly is by manual manipulation of the box border using the mouse. This behavior is also seen with list boxes when running within Internet Explorer 5.0.

8. Click **OK** to save and apply the changes.

Modifying host function keys

Although the host function keys at the bottom of the screen work by your clicking them, we want to remove the label field next to each button (Help, Exit, Profile...) and replace the Fx on each button with its actual function. This will involve enlarging the buttons, changing each button's caption and adding a macro to each button. To modify a button:

1. Start by hiding the label for each button; see "Working with fields" on page 954.
2. Next, change the F1 Help button. Left-click the button to select it, then right-click it to bring up the Set Button Properties window.
3. From the Set Button Properties window (Figure 28-16 on page 960) you can change the following properties of a button:

- **Caption**

To change the caption of a button, or to add a caption, type the new caption in the New field. In this case, we will replace F1 with Need Help?.

- **Font**

To change the font, select a font, then its style and size. In this case, we will use the defaults (Courier, Regular, and 12).

- **Macro**

To add or change a button's macro, type the text string that must be entered on the command line when the user clicks the button, link it to the appropriate host field, then select the appropriate function key. In this case, we need only to select the function key (F1) because the host application knows how to interpret F1 (see Figure 28-16 on page 960).

Notice in the figure that the Field Values and Link to Input Field or Parameter panes are blank. Only the F1 key is selected. This indicates that the button's function is screen specific; it is not driving any string input that would be field specific.

- Appearance

To change the appearance, type a different value for the horizontal position, vertical position, width, or height. In this case, we want the width to be 75 and the height to be 22.

- Read/Write

Allows a field to get or send its current value/caption from/to any other field of any screen.

4. Click **Apply** or **OK**.

Complete the same steps for the remaining function keys (F2 - F12).

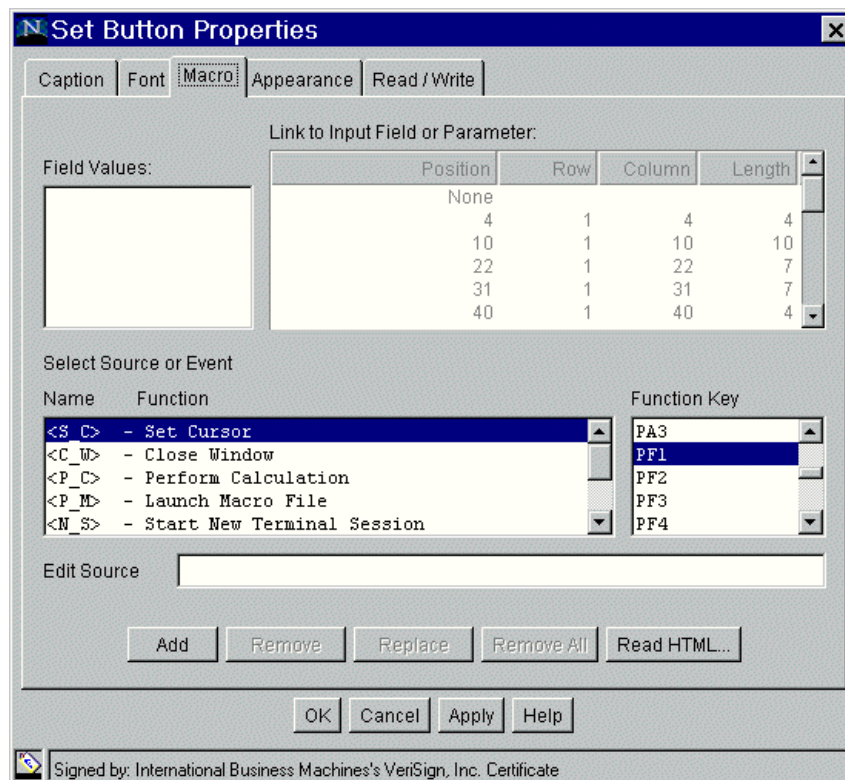


Figure 28-16 Set Button Properties window

Figure 28-17 on page 961 shows how the new graphical interface looks at this stage.

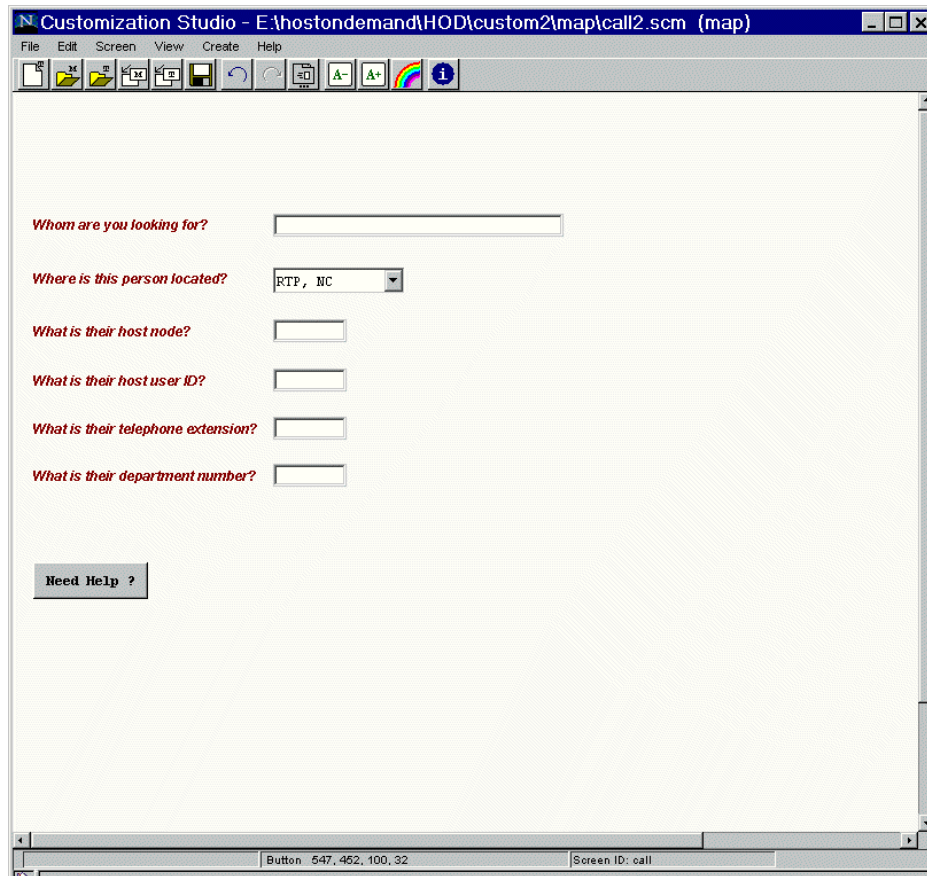


Figure 28-17 Customization Studio work in progress

Adding an image to the screen

To make the screen more interesting, we decided to put a banner, in the form of an image, at the top of the screen:

1. Copy the image file you want to use (.gif or .jpg) into the hostondemand\hod\custom\img directory.

Note: Screen Customizer only recognizes images with .GIF and .JPG extensions.

2. Click **Create -> Image**. When the Open File window appears, select the image file and click **Open**. The image becomes anchored to the mouse pointer.

3. Move the mouse to position the image, then left-click to land it.

Note: Objects created by Screen Customizer appear behind the existing host fields. The host fields must be hidden to make them visible.

4. You can resize the image by:
 - Clicking the image, then using the sizing handles that appear around the edges, or
 - Pressing the Ctrl key and using the arrows to resize it. Use the Up and Left arrows to reduce the image and the Down and Right arrows to enlarge it.

Adding buttons to the screen

To save users a few key strokes, we will add a couple of buttons to the screen. The first button is merely an Enter key in disguise and the second allows the user to log off the host system from this screen. To add a button to the screen:

1. Click **Create -> Button**. A button appears, anchored to the mouse pointer.
2. Move the mouse to position the button, then left-click to land it.
3. Click the right mouse button to display the Set Button Properties window. You can set or change the following properties:

- Caption

To add a caption to the button, type the caption in the New field. In this case, the caption will be Logoff System.

- Font

To change the font, select a font, then its style and size. In this case, we will use the defaults.

- Macro

To add a macro to your button, type the text string, link it to the appropriate host field, then select the appropriate function key. In this case, we want to use logoff as the string and have it entered on the command line, then have the Enter key sent to the host.

Type logoff in the Edit Source field, click **Add**. Select **logoff** in the Field Values pane to select it; then, in the Link to Host Field list or the Parameter list, select the appropriate host input field. If you're not sure which input field to select, move the properties window so that your customized screen is visible in the background. Now you can click the available parameters in the properties window and watch the focus change from field to field on the customized screen until the appropriate host field is selected. After you have selected the correct host field, select the appropriate function key, in this case **Enter**.

- Appearance

To change the appearance, type a different value for the horizontal position, vertical position, width, or height. In this case, we want the width to be 75 and the height to be 22.

4. Click **OK** to save and apply.

The same process is followed to add a Start Search button.

Creating a Web link button

Because some people prefer to use a Web-based version of this host program, we will add a Web link button that automatically enters the URL of the Web page into a browser. To add a Web link button to the screen:

1. Click **Create -> Web Link Button**. A button appears, anchored to the mouse pointer.

2. Move the mouse to position the button, then left-click to land it.

3. Click the right mouse button to bring up the properties window. From the Set Web Link Properties window, you can change the following properties of a Web link button:

- General

To add a caption, type a caption in the Enter Caption field. To change the position of the caption in the button, select Top, Center, or Bottom under Position Caption. In this case, Blue Pages will be the caption, which will be in the center of the button. We could also have set a status bar tip.

- Font

Select a font, then its style and size.

- Color

To change colors, select foreground or background, then select a color from the palette. In this case, we will use a blue foreground with a white background.

- Macro

To add a Web link, type the URL of the link in the Edit Source field, then click **Add**. In this case the URL is `http://w3.ibm.com/bluepages?`. No link to a host field is required.

- Appearance

To change the appearance, type a different value for the horizontal position, vertical position, width, or height. In this case, we want the width to be 80 and the height to be 22.

- Read/Write

Allows a field to get or send its current value/caption from/to any other field of any screen.

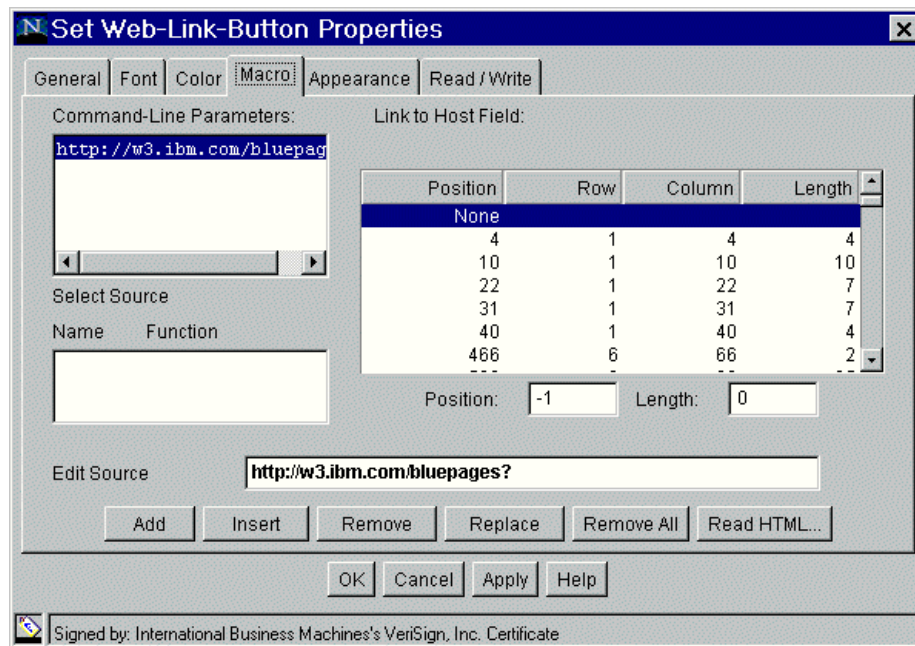


Figure 28-18 Web link button macro

- Click **OK** to save and apply.

Note: Web link buttons do not work in Administrator mode sessions. To test your button, you must run the Screen Customizer Client.

Creating a label

Since we have created a Web link button, we would like to add some text to the window to tell users what the button is for. To create a label:

- Click **Create -> Label**. A label appears, anchored to the mouse pointer.
- Move the mouse to position the label, then left-click to land it.
- Click the right mouse button to bring up the properties window. You can change the following properties:

- **Caption**

To add a caption to a label, type a caption in the New field. In this case, the caption will be:

If you cannot locate the person here, try the IBM Blue Pages >>>

- Font

Change the font if you want.

- Color

Change the colors if you want.

- Appearance

Change the appearance if you want. In this case, we want the width to be 375 and the height to be 22.

4. Click **Apply** or **OK**.

Fine tuning

Now it is time to tidy up the screen a bit; let's align and size the objects.

► Aligning fields and buttons

There is more than one way to get fields and buttons to line up vertically and horizontally on the window.

- Visually

You can align buttons and fields visually by just dragging them with the mouse, then paying close attention to the properties displayed at the bottom of the window. You can then fine-tune them by selecting them, and then using the Shift key and the arrow keys to position them more precisely.

- Set Properties window

If you right-click a field or button, then click the **Appearance** tab, the horizontal, vertical, width, and height properties are displayed; you can then type in the exact values you want. For example, if you want a certain group of fields all to start at the same horizontal position and to be vertically aligned, you can use the same value for the horizontal position and then calculate equal spacing to ascertain each label's vertical position.

- View objects

If you click **View -> Objects**, a list of objects for that screen will be displayed. From here, you can select a group of objects, then choose to align them up, down, right, or left. In addition, you can align a group of objects to the one that is the active (last selected) object.

► Sizing fields, buttons, and objects

To get fields and buttons sized the way you want them:

- Visually

You can size fields and buttons visually by just clicking them with the mouse, then holding down the Ctrl key and pressing the arrow keys to increase or decrease the size. Alternatively, you can click them, then run the pointer along the edge until a double arrow appears, then hold down the left mouse button and drag until you get the desired size.

- Set Properties window

Right-click the field or button, then click the **Appearance** tab; the horizontal, vertical, width, and height properties for the object are displayed. Type in the values you want.

- View objects

Click **View -> Objects**; a list of objects for that screen will be displayed. Select a group of fields or buttons, then resize them all to the same size as the one that is active.

With these steps taken, our completely customized screen would appear as shown in Figure 28-19.

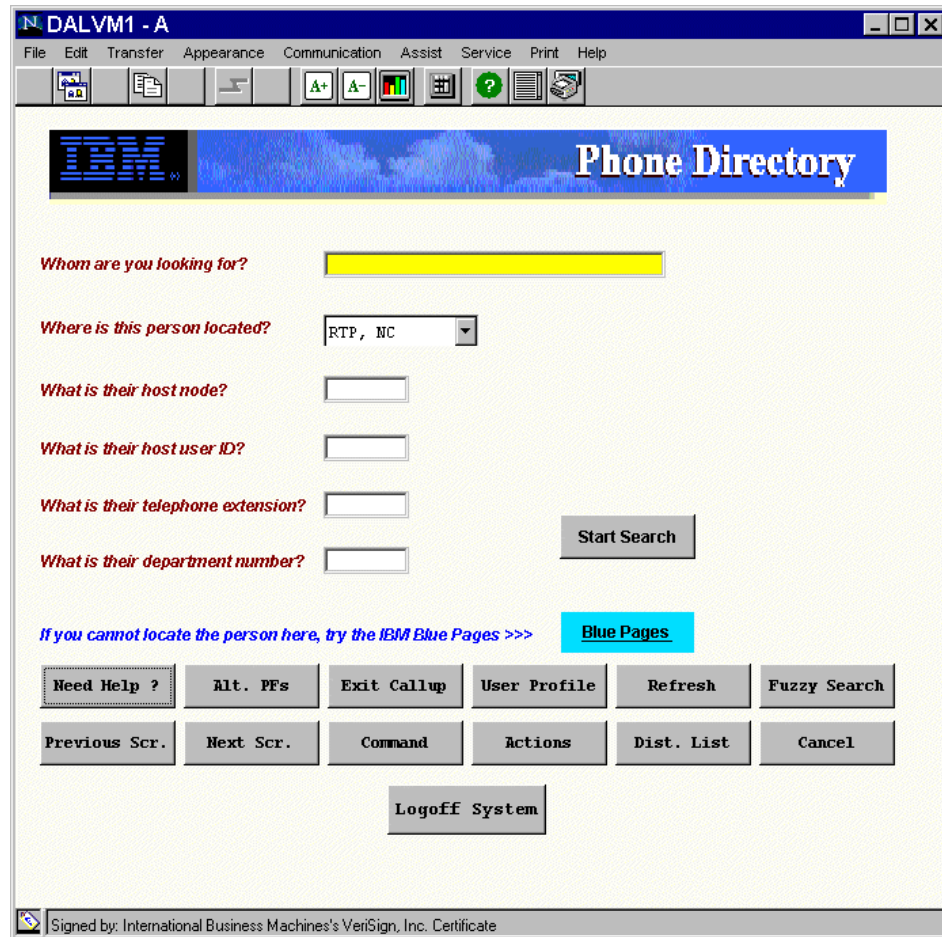


Figure 28-19 Screen customized by the Customization Studio

Other macro button functions

The Web link button is an example of the most common use for macros: to pass one or more strings of data to fields on a screen. There are other macro functions available in Screen Customizer for use with buttons.

► Close window

This predefined macro command in the Select Source/Event list (<C_W>) is used to free a connection by closing the session window and forcing a logoff from the system. It is most useful when application exit function keys only disconnect the console but do not free the session.

► New session

This macro command in the Select Source/Event list (<N_S>) starts a new session using the same configuration as the session instance in use.

► Run applet

This macro command allows a user-written class to be executed. The Run User Applet (<R_A>) item is selected from the Select Source/Event list.

Classes that are run using the Run User Applet command must implement a default constructor and the com.ibm.eNetwork.ECL.ECLAppletInterface. The interface definition can be found in the Host Access Class Library (HACL) reference of either Host On-Demand or Personal Communications.

► File transfer

There are three predefined commands for file transfer: (<S_F>) to send a file, (<R_F>) to receive a file, and (<F_D>) to set the transfer defaults. When any of these are used, the corresponding window to provide the parameters appears.

While the transfer defaults function is available, it is best to set the defaults for the session in the HOD or PCOMM session configuration itself. This will prevent users from altering the correct values.

► **Run macro file**

Macro command files can be created with an editor and used to send consecutive key stroke sequences to a session. There is no provision for linking the key strokes to a host field, so these macros must be screen-associated only.

Each line of the macro contains only one key stroke string or function key per line. The formatting of the line is <function key>* or keystroke string*. There can be no extra spaces in the lines, and the asterisk (*) must be included as shown.

You can use the following function keys: <Alt-f>, <Alt-r>, <ATTN>, <Backspace>, <Clear>, <Ctrl-Delete>, <Ctrl-F10>, <CursorDown>, <CursorLeft>, <CursorRight>, <CursorUp>, <Delete>, <End>, <Enter>, <ErsEOF>, <ErsInp>, <F1> through <F24>, <Home>, <Insert>, <NewLn>, <PA1>, <PA3>, <PgUp>, <PgDn>, <Reset>, <Shift-Tab>, <SysReq> and <Tab>.

The file must be saved in the custom/map directory with a .kbt extension.

Global customization enhancements

Customizing screens is easier with the template enhancements. You can control the look and function of many different emulator screens at once by creating templates that can be automatically applied to screens without having to modify each screen individually.

Simplified screen capture process

The Administrator toolbar makes the process of capturing and customizing screens quick and easy. The toolbar buttons provide quick access to the functions you use every day when working with screens. There are buttons to capture a screen, start the Studio, and work with screen IDs.

Web link button improvements

Additional options have been added for Web link buttons. Text for links changes color when the mouse pointer is held over it, displaying a standard Web link. Settings can be saved for individual Web links.

Light pen support

Use your mouse as a light pen pointer when accessing host applications that require a light pen. Light pen fields can be displayed as check boxes or buttons, depending on the type of field. Refer to Figure 28-20 for a sample application that uses light pen support, but it is not enabled.

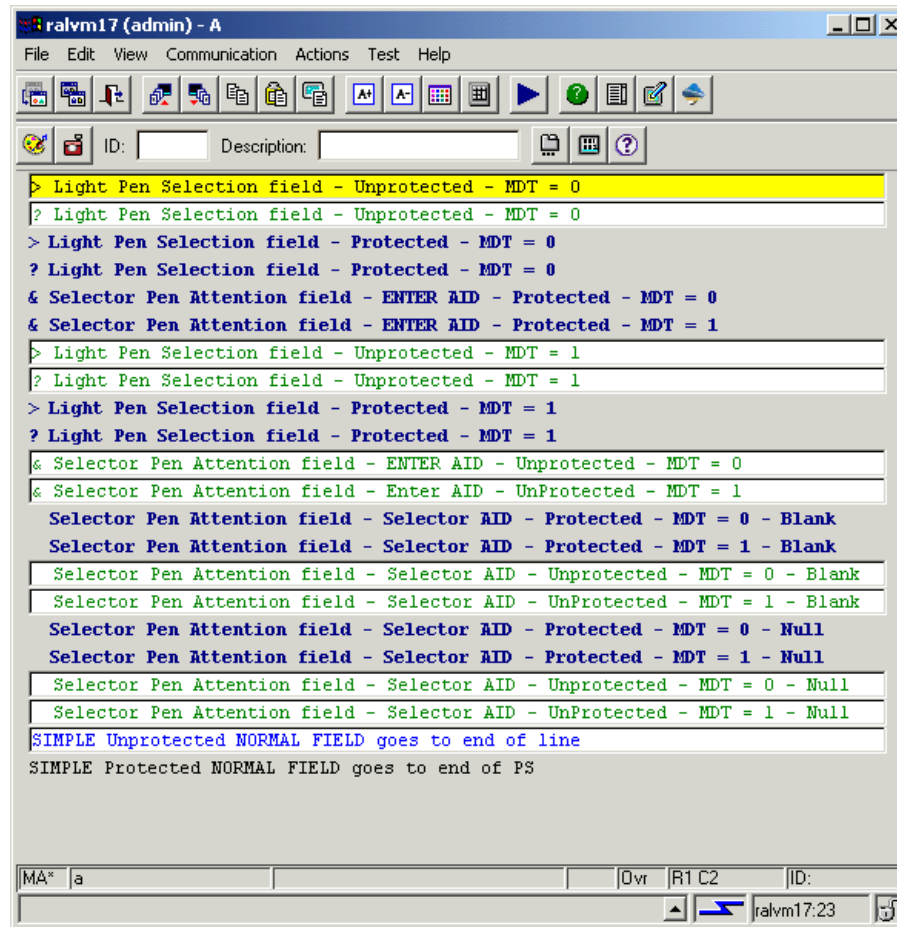


Figure 28-20 Light pen not enabled

The image shown in Figure 28-21 shows what the above window would look like with light pen support enabled.

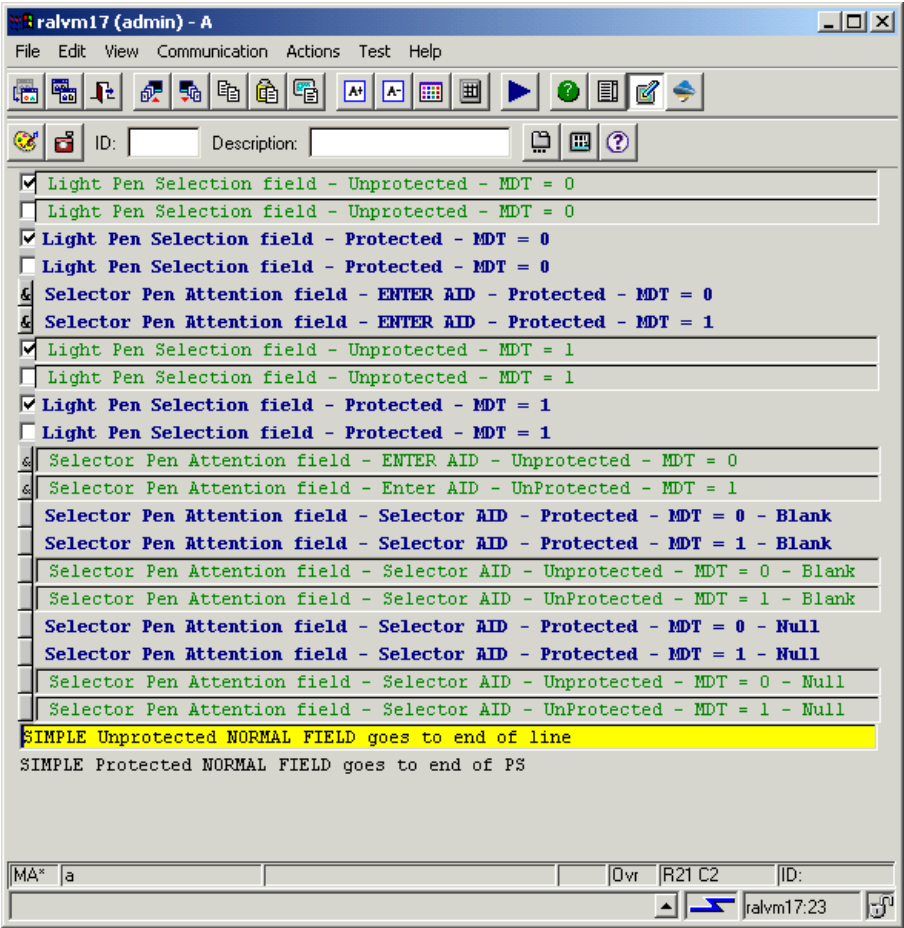


Figure 28-21 Light pen enabled window

Additional language support

Support for Hindi, Thai and Greek languages has been added.

AS/400 subfiles

AS/400 subfile support has been implemented to be consistent with the support provided by Client Access. When enabled, subfiles are automatically converted into multi-column tables with button hotspots that send the appropriate commands for manipulating objects in the subfile list. Figure 28-22 illustrates what a subfile would look like if subfile support were not enabled.

5250 Display - A

File Edit View Communication Actions Test Help

ID: Description:

WORK WITH FOLDERS

FOLDER /

POSITION TO STARTING CHARACTERS

TYPE OPTIONS (AND FOLDER), PRESS ENTER.

1=CREATE 3=NEXT LEVEL 4=DELETE 5=WORK WITH DOCUMENTS
7=RENAME 8=DETAILS 14=AUTHORITY

OPT	FOLDER	OPT	FOLDER	OPT	FOLDER	OPT	FOLDER
<input type="checkbox"/>		<input type="checkbox"/>	PRCTEST	<input type="checkbox"/>	QGA400RT	<input type="checkbox"/>	RENGAN
<input checked="" type="checkbox"/>	ALEX	<input type="checkbox"/>	QBKB00KS	<input type="checkbox"/>	QIWSADM	<input type="checkbox"/>	ROGERKG
<input type="checkbox"/>	BOB	<input type="checkbox"/>	QDIAD0CS	<input type="checkbox"/>	Q0TTMFLR	<input type="checkbox"/>	SHASHI
<input type="checkbox"/>	CINGLE	<input type="checkbox"/>	QFOEPRF	<input type="checkbox"/>	QPRFFLR	<input type="checkbox"/>	USEATEST
<input type="checkbox"/>	DEAN	<input type="checkbox"/>	QFOFWP2	<input type="checkbox"/>	QPRF2962	<input type="checkbox"/>	V& LOCKS
<input type="checkbox"/>	FRANKM	<input type="checkbox"/>	QFOFWP3	<input type="checkbox"/>	QWIN16	<input type="checkbox"/>	WINOPR
<input type="checkbox"/>	FRANKMA	<input type="checkbox"/>	QFOSDIA	<input type="checkbox"/>	QWPDOCS		
<input type="checkbox"/>	HODOPR	<input type="checkbox"/>	QFOS2962	<input type="checkbox"/>	RCASTRO		
<input type="checkbox"/>	KPERIAS	<input type="checkbox"/>	QFPBFLR.001	<input type="checkbox"/>	RCTEST		
<input type="checkbox"/>	KUMARP						

BOTTOM

F3=EXIT F5=REFRESH F6=PRINT LIST F9=WORK WITH
F11=DISPLAY DESCRIPTIONS F12=CANCEL F13=PREVIOUS LEVEL

MA* a MW Ovr R11 C2 ID: elcrtp11:23

Figure 28-22 Subfiles disabled

Once subfile support is enabled, the above window will appear as shown in Figure 28-23.

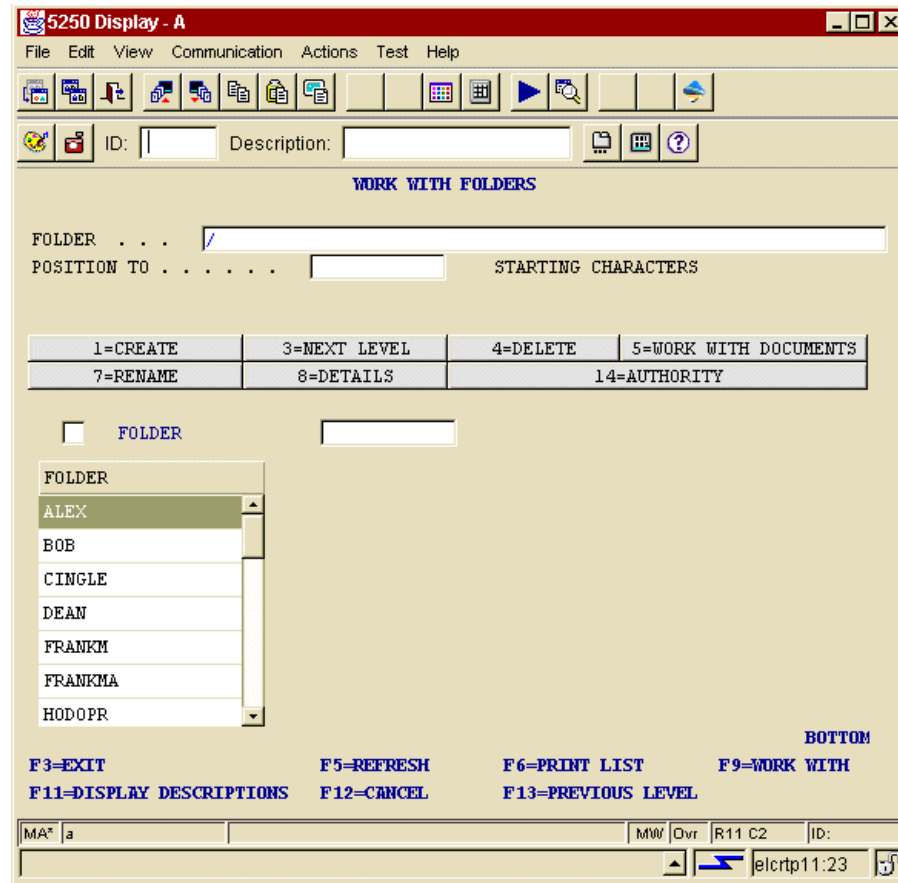


Figure 28-23 Subfile enabled

28.7.3 Template development

One of the most significant changes to Screen Customizer Version 2 is the introduction of the “template” feature. This feature allows you to provide customization defaults and features for large sets of screens without having to customize each screen. Templates can supply default colors, add customized objects to each screen’s toolbar, and even add Screen Customizer objects (such as a Web link) to every screen.

Templates are created in the Screen Customizer Studio and are saved in the same directory as the application’s screen maps (by default \custom\map). Screen Customizer templates have a file type of .tpl and otherwise can be named like other objects (for example, no spaces in the file name).

The global template

There is one special template, known as the global template. If you create a template and save it with a name of `sc_global.tpl` in the default map directory, it will be used by all screens (customized or not) unless you specifically override a particular screen's use of templates.

The global template can be overridden in the HTML using the `Template` parameter. It needs to be added in the HTML that starts the Screen Customizer Administrator, Screen Customizer client or any Host On-Demand client that has Screen Customizer enabled.

For example, in a full English-language installation of Screen Customizer, the Administrator is started by browsing an HTML file named `HODCustomAdmin_en.html`. Let's say we've created a template called `Ugly.tpl` (it's really ugly so we know it's working) and have modified the `HODCustomAdmin_en.html` file by adding the `Template` parameter. If we did, the resulting HTML would look like the sample in Example 28-1.

Example 28-1 Setting the template in the Administrators HTML

```
<PARAM NAME=BookmarkPage VALUE=AutoHOD_en.html>
<PARAM NAME=Admin          VALUE=true>
<PARAM NAME=Locale         VALUE=en_US>
<PARAM NAME=Template       VALUE=Ugly.tpl>

<p>If you are reading this message, your client platform is not capable of
running IBM Screen Customizer. To run IBM Screen Customizer, you must have a
Java-enabled web browser such as Netscape Navigator or Microsoft Internet
Explorer.
</APPLET>
```

There are other parameters related to template handling with different terminal sizes. It is possible to specify different default (global) templates for different terminal sizes. This is done by using the following Java parameters, in a similar way to how the `Template` parameter is used in other files.

Table 28-2 Screen size template parameters

Parameter Names	Screen Size
Templates	All unspecified
template 24x80	Model 2 (24 rows, 80 columns)
template32x80	Model 3 (32 rows, 80 columns)
template43x80	Model 4 (43 rows, 80 columns)
template27x132	Model 5 (27 rows, 132 columns)

You may or may not want to specify custom templates for each screen size, but you can use the screen size context menu's "snap to" function to test your template with different screen sizes. See Figure 28-24 on page 977 to see what this menu looks like.

Template hierarchy

When using Screen Customizer templates, it's very important to understand the hierarchy of customization that results in a particular screen's appearance. The template is really the "court of last resort" when it comes to a screen's appearance. We've already seen how a magic template name can override all screens in an application. But that template is overridden if you specify the Template parameter in the HTML. So what happens next?

A screen's appearance is determined by the following:

1. The default global template (sc_global.tpl, if it is present).
2. Any global template specified in the HTML (see Figure 28-24) and if specified will override sc_global.tpl.
3. If a template was specified in the Studio when customizing an individual screen by clicking **Screen ->Template Options...**, then that template will be in effect. This is referred to as a "map-specified template." The template name to be used with the map is actually stored within the map file.
4. User preferences have lowest precedence. For instance, let's consider background color. If the user specifies the background to be blue, then it will be so only if the maps/templates also specify blue, or if they specify that colors are to be inherited (using the Inherit Color option on the color windows). Maps would inherit from templates, which would inherit from the user preferences, which would be blue.

Note: If you are customizing a screen and want to disable all template effects, click **Test ->View with Different Template...** to turn off the use of templates altogether.

Developing a Screen Customizer template

To build a Screen Customizer template, you can either start directly from the Studio (the default is to bring up a blank template) or from a Studio session brought up from the Administrator. If you're customizing a host screen, just use the File pull-down and choose the **New Template** option, or use the CTL+N key shortcut.

When you start with a fresh template, you'll see a layout like that in Figure 28-24 on page 977. There are several areas on this page:

1. A large area towards the upper left-hand corner of the screen reserved for the host session. This is labeled "Host Screen Area" in Figure 28-24 on page 977.

Note: You cannot customize the host screen section of the template. This is done by customizing individual screens with the Screen Customizer Studio. When creating a template, you will be allowed to place objects in this area, but they will be overlaid by any objects (default or customized) at that position in the screen area.

2. The rest of the screen area is a palette for you to control as you please. It is labeled Template Customization Area in Figure 28-24 on page 977. A template can be customized much like any host screen except for these attributes:
 - Get-to-the-Point settings
 - Global Variable extensions
 - Template options
 - Tab order

The objects on the template (for example, a button) are considered outside of the screen's objects and therefore cannot be part of its tab order.

 - Tab-key controls
3. You'll also have a context (pop-up) menu that can snap the host area of the template to various host screen sizes. This is labeled Screen Size Context Menu in Figure 28-24 on page 977 and the actual menu of screen sizes is shown.

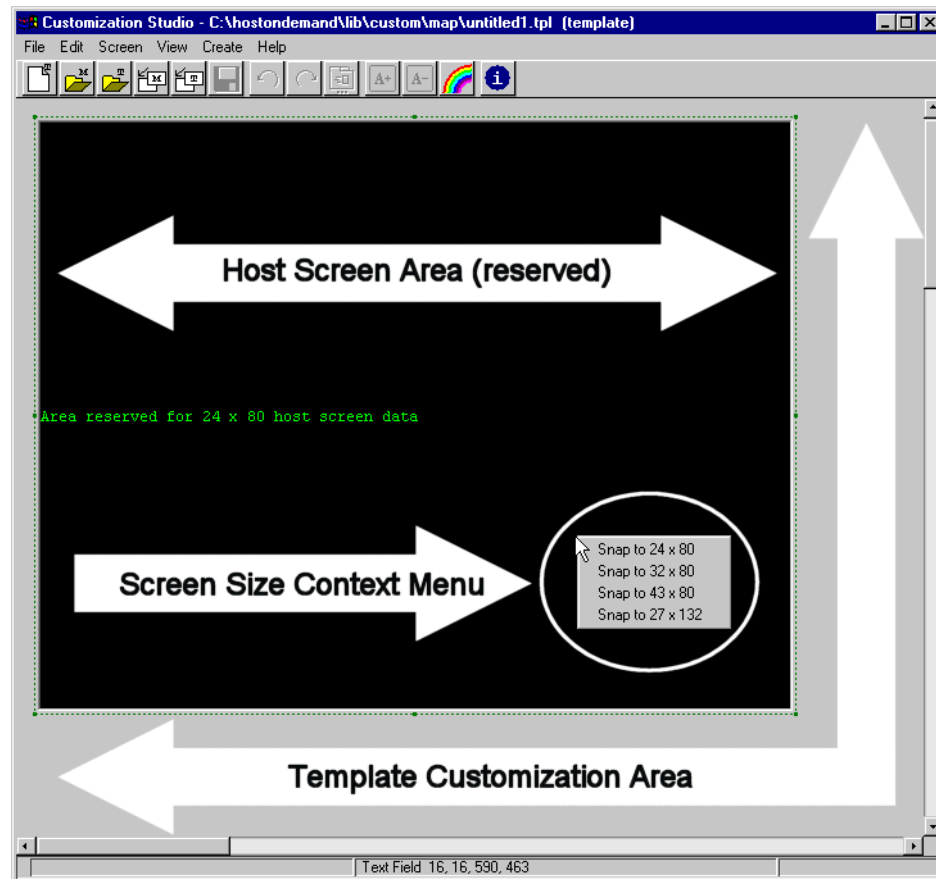


Figure 28-24 Creating a new template

It's important to note that although you can't customize the host area on the template directly, you can move and resize the area. By default, using the left-hand mouse button will turn on the "screen mover" cursor and dragging the screen area while holding this button down will move it. Figure 28-25 on page 978 shows a default template that has been moved down and to the right, allowing room for more customization.

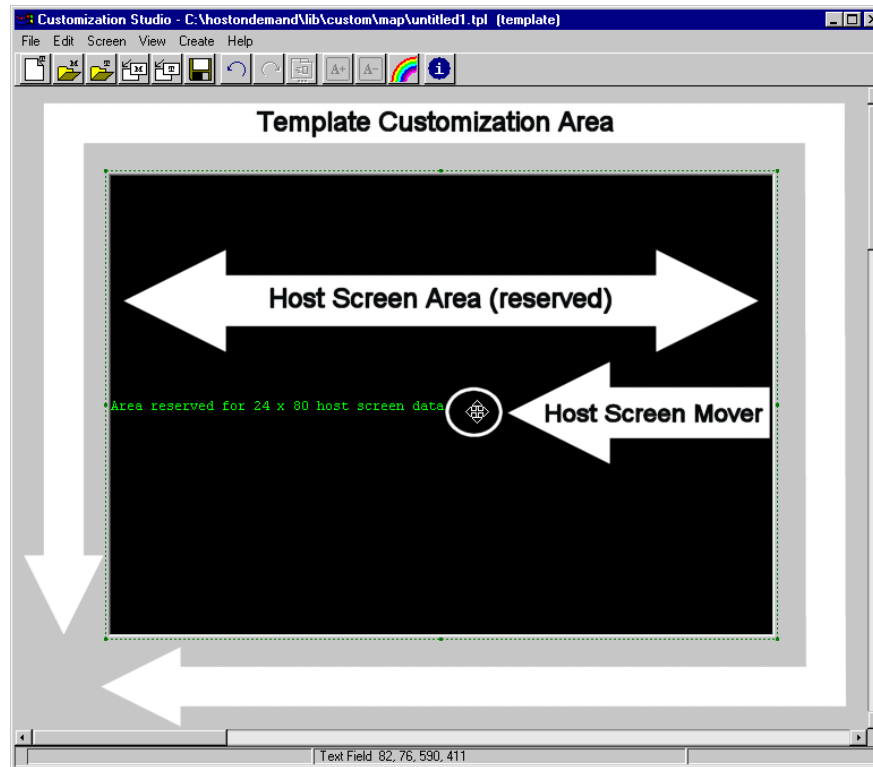


Figure 28-25 Moving the host area on a template

Moving the host area is allowed (even encouraged) since it allows for such activities as placing a banner image at the top of every customized screen that uses that template. This is a much more robust replacement for the Add Logo function in Screen Customizer Version 1.

28.7.4 Deployment

Deploying an IBM Screen Customizer application takes planning.

Screen Customizer objects

In order to understand how to coordinate development, testing and deployment of a Screen Customizer application, it is important to understand what components make this application. Strictly speaking, a Screen Customizer application is simply a set of files in a well-defined directory tree that uses naming standards understood by the Screen Customizer development and runtime applications.

With the exception of the stand-alone installation of the Studio, Screen Customizer depends on Host On-Demand and will be installed within the directory structure created by installing that product. When a Screen Customizer application is developed or deployed, it is normally stored in the \custom directory in the main Host On-Demand directory tree by default. Within the \custom directory, there is a well-defined structure of directories and files that looks like this:

Table 28-3 Screen customizer directory and object structures

Directory	Contents
\map	Contains the screen.db (screen database) file, all screen maps (.scm files) and template files (.tpl files)
\img	Contains all graphics used by the application, restricted to GIF (.gif) and JPEG (.jpg) files
\lst	Valid values list
\ps	Base screen data
\ref	Reference files for field help and valid-value list
\wsp	Global customizations
\en (and \en\help)	Help information

In the application development environment, this directory tree is located in the %HOSTONDEMAND%\LIB\CUSTOM folder (where %HOSTONDEMAND% is the root directory where the local copy of Host On-Demand was installed). And, by default, all objects created on that machine will be created in the \custom directory tree as illustrated above. For the location of customized files for OS/390, refer to Table 28-1 on page 939.

Testing the application

Testing a Screen Customizer application is important for two reasons. First, it's important that all of the components work as intended in the environment in which they will be deployed in. Second, it is important that some usability testing be done so that the application truly meets the user's needs. Since a Screen Customizer application is deployed on a Web server, it lends itself very well to iterative testing with a small pilot group of motivated users. Changes can literally be made "on the fly" according to user feedback and this has been done in actual practice.

The mechanics of Screen Customizer testing can be done on many levels:

- ▶ The Administrator's workstation
- ▶ A stand-alone client

- ▶ A test server
- ▶ A test server with the Administrator and Studio code installed

Each scenario has its unique needs. We won't cover these in great detail but the common links are:

1. Understanding of the file and directory structure that makes up a Screen Customizer application
2. How to use the Screen Customizer HTML parameters (specifically the subdir and template parameters) in server-based test scenarios

Let's start with the first line of defense: testing on the developer's workstation. That's pretty simple, since all you must do is reconfigure your session from "Administrator" to "Client" (see Figure 28-26) and run the session.

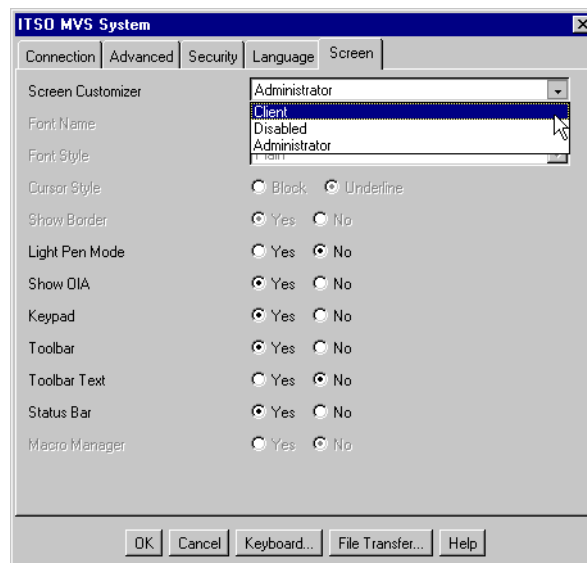


Figure 28-26 Reconfiguring a Screen Customizer client for re-testing

Next, it may be useful to test on a stand-alone client. It could be for a "reality check" because the user in question knows the host application very well or could be simply a tentative first step before pilot deployment.

Either way, you must package up the files from your /custom directory and get them to the client. It's probably best to use a tool such as WinZip to package the files, possibly as a self-unpacking file.

Preparing the application for deployment

When the Screen Customizer application is ready for productive use, it can be deployed in several ways, depending on how your Host On-Demand users are administered.

By default all Screen Customizer data must be put in one place, the \custom directory below the main \hod alias. Doing this will require that you:

1. Download the current screen.db file from the \custom directory.
2. Use the MergeDB process to combine it with the screen.db created during the creation of the new application.
3. Upload the new screen.db file and all the collateral files (individual screen maps and graphics) to their respective directories in the \custom directory tree.

You can provide different screen views of the same application to different groups of users, such as a call center, executives, or an extranet application. Since each group will be using the same applications, each group of users must have a different set of maps; therefore, they must retrieve their maps from a different subdirectory. To do this you must modify the launching HTML page and add the `subdir` (which was available in Screen Customizer Version 1) and/or `template` parameters (see “The global template” on page 974). Thus when a user connects, all will appear normal except that the screen maps for Screen Customizer enabled sessions will use the parameters specified in the HTML.

Tip: If deploying in an iSeries, Screen Customizer application developers have the choice of accessing and modifying the Screen Customizer maps and templates through a network drive; see 4.8.4, “Mapping a network drive to the iSeries” on page 162. The traditional FTP based deployment will also work.

28.8 Service Bundler

A new utility has been added to the IBM Screen Customizer, the Service Bundler. This utility is available on all platforms where IBM Screen Customizer IBM Screen Customizer is installed, whether a local installation or a server installation. The Service Bundler is run as a graphical interface on a Windows system and as a command-line utility on other platforms.

28.8.1 Windows system

On a Windows platform it is launched by clicking **Start -> Programs -> IBM Screen Customizer -> Utilities -> Service Bundler**. This launches an applet that displays the window shown in Figure 28-27 to collect the necessary information.

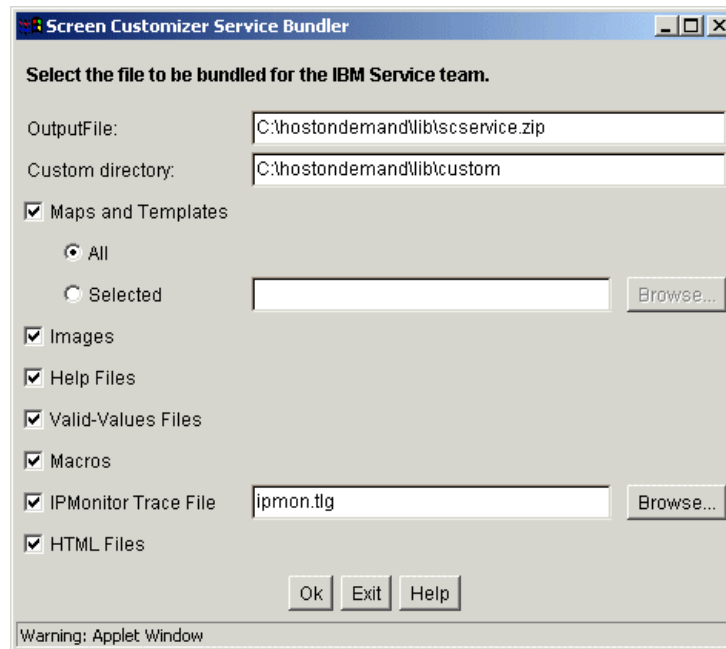


Figure 28-27 Screen Customizer Service Bundler

You must specify the following information:

- ▶ **Output File**
Specify the name of the output file without the extension. This is the file that you will send to IBM Service.
- ▶ **Custom Directory**
The default directory structure is specified by default. Select an alternate directory if you are using the subdir parameter.
- ▶ **Maps and Templates**
Check this box if you want to include maps and templates. If you select **Maps and Templates**, click either **All** or **Selected** to specify which maps and templates you want.

If you click **Selected**, do one of the following:

- Enter the name of the map or template
- Click the **Browse** button to select them

► Images

Check this box if you want to include image files.

► Help Files

Check this box if you want to include help files.

► Valid-Values Files

Check this box if you want to include valid-values files.

► Macros

Check this box if you want to include macros.

► IPMonitor trace file

Check this box if you want to include IPMonitor trace files. An IPMonitor trace file is generated when Service asks you to capture a Telnet data stream dump using the IPMonitor tool in Host On-Demand or Personal Communications. A Browse button is available to locate the desired file.

► HTML Files

Check this box if you want to include all the HTML files in the root (publish) directory (for example, HODCustomAdmin.html and HODCustomClientBasic.html).

28.8.2 Command-line interface

A command-line interface is fully documented in the *Getting Started Guide*. Make sure to refer to this document for any changes.

When invoking the Screen Customizer Service Bundler from a Host On-Demand server running on a non-Windows platform, invoke the following from a command line (the example must be entered on a single line) while in the Host On-Demand publish directory (usually, <hod-installpath>\HOD; a JRE is also required):

```
"\ <hod-installpath>..\bin\java -classpath .;<hod-installpath>\scbundle.jar;  
com.ibm.hi.customizer.util.bundler.SCBundler" <options>
```

The options are shown in Table 28-4

Table 28-4 Service bundler command line parameters

Parameters	Description
/? or /h	Print out the help message

Parameters	Description
/o filename	Specify the name of the output file (default = scservice.zip)
/d customDir	Specify an alternate custom directory name
/a	Include all file types, same as /m, /i, /p, /v, /c, /t, /f
/m	Include all maps and templates
/i	Include all images
/p	Include all field help files
/v	Include all valid-values files
/c	Include all macro files

The output will be placed into the service.zip file.

28.9 Application programming interface

IBM Screen Customizer Version 2 has been enhanced with the addition of a Custom Terminal Bean and a Screen Customizer Component Interface (SCCI). Together they constitute the IBM Screen Customizer API. This section provides only an overview of the capabilities of this API.

The IBM Screen Customizer API allows user code, written in Java, to interact with Host On-Demand and IBM Screen Customizer. The code may be run as a IBM Screen Customizer applet or stand-alone Java application. The custom applet can be started from one of the following:

- ▶ A button click
- ▶ Get-to-the-Point
- ▶ During startup of the session:
 - If the session was configured to launch an applet at startup
 - If the applet was specified to auto launch via an HTML parameter.

The IBM Screen Customizer programming interface requires the Host On-Demand Toolkit, and as such is only supported on the Windows environment.

28.9.1 Custom Terminal Bean

The Custom Terminal Bean is an extension of the Terminal Bean designed to closely interact with Screen Customizer. It encapsulates all Terminal Bean functionality. It allows users to programmatically interact with IBM Screen Customizer and to set/get current settings, such as font properties, code page, host, HTML parameters (such as customURL), current graphical interface components, etc. or invoke functions such as print screen, send keys, refresh and others.

28.9.2 Screen Customizer Component Interface (SCCI)

SCCI is an API implemented by IBM Screen Customizer's graphical components. It was implemented to allow customers to add business logic to customized screens. It allow the customer to programmatically interact with the following components:

- ▶ Button
- ▶ Valid Values Button
- ▶ Checkbox
- ▶ Choicebox
- ▶ Frame
- ▶ HostList
- ▶ Image
- ▶ ImageButton
- ▶ Label
- ▶ List
- ▶ RadioButton
- ▶ Textfield
- ▶ WebLink

A program using the power of Custom Terminal and SCCI could custom-tailor a session, further customize individual screen, auto-navigate through screens, collect data, import data from external sources such as a JDBC database or a flat text file, simply log a user's session, and many other tasks.

28.9.3 Application programming interface documentation

The IBM Screen Customizer documentation is installed in the Host On-Demand Toolkit directory structure. For illustration purposes, we will assume that the Toolkit is installed in C:\Program Files\IBMHost Access Toolkit\.

Custom Terminal Bean documentation

The documentation can be categorized into three types:

1. Reference material
 - ...\\en\\doc\\beans\\beanReference.html
 - ...\\en\\doc\\beans\\CustomTerminal.html
2. Javadoc
 - ...\\en\\doc\\beans\\com.ibm.hi.customizer.beans.CustomTerminal.html
 - ...\\en\\doc\\beans\\com.ibm.eNetwork.beans.HOD.HostTerminal.html
3. Sample programs
 - ...\\toolkit\\beans\\samples\\CustomTerminalDemo\\CustomTerminalDemo.java
 - ...\\toolkit\\beans\\samples\\CustomTerminalDemo\\readme.txt

SCCI documentation

The documentation can be categorized into three types:

1. Reference material
 - ...\\en\\doc\\beans\\SCCI_reference.html
2. Javadoc
 - ...\\en\\doc\\beans\\packages.html
 - ...\\en\\doc\\beans\\Package-com.ibm.hi.customizer.beans.scci.thml
 - ...\\en\\doc\\beans\\com.ibm.eNetwork.HOD.HIFramework.html
 - ...\\en\\doc\\beans\\com.ibm.eNetwork.HOD.CustomInterface.html
3. Sample programs (all also use Custom Terminal)
 - ...\\en\\doc\\beans\\SCCI_helloWorld.html
 - ...\\toolkit\\scci\\samples\\SCCITestDriver\\SCCITestDriver.java
 - ...\\toolkit\\scci\\samples\\SCCITestDriover\\readme.txt
 - ...\\toolkit\\scci\\samples\\SCLogicDemo\\SCLogicDemo.java
 - ...\\toolkit\\scci\\samples\\SCLogicDemo\\readme.txt

28.10 More information

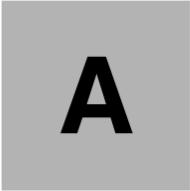
For more information regarding Screen Customizer and how it can be used with Host On-Demand, refer to the following URL:

<http://www.ibm.com/software/network/ScreenCustomizer/>



Part 4

Appendixes



Introduction to TCP/IP security

This appendix discusses basic network security techniques available with TCP/IP, and provides an overview of a number of solutions for addressing security issues in networks.

The field of network security in general and of TCP/IP security in particular is very wide, so this appendix concentrates on the most recent and most widely used security techniques. The following topics are covered:

1. Basic concepts of cryptography and digital certificates
2. Firewall concepts
3. Virtual private network (VPN) and IPsec
4. Secure Sockets Layer (SSL)
5. Transport Layer Security (TLS)

For more details on the concepts covered in this chapter, please see *TCP/IP Tutorial and Technical Overview*, GG24-3376.

Basic concepts of cryptography and digital certificates

If you are sending data in the clear over a network that is not completely under your control from the receiver to the sender, you will be unable to ensure the following security functions:

- ▶ **Privacy**

Anyone who is able to intercept your data might be able to read it.

- ▶ **Integrity**

An intermediary might be able to alter your data.

- ▶ **Accountability or non-repudiation**

It may be impossible to determine the originator of a message with confidence, and thus the person who sent the message could deny being the originator.

Security functions such as identification and authentication are also impacted because if authentication data such as passwords are sent without integrity and privacy, they can be intercepted in transit between sender and receiver, making the authentication compromised and worthless.

To ensure privacy, integrity and accountability in non-secure networks, cryptographic procedures need to be used. Today, two distinct classes of encryption algorithms are in use: symmetric and asymmetric algorithms. They are fundamentally different in *how* they work, and thus in *where* they are used.

Symmetric encryption algorithms

An encryption algorithm is called symmetric because the same key that is used to encrypt the data is also used to decrypt the data and recover the clear text (see Figure A-1). The cipher and decipher processes are usually mathematically complex nonlinear permutations.

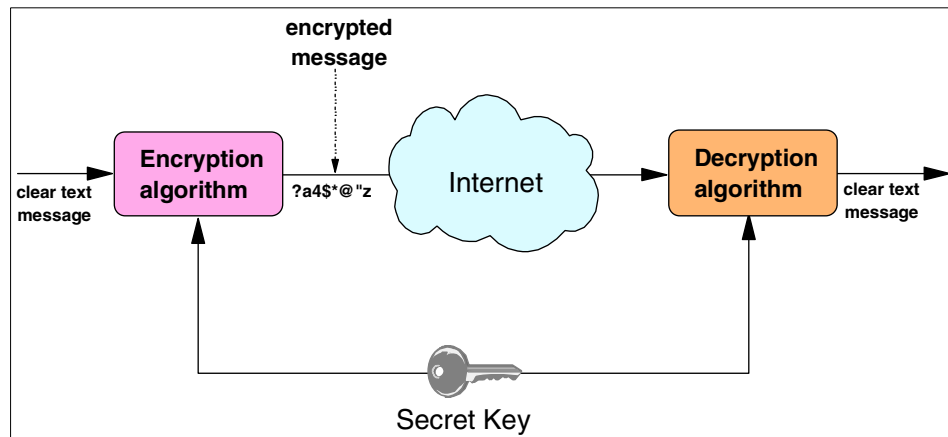


Figure A-1 Symmetric encryption and decryption: using the same key

Symmetric algorithms are usually efficient in terms of processing power, so they are ideal for encryption of bulk data. However, they have one major drawback, which is key management. The sender and receiver on any secure connection must share the same key; in a large network where thousands of users may need to communicate securely, it is extremely difficult to manage the distribution of keys so as not to compromise the integrity of any one of them.

Frequently used symmetric algorithms include:

► **Data Encryption Standard (DES)**

Developed in the 1970s by IBM scientists, DES uses a 56-bit key. Stronger versions called Triple DES have been developed that use three operations in sequence: “2-key Triple DES” encrypts with key 1, decrypts with key 2, and encrypts again with key 1. The effective key length is 112 bits. “3-key Triple DES” encrypts with key 1, decrypts with key 2, and encrypts again with key 3. The effective key length is 168 bits.

► **Commercial Data Masking Facility (CDMF)**

This is a version of the DES algorithm approved for use outside the U.S. and Canada (in times when export control was an issue). It uses 56-bit keys, but 16 bits of the key are known, so the effective key length is 40 bits.

► **RC2**

Developed by Ron Rivest for RSA Data Security, Inc., RC2 is a block cipher with variable key lengths operating on 8-byte blocks. Key lengths of 40, 56, 64, and 128 bits are in use.

► **RC4**

Developed by Ron Rivest for RSA Data Security, Inc., RC4 is a stream cipher operating on a bit stream. Key lengths of 40 bits, 56 bits, 64 bits, and 128 bits are in use. The RC4 algorithm always uses 128-bit keys; the shorter key lengths are achieved by “salting” the key with a known, non-secret random string.

► **Advanced Encryption Standard (AES)**

As a result of a contest for a follow-on standard to DES held by the National Institute for Standards and Technology (NIST), the Rijndael algorithm was selected. This is a block cipher created by Joan Daemen and Vincent Rijmen with variable block length (up to 256 bits) and variable key length (up to 256 bits).

► **The International Data Encryption Algorithm (IDEA)**

IDEA was developed by James Massey and Xuejia Lai at ETH in Zurich. It uses a 128-bit key and is faster than triple DES.

DES is probably the most scrutinized encryption algorithm in the world. Much work has been done to find ways to break DES, notably by Biham and Shamir, but also by others. However, a way to break DES with appreciably less effort than a brute-force attack (breaking the cipher by trying every possible key) has not been found.

Both RC2 and RC4 are proprietary, confidential algorithms, that have never been published. They have been examined by a number of scientists under non-disclosure agreements.

With all the ciphers listed above, it can be assumed that a brute-force attack is the only means of breaking the cipher. Therefore, the work factor depends on the length of the key. If the key length is n bits, the work factor is proportional to $2^{(n-1)}$.

Today, a key length of 56 bits is generally only seen as sufficiently secure for applications that do not involve significant amounts of money or critically secret data. If specialized hardware is built (such as the machine built by John Gilmore and Paul Kocher for the Electronic Frontier Foundation), the time needed for a brute-force attack can be reduced to about 100 hours or less (see: *Cracking DES: Secrets of Encryption Research, Wiretap Politics & Chip Design*, by Electronic Frontier Foundation, John Gilmore (Editor), 1988). Key lengths of 112 bits and above are seen as unbreakable for many years to come, since the work factor rises exponentially with the size of the key.

Asymmetric encryption algorithms

Asymmetric encryption algorithms are so called because the key that is used to encrypt the data cannot be used to decrypt the data; a different key is needed to recover the clear text (see Figure A-2). This key pair is called a public key and a private key. If the public key is used to encrypt the data, the private key must be used to recover the clear text. If data is encrypted with the private key, it can only be decrypted with the public key.

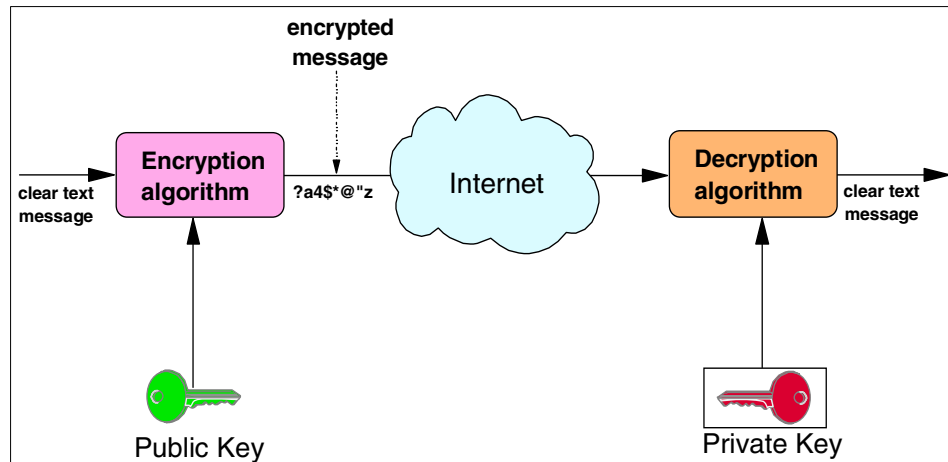


Figure A-2 Public-key cryptography: using a key pair

Asymmetric encryption algorithms, commonly called Public Key Cryptography Standards (PKCS), are based on mathematical algorithms. The basic idea is to find a mathematical problem that is very hard to solve. The algorithm in most widespread use today is RSA. However, some companies have begun to implement public-key cryptosystems based on elliptic curve algorithms. With the growing proliferation of IPsec, the Diffie-Hellman algorithm is gaining popularity.

A brief overview of all three methods follows:

► RSA

Invented 1977 by Rivest, Shamir, and Adleman (who formed RSA Data Security Inc.). The idea behind RSA is that integer factorization of very large numbers is extremely hard to do. Key lengths of public and private keys are typically 512 bits, 768 bits, 1024 bits, or 2048 bits. The work factor for RSA with respect to key length is sub-exponential, which means the effort does not rise exponentially with the number of key bits. It is roughly $2^{0.3n}$.

► **Elliptic Curve**

Public-key cryptosystems based on elliptic curves use a variation of the mathematical problem of finding discrete logarithms. It has been stated that an elliptic curve cryptosystem implemented over a 160-bit field has roughly the same resistance to attack as RSA with a 1024-bit key length. Properly chosen elliptic curve cryptosystems have an exponential work factor (which explains why the key length is so much smaller). Elliptic curve cryptosystems are now standardized by FIPS PUB 186-2, the digital signature standard (January 2000).

► **Diffie-Hellman**

W. Diffie and M.E. Hellman, the inventors of public key cryptography, published this algorithm in 1976. The mathematical problem behind Diffie-Hellman is computing a discrete logarithm. Both parties have a public-private key pair each; they are collectively generating a key only known to them. Each party uses its own private key and the public key of the other party in the key generation process. Diffie-Hellman public keys are often called *shares*.

The beauty of asymmetric algorithms is that they are not subject to the key management issues that beset symmetric algorithms. Your public key is freely available to anyone, and if someone wants to send you a message he or she encrypts it using that key. Only you can understand the message, because only you have the private key. Asymmetric algorithms are also very useful for authentication. Anything that can be decrypted using your public key must have been encrypted using your private key, in other words, by you.

Performance issues of cryptosystems

Elliptic curve cryptosystems are said to have performance advantages over RSA in decryption and signing. While the possible differences in performance between the asymmetric algorithms are somewhere in the range of a factor of 10, the performance differential between symmetric and asymmetric cryptosystems is far more dramatic.

For instance, it takes about 1000 times as long to encrypt the same data with RSA (an asymmetric algorithm) as with DES (a symmetric algorithm), and implementing both algorithms in hardware does not change the odds in favor of RSA.

As a consequence of these performance issues, the encryption of bulk data is usually performed using a symmetric cryptosystem, while asymmetric cryptosystems are used for electronic signatures and in the exchange of key material for secret-key cryptosystems. With these applications, only relatively small amounts of data need to be encrypted and decrypted, and the performance issues are less important.

Cryptosystems for data integrity

Data integrity is the ability to assert that the data received over a communication link is identical to the data sent. Data integrity in an insecure network requires the use of cryptographic procedures. However, it does not imply that only the receiver is able to read the data, as with data privacy. Data could be compromised not only by an attacker, but also by transmission errors (although those are normally handled by transmission protocols such as TCP).

Message digest algorithms

A message digesting algorithm (often also called a “digital hash”) is an algorithm that “digests” (condenses) a block of data into a shorter string (usually 128 or 160 bits), which is called a message digest, secure hash, or Message Integrity Code (MIC). See Figure A-3 for a graphical representation. The principle behind message digest algorithms is as follows:

- The message cannot be recovered from the message digest.

It is very hard to construct a block of data that has the same message digest as another given block.

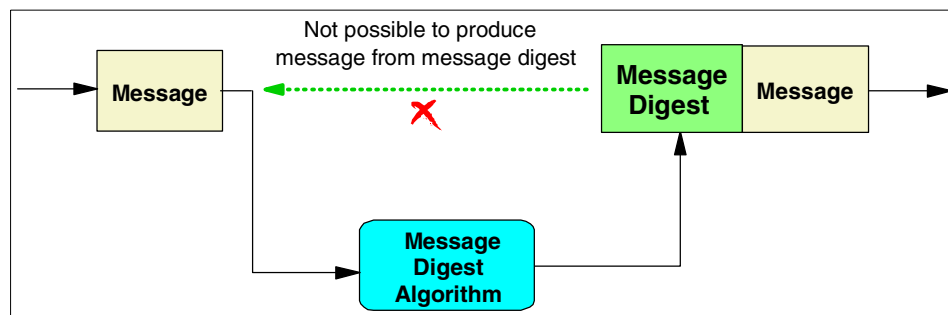


Figure A-3 Message digest

Common message digest algorithms are:

► **MD2**

Developed by Ron Rivest of RSA Data Security, Inc. The algorithm is mostly used for Privacy Enhanced Mail (PEM) certificates. MD2 is fully described in RFC 1319. Since weaknesses have been discovered in MD2, its use is discouraged.

► **MD5**

Developed in 1991 by Ron Rivest. The algorithm takes as input a message of arbitrary length and produces as output a 128-bit message digest of the input. The MD5 message digest algorithm is specified in RFC 1321, *The MD5 Message-Digest Algorithm*. Collisions have been found in MD5; see *Cryptanalysis of MD5 Compress*, by Hans Dobbertin, available at <http://www.cs.ucsd.edu/users/bsy/dobbertin.ps>.

► **SHA-1**

Developed by the National Security Agency (NSA) of the U.S. Government. The algorithm takes as input a message of arbitrary length and produces as output a 160-bit “hash” of the input. SHA-1 is fully described in standard FIPS PUB 180-1, also called the Secure Hash Standard (SHS). SHA-1 is generally recognized as the strongest and most secure message digesting algorithm.

► **SHA-256, SHA-512**

Developed by the National Security Agency (NSA) of the U.S. Government. The security of a hash algorithm against collision attacks is half the hash size and this value should correspond with the key size of encryption algorithms used in applications together with the message digest. Since SHA-1 only provides 80 bits of security against collision attacks, this is deemed inappropriate for the key lengths of up to 256 bits planned to be used with AES. Therefore, extensions to the Secure Hash Standard (SHS) have been developed. SHA-256 provides a hash size of 256 bits while SHA-512 provides a hash size of 512 bits.

Message digests for data integrity

The sender of a message (block of data) uses an algorithm, for example, SHA-1, to create a message digest from the message (see Figure A-4). The message digest can be sent together with the message to provide data integrity. The receiver runs the same algorithm over the message and compares the resulting message digest to the one sent with the message. If both match, the message is unchanged.

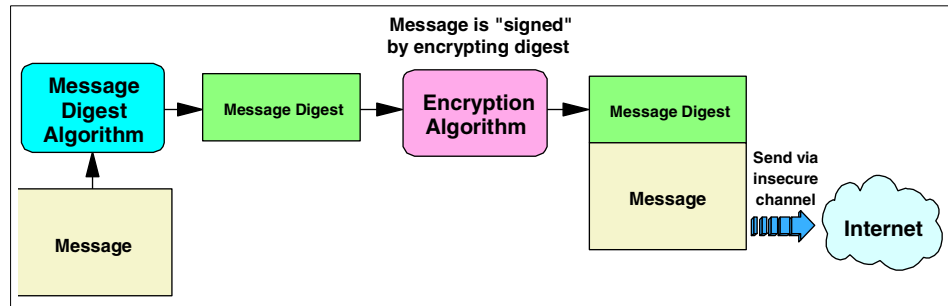


Figure A-4 Message digest for data integrity

The message digest should not be sent in the clear: Since the digest algorithms are well-known and no key is involved, a man-in-the-middle could not only forge the message but also replace the message digest with that of the forged message. This would make it impossible for the receiver to detect the forgery. The solution for this is to encrypt the message digest, that is, to use a message authentication code (MAC).

Message authentication codes

Secret-key cryptographic algorithms, such as DES, can be used for encryption with message digests. A disadvantage is that, as in secret-key cryptosystems, the keys must be shared by sender and receiver. Furthermore, since the receiver has the key that is used in MAC creation, this system does not offer a guarantee of non-repudiation. That is, it is theoretically possible for the receiver to forge a message and claim it was sent by the sender. Therefore, message authentication codes are usually based on public/private key encryption in order to provide for non-repudiation. This is discussed further in “Digital signatures” on page 998.

Keyed hashing for message authentication (HMAC)

H. Krawczyk and R. Canetti of IBM Research and M. Bellare of UCSD invented a method to create a message authentication code called HMAC, which is defined in RFC 2104 as a proposed Internet standard. A simplified description of how to create the HMAC is as follows: The key and the data are concatenated and a message digest is created. The key and this message digest are again concatenated for better security, and another message digest is created, which is the HMAC.

HMAC can be used with any cryptographic hash function. Typically, either MD5 or SHA-1 are used. In the case of MD5, a key length of 128 bits is used (the block length of the hash algorithm). With SHA-1, 160-bit keys are used. Using HMAC actually improves the security of the underlying hash algorithm. For instance, some collisions (different texts that result in the same message digest) have been found in MD5. However, they cannot be exploited with HMAC. Therefore the weakness in MD5 does not affect the security of HMAC-MD5.

HMAC is now a PKCS#1 V.2 standard for RSA encryption (proposed by RSA Inc. after weaknesses were found in PKCS#1 applications). For further details, see <http://www.ietf.org/rfc.html>. HMAC is also used in the Transport Layer Security (TLS) Protocol, the successor to SSL.

Message authentication used with SSL

In the Secure Sockets Layer Protocol (SSL), a slightly different MAC algorithm has been implemented. The MAC write-secret and the sequence number of the message are concatenated with the data, and a message digest is created. The MAC write-secret and this message digest are again concatenated for better security, and another message digest is created, which is the MAC. Again, for the hash function, either MD5 or SHA-1 can be used. If compression is used, the text is compressed before the MAC is calculated.

Digital signatures

Digital signatures are an extension to data integrity. While data integrity only ensures that the data received is identical to the data sent, digital signatures go a step further: they provide non-repudiation. This means that the sender of a message (or the signer of a document) cannot deny authorship, similar to signatures on paper. As illustrated in Figure A-5, the creator of a message or electronic document that is to be signed uses a message digesting algorithm such as MD5 or SHA-1 to create a message digest from the data. The message digest and some information that identifies the sender are then encrypted with an asymmetric algorithm using the sender's private key. This encrypted information is sent together with the data.

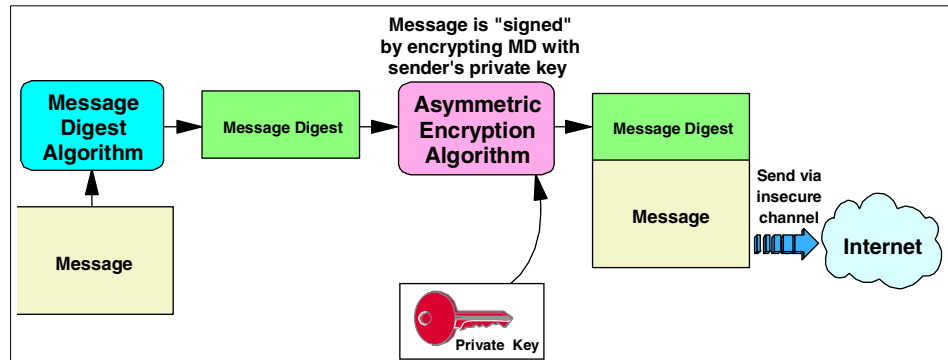


Figure A-5 Digital signature creation

The receiver, as shown in Figure A-6, uses the sender's public key to decrypt the message digest and identification of the sender. He or she will then use the message digesting algorithm to compute the message digest from the data. If this message digest is identical to the one recovered after decrypting the digital signature, the signature is recognized as valid proof of the authenticity of the message.

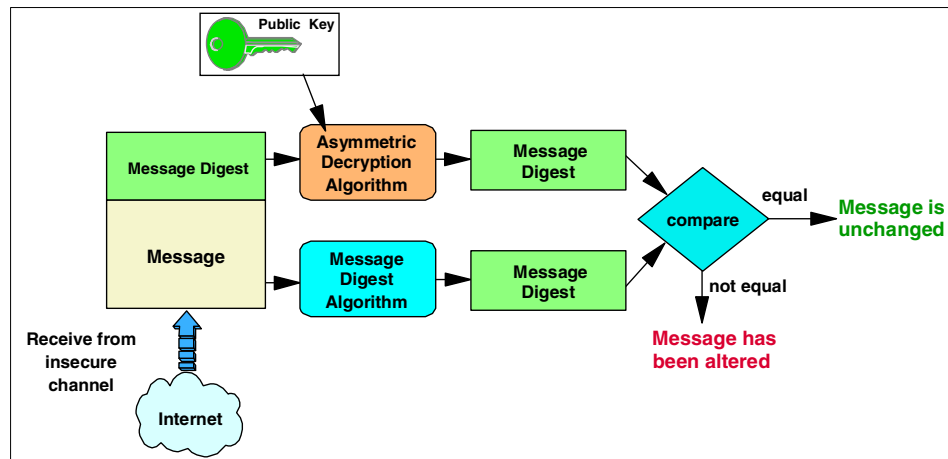


Figure A-6 Digital signature verification

With digital signatures, only public-key cryptosystems can be used. If secret-key cryptosystems would be used to encrypt the signature, it would be very difficult to make sure that the receiver (having the key to decrypt the signature) could not misuse this key to forge a signature of the sender. The private key of the sender is known to nobody else, so nobody is able to forge the sender's signature.

Note the difference between encryption using public-key cryptosystems and digital signatures:

- With encryption, the sender uses the receiver's public key to encrypt the data, and the receiver decrypts the data with his private key. This means everybody can send encrypted data to the receiver that only the receiver can decrypt. See Figure A-7 for a graphical representation.

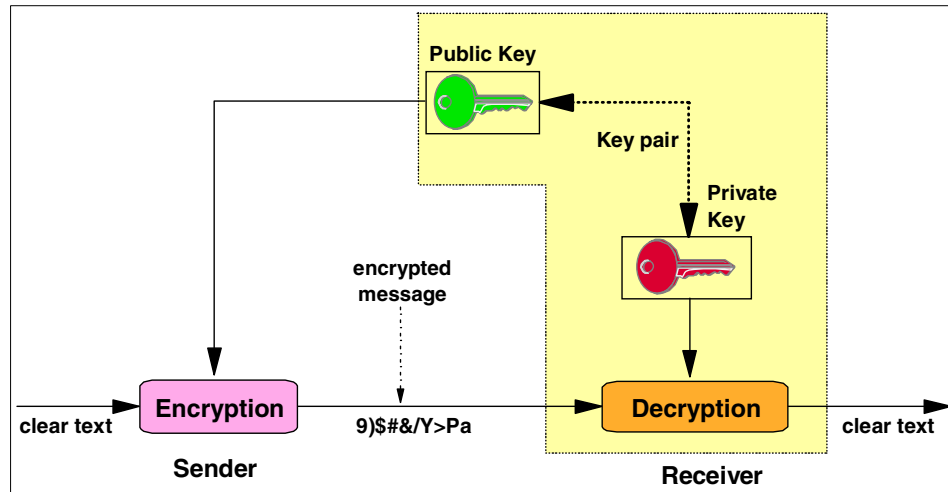


Figure A-7 Encrypting data with the receiver's public key

- With digital signatures, the sender uses his private key to encrypt his signature, and the receiver decrypts the signature with the sender's public key. This means that only the sender can encrypt the signature, but everybody who receives the signature can decrypt and verify it.

The tricky part with digital signatures is the trustworthy distribution of public keys, since a genuine copy of the sender's public key is required by the receiver. A solution to this problem is provided by digital certificates, which are discussed next.

Public Key Infrastructure

A Public Key Infrastructure (PKI) offers the basis for practical usage of public key encryption. A PKI defines the rules and relationships for certificates and Certificate Authorities (CAs). It defines the fields that can or must be in a certificate, the requirements and constraints for a CA in issuing certificates, and how certificate revocation is handled.

PKI has been exploited in many applications or protocols, such as Secure Sockets Layer (SSL), Secure Multimedia Internet Mail Extensions (S/MIME), IP Security (IPSec), Secure Electronic Transactions (SET), and Pretty Good Privacy (PGP). PKI is described here, only insofar as its use with Web serving and Secure Sockets Layer (SSL) is concerned. For more information on PKI, refer to *Deploying a Public Key Infrastructure*, SG24-5512.

Digital certificates

When using a PKI, the user must be confident that the public key belongs to the correct remote person (or system) with which the digital signature mechanism is to be used. This confidence is obtained through the use of public key digital certificates. A digital certificate is analogous to a passport: the passport certifies the bearer's identity, address and citizenship. The concepts behind passports and other identification documents (for instance, drivers' licenses) are very similar to those that are used for digital certificates.

Passports are issued by a trusted authority, such as a government passport office. A passport will not be issued unless the person who requests it has proven their identity and citizenship to the authority. Specialized equipment is used in the creation of passports to make it very difficult to alter the information in it or to forge a passport altogether. Other authorities, for instance, the border police in other countries, can verify a passport's authenticity. If they trust the authority that issued the document, they implicitly trust the passport.

A digital certificate serves two purposes: it establishes the owner's identity and it makes the owner's public key available. Similar to a passport, a certificate must be issued by a trusted authority, the CA; and, like a passport, it is issued only for a limited time. When its expiration date has passed, it must be replaced.

Trust is a very important concept in passports, as well as in digital certificates. In the same way as, for instance, a passport issued by the governments of some countries, even if recognized to be authentic, will probably not be trusted by the US authorities, each organization or user has to determine whether a CA can be accepted as trustworthy.

For example, a company might want to issue digital certificates for its own employees from its own Certificate Authority; this could ensure that only authorized employees are issued certificates, as opposed to certificates being obtained from other sources such as a commercial entity such as VeriSign.

The information about the certificate owner's identity is stored in a format that follows RFC 2253 and the X.520 recommendation, for instance: CN=George Baker O=IBM Corporation; the complete information is called the owner's distinguished name (DN). The owner's distinguished name and public key and the CA's distinguished name are digitally signed by the CA; that is, a message digest is calculated from the distinguished names and the public key. This message digest is encrypted with the private key of the CA.

Figure A-8 shows the layout of a digital certificate.

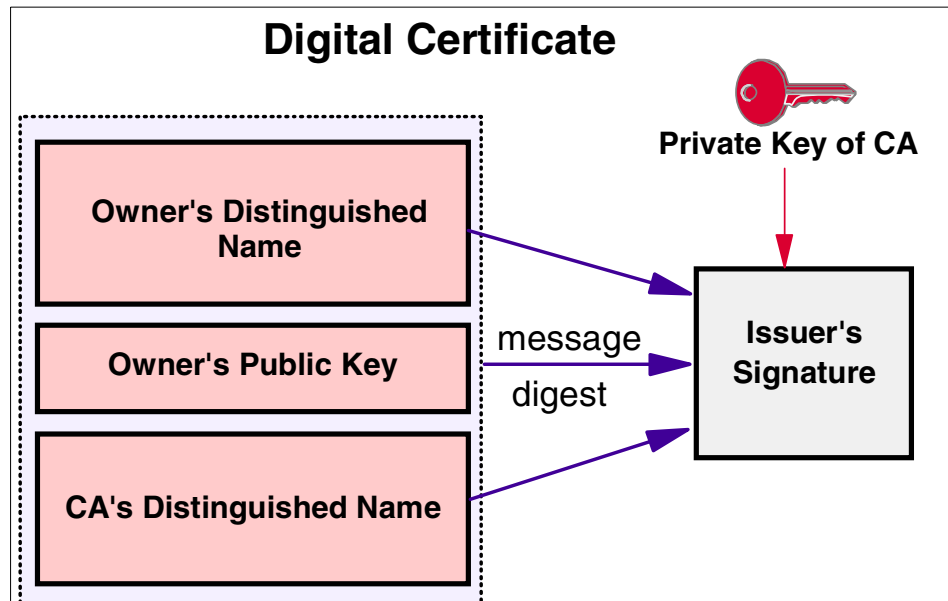


Figure A-8 Simplified layout of a digital certificate

The digital signature of the CA serves the same purpose as the special measures taken for the security of passports such as laminating pages with plastic material; it allows others to verify the authenticity of the certificate. Using the public key of the CA, the message digest can be decrypted. The message digest can be recreated; if it is identical to the decrypted message digest, the certificate is authentic.

Security considerations for certificates

If I send my certificate with my public key in it to someone else, what keeps this person from misusing my certificate and posing as myself? The answer is: my private key.

A certificate alone can never be proof of anyone's identity. The certificate just allows the identity of the certificate owner to be verified by providing the public key that is needed to check the certificate owner's digital signature. Therefore, the certificate owner must protect the private key that matches the public key in the certificate. If the private key is stolen, the thief can pose as the legitimate owner of the certificate. Without the private key, a certificate cannot be misused.

An application that authenticates the owner of a certificate cannot accept just the certificate. A message signed by the certificate owner should accompany the certificate. This message should use elements such as sequence numbers, time stamps, challenge-response protocols, or other data that allow the authenticating application to verify that the message is a "fresh" signature from the certificate owner and not a replayed message from an impostor.

Certificate Authorities and trust hierarchies

A user of a security service requiring knowledge of a public key generally needs to obtain and validate a certificate containing the required public key. To verify that the certificate is authentic, the receiver needs the public key of the CA that issued the certificate.

Most Web browsers come configured with the public keys of common CAs (such as VeriSign). However, if the user does not have the public key of the CA that signed the certificate, an additional certificate would be needed in order to obtain that public key. In general, a chain of multiple certificates may be required, comprising a certificate of the public key owner signed by a CA, and possibly additional certificates of CAs signed by other CAs. Many applications that send a subject's certificate to a receiver send, not only just that certificate but also all the CA certificates necessary to verify the certificate up to the root.

Obtaining and storing certificates

As has been discussed, certificates are issued by a CA. Clients usually request certificates by going to the CA's Web site. After verifying the validity of the request, the CA sends back the certificate in an e-mail message or allows it to be downloaded.

Requesting server certificates

Server certificates can be either self-signed or they can be signed by an external CA. The server environment will determine which kind of certificate should be used: in an intranet environment, it is generally appropriate to use self-signed certificates. In an environment where external users are accessing the server over the Internet, it is usually advisable to acquire a server certificate from a well-known CA, because the steps needed to import a self-signed certificate

might seem obscure, and most users will not have the ability to discern whether the action they are performing is of trivial consequence or not. It should also be noted that a root CA certificate received over a channel that is not trusted, such as the internet, does not deserve any kind of trust.

Firewall concepts

A firewall machine is a computer used to separate a secure network from a non-secure network (Figure A-9). Such networks are typically based on the TCP/IP protocol, but the concept of a firewall concept is not restricted to just TCP/IP.

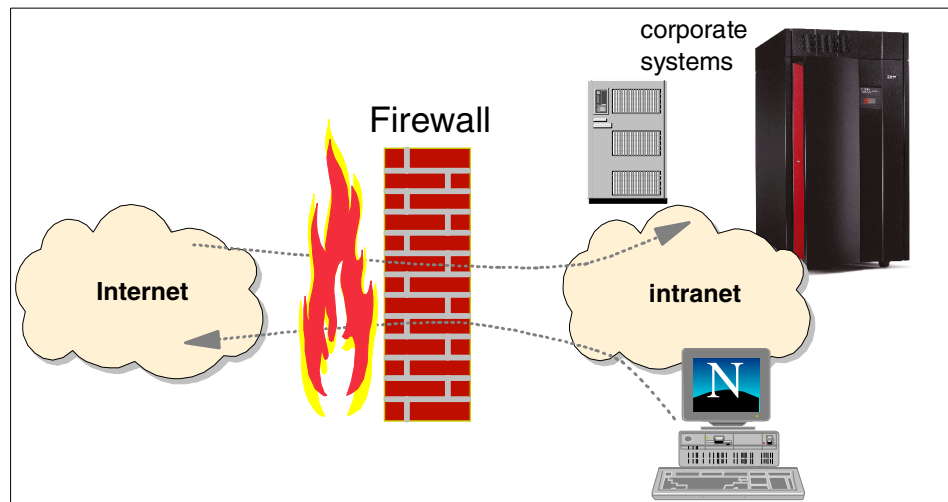


Figure A-9 The firewall concept

Firewalls have become an important concept in TCP/IP-based networks, because the global Internet is a TCP/IP-based network and is often perceived as being a non-secure place to enter or traverse. Yet you still want your intranet (perceived as being a secure place) to be connected to the non-secure Internet.

The reasons for establishing connections between an intranet and the Internet are many, but generally fall into two categories:

- ▶ You want to provide a service to the Internet community or want to conduct business on the Internet.
- ▶ You want to allow your internal employees to access the vast amount of services on the Internet, as well as the ability to exchange or share information with other users on the Internet or through the Internet.

At this point, it might be useful to define the following terms:

- ▶ The term *intranet* refers to an internal TCP/IP network.
- ▶ The term *Internet* refers to the World Wide Web, and the associated infrastructure of news groups, e-mail, chat rooms and other services.
- ▶ The term *extranet* refers to TCP/IP networks of different companies connected with a secure connection, perhaps using virtual private network technology (VPN).

Doing e-business on the Internet is very different from just serving static information out of a Web server. Doing e-business means that you have to establish an environment where users on the Internet are able to interact with the applications and data that your daily existence as a company is based on and relies upon.

That data and those applications are likely, to a large extent, to be located in your environment, which means that you probably already are, or in the near-term future will be, challenged with the request to establish Internet access to your production environment.

When you connect your intranet to the Internet and define a strategy for how your firewall should function, you may think that it is sufficient to block all types of traffic that represent a risk, and allow the remaining traffic to pass through the firewall. However, such a strategy is based on the assumption that all risks are known in advance and that existing well-behaving traffic will remain well-behaving; such an assumption is a mistake. New ways of exploiting existing applications and well-known application protocols are being found every week, so an application that may be considered harmless today may be the instrument of an attack tomorrow.

General guidelines for implementing firewalls

A few general guidelines for implementing firewall technologies are worth including.

Before you start connecting your intranet to the Internet, define a security policy for how your firewall should function and how demilitarized zones should be configured. Decide what type of traffic is allowed through the firewall, and under what conditions, what kind of servers are to be placed in demilitarized zones, and what type of traffic is allowed between the demilitarized zone and the intranet.

When actually configuring your firewall, start by disallowing everything and then proceed by enabling those services you have defined in your security policy. Everything that is not specifically allowed should be prohibited.

If you establish more than a single gateway between your internal network and the Internet, make sure that all gateways implement the same level of security. It is common practice to use different firewall products in a vertical setup (product A between the Internet and the demilitarized zone and product B between the demilitarized zone and the intranet). That way, a hacker exploiting a vulnerability in product A is still stopped by product B. Of course, it does not make sense to use this concept in a horizontal setup (one gateway uses product A, the other one product B) because a hacker will get in at the weakest link.

If you build a perfect firewall on one end of your network while users on the other end dial in to the Internet from their LAN-attached PCs, enabling those PCs to act as IP routers between your internal network and the Internet, a hacker is soon going to exploit that back door into your network instead of wasting his time trying to break through your firewall.

One of the most important aspects of a firewall is its ability to log both successful and rejected access events. However, these logs are worth nothing if you do not set up daily administrative procedures to analyze and react to the information that can be derived from these logs.

By analyzing the firewall logs, you should be able to detect if unauthorized accesses were attempted and if your firewall protection succeeded in rejecting such attacks, or if it failed and allowed an intruder to gain access to resources that should not have been accessed. In addition, it might be a good idea to install an intrusion detection system.

This list is not all-inclusive, but merely points out some of the most important aspects of implementing firewall technologies in your network.

So far, the Internet has been considered to be the non-secure place, while your internal network has been considered the secure place. However, that may in some situations be an oversimplification. For example, consider a research department that works with highly confidential information. In such an environment, you may want to protect that research department from your regular users by implementing a firewall between your regular internal network and the network in your research department.

Firewall categories

There are many firewall technologies available, but they can in general be grouped into two major categories:

- ▶ Those that allow IP packets to be routed between two or more networks, namely packet-filtering routers.

- Those that disable IP routing, but relay data through specialized application programs, namely application-level gateways or proxies.

Packet filtering

A packet filtering router, as shown in Figure A-10, is a special type of IP router. What differentiates a firewall packet filtering router from a normal IP router is that it applies one or more technologies to analyze the IP packets and decide if a packet is allowed to flow through the firewall or not. Such a firewall is sometimes also referred to as a screening filter, or router firewall.

Some packet-filtering techniques only act on data in the headers of individual packets, while others also look at data depending on the type of packet. The traditional packet-filtering router is stateless (each packet is handled independently) but there are products that save state over multiple packages and base their actions on the state information.

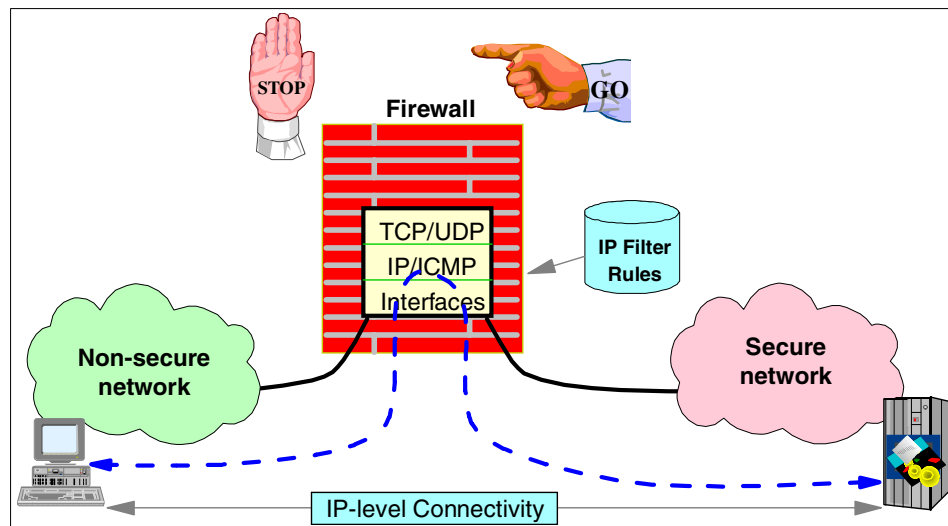


Figure A-10 Packet filtering firewall

Application-level gateway

An application-level gateway, sometimes referred to as a bastion host, is a machine that disables IP-level routing between the non-secure network and the secure network, but allows specialized application gateway programs (termed proxies) that run on the firewall to communicate with both the secure network and the non-secure network. See Figure A-11.

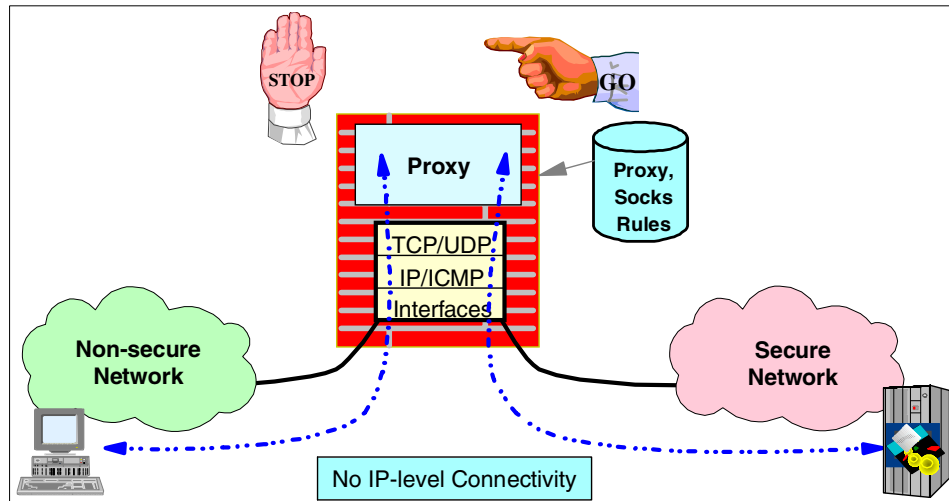


Figure A-11 Application gateway firewall

The proxy applications on the firewall act as relay applications between users or applications on the secure and the non-secure networks. Examples of such proxy applications are HTTP or FTP proxy servers. The SOCKS server is also an application-level gateway, but a special kind, sometimes referred to as a circuit level gateway. A SOCKS server can relay all TCP and UDP connections, not just HTTP or FTP sessions. It does not provide any extra packet processing or filtering, and unlike proxy servers, it is often used for outbound connections through a firewall.

A firewall may not always have to be configured as either a packet-filtering router or as a proxy; it may be configured to perform the following functions:

- ▶ IP filtering
- ▶ Network address translation (NAT)
- ▶ Virtual private networks (VPN)
- ▶ FTP proxy server
- ▶ SOCKS server
- ▶ Domain name services

An excellent discussion of firewall technologies can be found in *TCP/IP Tutorial and Technical Overview*, GG24-3376.

The demilitarized zone

The demilitarized zone (DMZ) is a term often used when describing firewall configurations. Figure A-12 shows a typical example. A DMZ is an isolated subnet between your secure network and the Internet. Much as the no-man's land between two entrenched armies, anyone can enter it, but the only things present are those that you want to allow access to anyway. Nowadays, a demilitarized zone is an area in which you place the Web servers and other servers for public access, but which you also wish to protect to some degree.

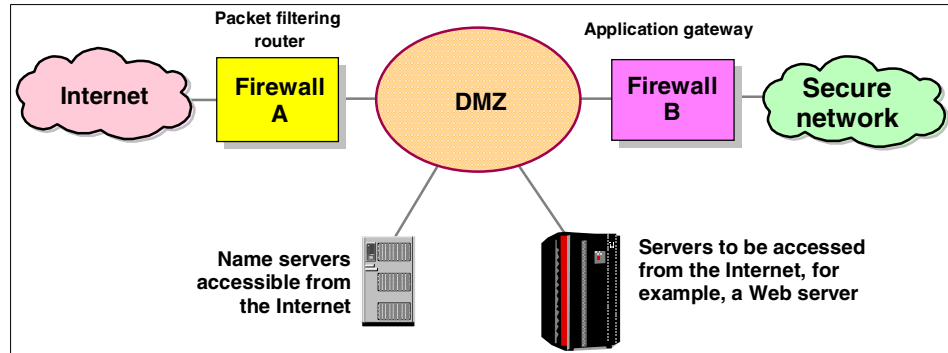


Figure A-12 A demilitarized zone

This is achieved by placing an outer firewall (often a packet-filtering router) between the Internet and the servers in the DMZ, and another firewall (often an application-level gateway) between your secure network and the DMZ. The outer firewall is designed to allow into the DMZ only those requests you wish to receive at your Web servers, but could also be configured to block denial-of-service attacks and to perform network address translation of the servers in your DMZ. The inner firewall is designed to prevent unauthorized access to your secure network from the DMZ and also perhaps to prevent unauthorized access from your secure network to the DMZ or the connected non-secure network.

When you put a server into a DMZ, it is strongly recommended that you use firewall technologies. You should use firewall technologies to block all traffic into and out of your server that does not belong to the services you are going to offer from this server. This control should be in place even if you already have a packet-filtering router or firewall between the insecure network and this server.

Hardening

Hardening is a process done to firewalls to make them more secure. All unnecessary services, user accounts, and software on the operating system are removed or disabled.

An operating system is designed to fit computers with different configurations doing different tasks. To accomplish this, extra items are installed with the operating system that will only be used in certain situations. Much of the time, these extra items just take up resources and might cause the computer to run slower. On a firewall, these items become more of a problem. They become unnecessary, and potential security exposures.

Almost everything on a firewall is a potential security exposure. By disabling and removing the unnecessary items, there will be less exposure for a hacker to exploit. All services, user IDs, and software on a firewall should be required only by the operating system or the firewall. Everything else should be removed.

Many firewalls will perform a limited amount of hardening. However, it is the responsibility of the firewall administrator to finish the task.

Virtual private network (VPN) and IPSec

A virtual private network (VPN) provides secure connections across the Internet, by establishing a “tunnel” between two secure networks. It is a generic solution that is application- and protocol-independent. A VPN encapsulates the IP datagram into another IP datagram in order to maintain data privacy. It can be used by two disparate parts of a corporation to connect their internal private networks by means of a non-secure network such as the Internet. An example of a VPN configuration is shown in Figure A-13.

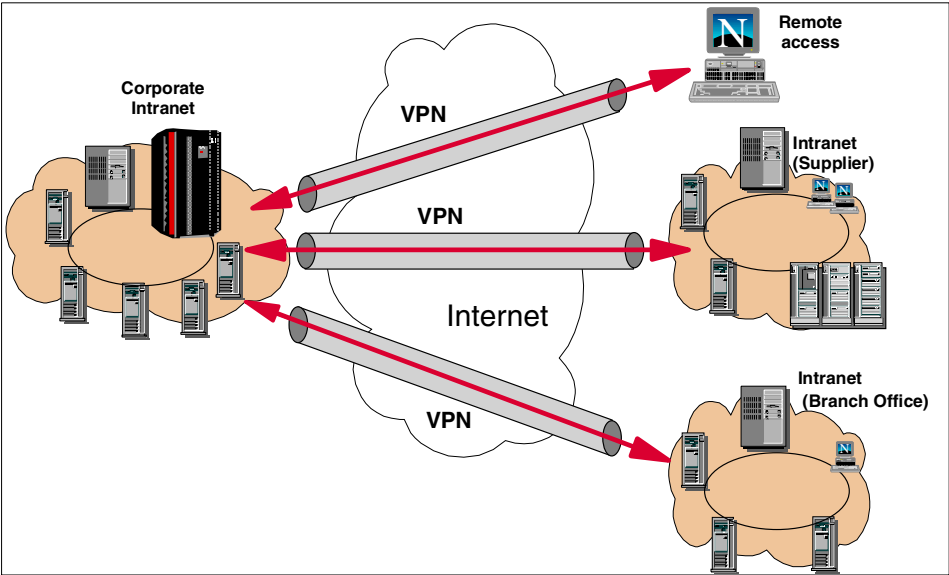


Figure A-13 Virtual private networks

IPSec

In Figure A-14 the TCP/IP layered protocol stack is shown, with the security-related protocols associated with each layer:

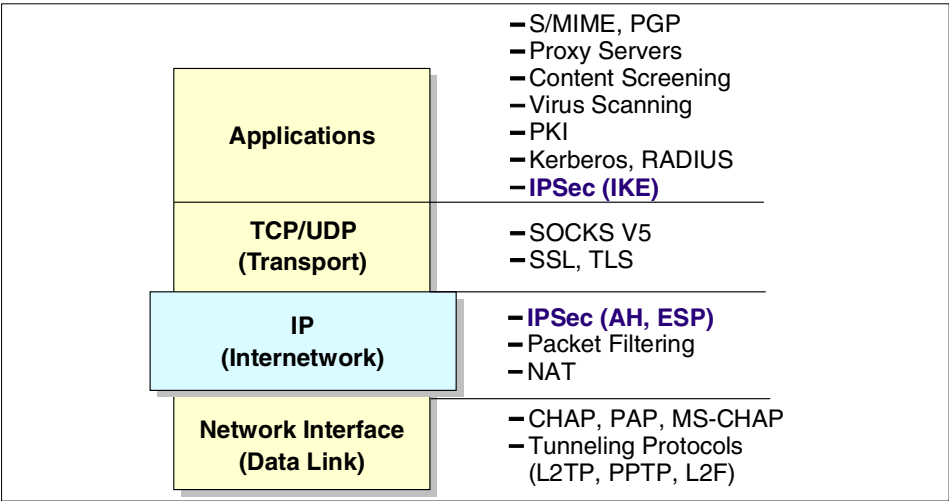


Figure A-14 The TCP/IP protocol stack and the security-related protocols

Within the layered communications protocol stack model, the network layer (IP in the case of the TCP/IP stack) is the lowest layer that can provide end-to-end security. Network-layer security protocols provide blanket protection for all upper-layer application data carried in the payload of an IP datagram, without requiring a user to modify the applications.

The IP Security Architecture (IPSec) open framework is defined by the IPSec Working Group of the IETF. IPSec is called a framework because it provides a stable, long-lasting base for providing network layer security. It can accommodate today's cryptographic algorithms, and can also accommodate newer, more powerful algorithms as they become available. IPv6 implementations must support IPSec, and IPv4 implementations are strongly recommended to do so.

IPSec is comprised of a number of components described in individual RFCs that are designed to operate together:

- ▶ Security Protocols - IP Authentication Header (AH) provides data origin authentication, data integrity, and replay protection while IP Encapsulating Security Payload (ESP) provides data confidentiality, data origin authentication, data integrity, and replay protection.
- ▶ Security Associations - an SA is a kind of session between two hosts defining the protocols to be used when transmitting data. ISAKMP (Internet Security Association and Key Management Protocol) is a generic framework for negotiating SAs and keys.
- ▶ Key Management - Internet Key Exchange (IKE) provides a method for automatically setting up security associations and managing and exchanging their cryptographic keys.

Security Associations

An IPSec Security Association (SA) corresponds to a session between two hosts. It defines the set of protocols and, with these, the negotiated algorithms and keys that are to be used when transmitting data between two hosts. An SA for data traffic is always unidirectional, so for a pair of hosts that are to communicate securely, at least two SAs, one for each direction, are needed. This differs from other protocols that make use of sessions such as, for instance, SSL. An SSL session covers the transmission in both directions.

Negotiating Security Associations (ISAKMP and IKE)

Before any data can be sent between two hosts using IPSec, a SA needs to be established. The IPSec architecture provides two methods for establishing an SA: a manual tunnel or ISAKMP/IKE.

With manual tunnels, the SA and keying material are generated on one of the hosts (the tunnel owner), transferred to the other host (the tunnel partner) with an out-band transport mechanism, and then imported. This procedure needs to be repeated whenever the validity of the keys has expired and new keying material needs to be generated.

Contrary to manual tunnels, ISAKMP and IKE provide automatic management of sessions and keys. ISAKMP provides a generic framework for the negotiation of SAs and keying material. It defines the procedures and packet formats to establish, negotiate, modify and delete SAs, but it does not provide any specific key-generation techniques or cryptographic algorithms.

Internet Key Exchange (IKE) is based on two protocols: Oakley (*The Oakley Key Determination Protocol*, by H. Orman; RFC 2412, November 1998) and SKEME (*SKEME: A Versatile Secure Key Exchange Mechanism for the Internet*, by H. Krawczyk; IEEE Proceedings, 1996). For the key exchange, Diffie-Hellman (DH) shares are used and the shared key thus obtained is used to derive the keys for data encryption and message authentication. Authentication can be performed with one of three alternatives:

- ▶ Digital signatures
- ▶ Public key encryption
- ▶ A shared secret (a key previously known to both parties)

The use of DH shares causes the connection to have a property called “perfect forward secrecy”. This means that even if the keys for one session are completely compromised, the keys for previous sessions are still safe.

Phases: it takes two

Two hosts can communicate with each other in many different ways that may need different sorts of protection. For instance, some traffic may need encryption and authentication, while other traffic may only need authentication.

IKE uses a two-phase approach to be able to meet these different needs with minimal overhead. In phase 1, an ISAKMP SA is negotiated to create a secure, authenticated channel between the two hosts. The ISAKMP SA is a single, bidirectional security association. In phase 2, the SAs for the individual type of traffic (one SA for each direction) are negotiated using the authenticated channel established in phase 1.

Due to the Diffie-Hellman key exchange and the authentication, phase 1 is computationally rather expensive. Phase 2 does not involve key exchange nor authentication and is much less expensive. Performing phase 1 just once for a pair of hosts and then multiple phase 2 operations for the individual connections is a concept that can improve performance considerably.

Identity protection

In phase 1, certificates and authentication data are exchanged between the hosts. IKE offers *identity protection*, meaning that all information that could identify a host to an attacker or eavesdropper can be encrypted. Depending on whether identity protection is really required, IKE supports two modes for phase 1: *main mode* offers identity protection, while *aggressive mode* does not. In main mode, a shared, secret key is established before the identification information (for instance, the host's digital certificate) is sent. For a diagram showing IKE main mode, see Figure A-15.

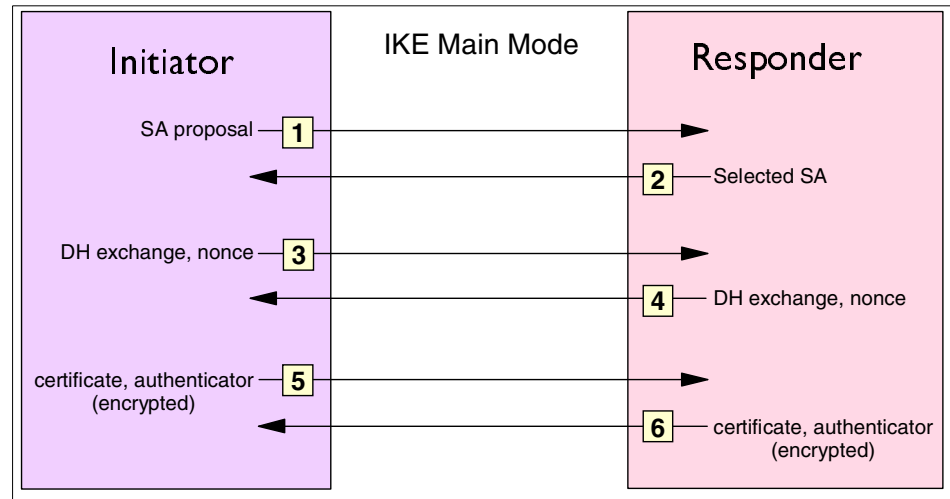


Figure A-15 IKE phase 1 main mode

Aggressive mode does not require the DH key exchange to be completed before sending the remaining information. Therefore, there is only one exchange of messages in aggressive mode (see Figure A-16).

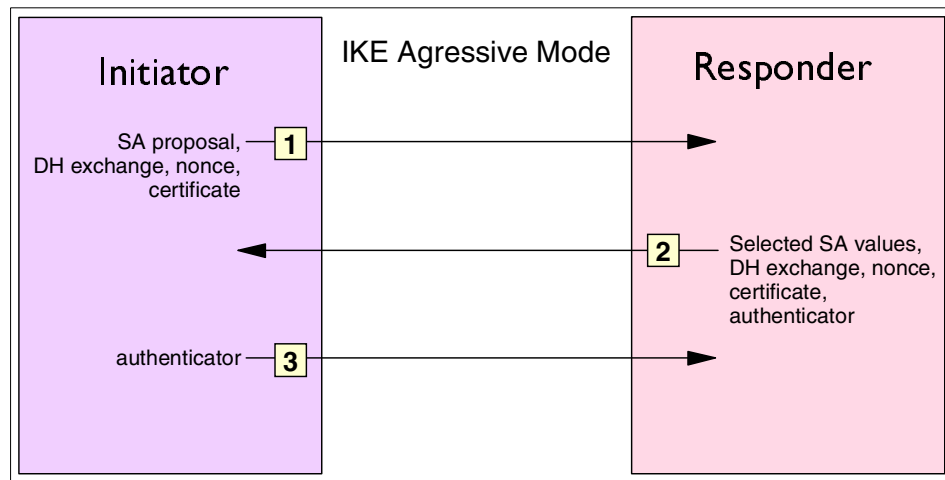


Figure A-16 IKE phase 1 aggressive mode

The exchange of messages taking part in phase 2 (negotiation of the SAs for the individual type of traffic) is called *quick mode*. In this mode, the pair of SAs for the intended type of communication is set up. The required keys for encryption and message authentication are generated from the shared key obtained in phase 1.

Transmitting data with IPSec

When a host wants to transmit one or more packets to another host it had not contacted before, it will perform the necessary IKE exchanges to set up the required SAs with the other hosts. Once this has all been performed and the necessary keys are generated, the host proceeds to send the first packet.

IPSec has two formats for sending data, which serve slightly different purposes. *Authentication Header (AH)* provides for message authentication and replay protection, whereas *Encapsulating Security Payload (ESP)* provides for data encryption in addition to message authentication and replay protection. The SA for a communication selects whether AH, ESP, or a combination of both is to be used.

Depending on the type of VPN connection between the two hosts, there are two modes, *tunnel mode* and *transport mode*, that are to be used.

ESP and AH in transport mode

If a VPN connection is being established between two hosts that are the endpoints for the packets transmitted between them, transport mode should be used. Figure A-17 shows the format of an Authentication Header (AH) in transport mode.

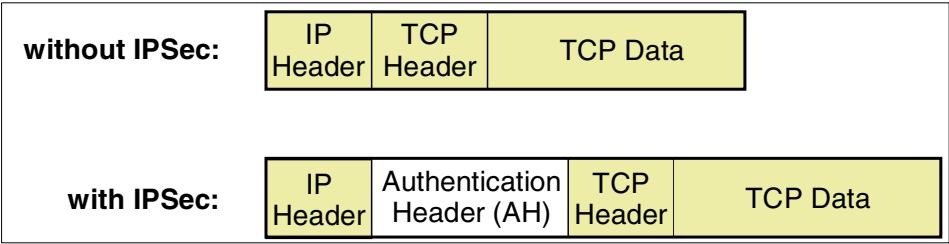


Figure A-17 AH in transport mode

The message authentication applied by AH protects the parts of the packet that are shaded in Figure A-17. Note that although the IP header is shaded in the diagram, parts of it are not authenticated because they can change in transit between sender and receiver.

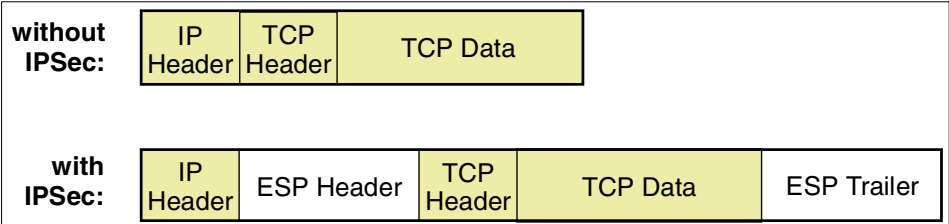


Figure A-18 ESP in transport mode

With the Encapsulating Security Payload (ESP) format in transport mode, the TCP header and data are encrypted and, optionally, authenticated. But as can be seen in Figure A-18 (the protected areas are shaded), the IP header is afforded no protection at all. However, this should not be a problem because sending and receiving hosts have been authenticated and verified in the SA.

ESP in tunnel mode

A common application of VPNs is the use of a protected tunnel between two secure networks. IPSec-capable firewalls at each end of the tunnel encrypt the packets they send from the secure network through the tunnel; they decrypt the packets they receive from the tunnel and route them to the destination hosts. In this scenario, the SAs do not authenticate the destination hosts (just the firewalls) and an attacker's modification of the IP headers could go undetected.

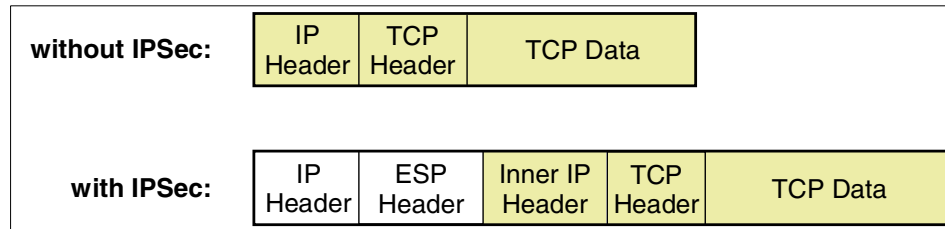


Figure A-19 ESP in tunnel mode

In this environment, tunnel mode is to be used. Figure A-19 shows the format of ESP packets in this mode; again, protected areas are shaded. The complete original packet, including the original IP header, is used as payload for an ESP packet. The inner IP header has the address of the destination host while the outer IP header addresses the firewall at the end of the tunnel. In this way, the complete packet including the IP header is protected.

In some cases, the AH and ESP formats are combined (applied one after the other) in order to reap both the benefits of IP header authentication with AH and payload (data) encryption with ESP.

For detailed information, read:

- ▶ *A Comprehensive Guide to Virtual Private Networks, Volume I: IBM Firewall, Server and Client Solutions*, SG24-5201
- ▶ *Secure e-business in TCP/IP Networks on OS/390 and z/OS*, SG24-5383

Alternative VPN solutions: Layer 2 Tunnel Protocol

A remote access dial-up solution for mobile users is a very simple form of a virtual private network, typically used to support dial-in access to a corporate network whose users are all company employees. To eliminate the long-distance charges that would occur if a remote user were to dial in directly to a gateway on the home network, the IETF developed a tunneling protocol, Layer 2 Tunnel Protocol (L2TP). This protocol extends the span of a PPP connection: instead of beginning at the remote host and ending at a local ISP's point of presence, the virtual PPP link now extends from the remote host all the way back to the corporate gateway. In effect, the remote host appears to be on the same subnet as the corporate gateway.

Since the host and the gateway share the same PPP connection, they can take advantage of PPP's ability to transport protocols other than just IP. For example, L2TP tunnels can be used to support remote LAN access as well as remote IP access. Figure A-20 outlines a basic L2TP configuration:

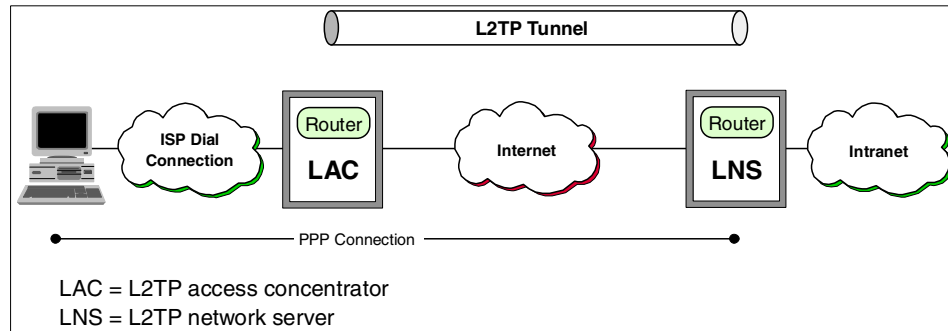


Figure A-20 Layer 2 Tunnel Protocol (L2TP) scenario

Although L2TP provides cost-effective access, multi-protocol transport, and remote LAN access, it does not provide cryptographically robust security features. For example:

- ▶ Authentication is provided only for the identity of tunnel endpoints, but not for each individual packet that flows inside the tunnel. This can expose the tunnel to various attacks.
- ▶ Without per-packet integrity, it is possible to mount denial-of-service attacks by generating bogus control messages that can terminate either the L2TP tunnel or the underlying PPP connection.
- ▶ L2TP itself provides no facility to encrypt user data traffic. This can lead to embarrassing exposures when data confidentiality is an issue.
- ▶ While the payload of the PPP packets can be encrypted, the PPP protocol suite does not provide mechanisms for automatic key generation or for automatic key refresh. This can lead to someone listening in on the wire to finally break that key and gain access to the data being transmitted.

Secure Sockets Layer

Secure Sockets Layer (SSL) is a protocol developed by the Netscape Communications Corporation that uses encryption to provide privacy and authentication between two applications using TCP/IP. SSL can be regarded as a *transport layer* equivalent of IPsec. Like IPsec, it uses asymmetric cipher algorithms (RSA is normally used) to authenticate users and sign messages, and symmetric algorithms to ensure confidentiality. Unlike IPsec, it is used to protect sessions between particular applications on particular ports; IPsec provides blanket protection between two hosts.

HTTP can use SSL to secure its communications. This allows Web browsers and servers to pass confidential or sensitive data through the Internet or intranet. SSL is also implemented by the Lightweight Directory Access Protocol (LDAP) for secure connections between LDAP clients and LDAP servers, by Telnet, and by a Telnet client such as Host On-Demand for connections between the client and the host system.

SSL overview

SSL was originally developed to protect traffic between a client and a server communicating across the Internet. The latest version of SSL from Netscape (and final version from Netscape) is SSL 3.0. At time of writing, it is by far the most commonly used SSL protocol. According to the latest SSL standard (RFC 2246, *The TLS Protocol Version 1.0*) SSL 2.0 should be phased out “with all due haste”. The IETF TLS-Based Telnet Security document (see “Transport Layer Security Protocol (TLS)” on page 1023) goes a step further to say that SSL 2.0 is not an acceptable protocol at all. See Figure A-21 on page 1020 for an outline of some of the SSL protocols and standards.

The use of SSL for Web access is through a protocol called HTTPS. HTTPS is a unique protocol that combines SSL and HTTP. You need to specify `https://` instead of `http://` as an anchor in HTML documents that link to SSL-protected documents. A client user can also open a URL by specifying `https://` to request SSL-protected documents.

Because HTTPS and HTTP are different protocols and use different ports (the default ports are 443 and 80, respectively), you can run both SSL and non-SSL requests at the same time. As a result, you can elect to provide information to all users using no security, and specific information only to browsers that make secure requests. This is how a retail company on the Internet can allow users to look through the merchandise without security, but then fill out order forms and send their credit card numbers using security.

SSL relies on digital certificates and a hierarchy of trusted authorities, as described in “Digital certificates” on page 1001, to ensure authentication of clients or servers.

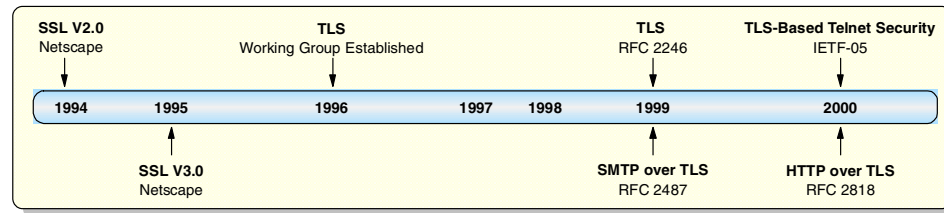


Figure A-21 Evolution of SSL

Establishing secure communications with SSL

To use SSL, both the client and the server need to have the software to support this protocol. Because SSL started with HTTP communication, it is used as an illustration.

The latest Netscape and Microsoft browsers support SSL 3.0 and all its features on the client. SSL is composed of two sub-protocols:

- ▶ SSL Handshake Protocol
- ▶ SSL Record Protocol

The SSL Handshake Protocol initializes a secure session, with authentication of the server (and optionally, the client), agreement of encryption scheme, and transfer of encryption keys. A public-key algorithm, usually RSA, is used for the exchange of the symmetric encryption key and for digital signatures. With the server certificate, the client is also able to verify the server's identity. With SSL Version 3.0, the possibility of authenticating the client identity by using client certificates in addition to server certificates was added. The overall flow of these steps is shown in Figure A-22.

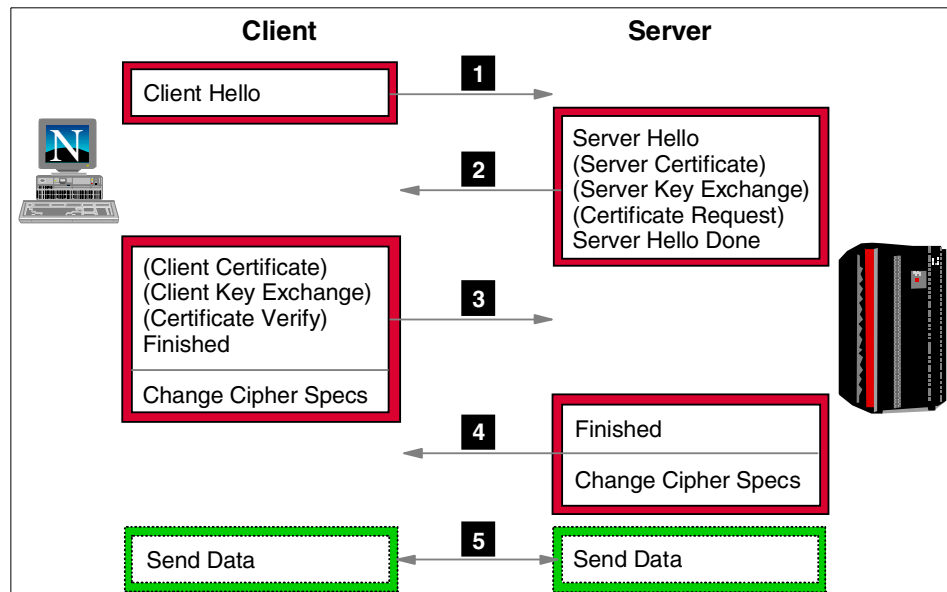


Figure A-22 Overview of SSL Handshake Protocol

- **Step 1:** The client sends a connection request with a `client hello` message. This message includes:
 - Desired version number.
 - Time information (the current time and date in standard UNIX 32-bit format).
 - Optionally session-ID. If it is not specified the server will try to resume previous sessions or return an error message
 - Cipher suites. (List of the cryptographic options supported by the client. These are authentication modes, key exchange methods, encryptions and MAC algorithms.)
 - Compression methods supported by the client.
 - A random value (nonce). A nonce is a random value used in communication protocols, typically for replay protection.
- **Step 2:** The server evaluates the parameters sent by the `client hello` message and returns a `server hello` message that includes the following parameters which were selected by the server to be used for the SSL session:
 - Version number
 - Time information (the current time and date in standard UNIX 32-bit format)
 - Session ID
 - Cipher suite
 - Compression method

- A random value

Following the server hello message the server sends the following messages:

- Server certificate if the server is required to be authenticated
- A server key exchange message if there is no certificate available or the certificate is for signing only
- A certificate request if the client is required to be authenticated

Finally, the server sends a server hello done message and begins to wait for the client response.

- ▶ Step 3: The client sends the following messages:
 - If the server has sent a certificate request, the client must send a certificate or a no certificate message.
 - If the server has sent a server key exchange message, the client sends a client key exchange message based on the public key algorithm determined with the hello messages.
 - If the client has sent a certificate, the client verifies the server certificate and sends a certificate verify message indicating the result.

The client then sends a finished message indicating the negotiation part is completed. The client also sends a change cipher spec message to generate shared secrets. It should be noted that this is not controlled by the handshake protocol, the change cipher spec protocol manages this part of the operation.

- ▶ Step 4: The server sends a finished message indicating the negotiation part is completed. The server then sends the change cipher spec message.
- ▶ Step 5: Finally, the session partners separately generate an encryption key, in which they derive the keys to use in the encrypted session that follows from the master key. The Handshake protocol changes the state to the connection state. All data taken from the application layer is transmitted as special messages to the other party.

The SSL Record Protocol transfers application data using the encryption algorithm and keys agreed upon during the handshake phase. As explained above, symmetric encryption algorithms are used, because they provide much better performance than asymmetric algorithms.

SSL considerations

As discussed, security functions such as SSL are needed to send sensitive data safely if you connect your system to an insecure network such as the Internet. On the other hand, using such security functions has performance impacts, including utilizing additional CPU cycles and degrading Web server performance.

Furthermore, SSL does not satisfy every security requirement. While it protects against eavesdropping and alteration of data, it cannot protect the server from an attacker masquerading as a trusted user. For these security concerns, the risk can be minimized by the use of access controls or firewalls.

To maintain SSL security you have to manage the key carefully, especially when using self-signed certificates, because the whole system environment is affected by the security of the Certificate Authority's key database.

Transport Layer Security Protocol (TLS)

Continued development of the SSL protocol moved into the hands of the Internet Engineering Task Force in 1996. The result was that SSL 3.0 evolved into Transport Layer Security, RFC 2246.

So, what exactly is new in TLS? The protocol syntax and handshake flow remains virtually unchanged. The significant difference is that the hello message for TLS must contain Version 3.1. Once it has been agreed by both client and server that 3.1 is to be used, cipher suite exchanges will use a prefix of TLS_ instead of the SSL 3.0 prefix of SSL_.

Telnet-negotiated sessions

Host On-Demand and Personal Communications Version 5.6 support Telnet-negotiated sessions. Telnet-negotiated session protocol is based on an IETF Internet draft that allows the negotiation of the secure protocol (SSL) prior to establishing the Telnet connection. This draft allows Telnet servers that support SSL V3.0, but not the full TLS RFC (RFC 2246), to negotiate a secure SSL connection.

How does this Internet draft work? It adds a new IAC (Interpret As Command) option and sub-option. The START_TLS option allows the client (WILL START_TLS) or the server (DO START_TLS) to initiate a request for a secure session. Once TLS has been agreed upon, the session immediately drops into negotiation of either SSL or TLS. Negotiation of other Telnet IAC options is suspended until the security negotiation has successfully completed.

If the client and server cannot agree upon the START_TLS option, then the Telnet server can opt to drop into native TLS/SSL security negotiation (identified by CONNTYPE SECURE in the TCP/IP profile data set).

Why add a new IAC option? The foremost advantage is that this option places the control of session security into the TN3270 world (instead of leaving it up to the transport layer). If a Telnet client won't accept a DO START_TLS option, the Telnet server can choose to end the session (CONNTYPE NEGTSURE in the TCP/IP profile data set).

The other significant advantage of placing encryption negotiation into the TN3270 option data stream is that a single port can be used for encrypted and non-encrypted sessions. Prior to negotiated Telnet, a separate port for secure and non-secure sessions had to be used. Since all TN3270 clients default to port 23, this was not an ideal situation.

A typical negotiated Telnet SSL flow is shown in Figure A-23.

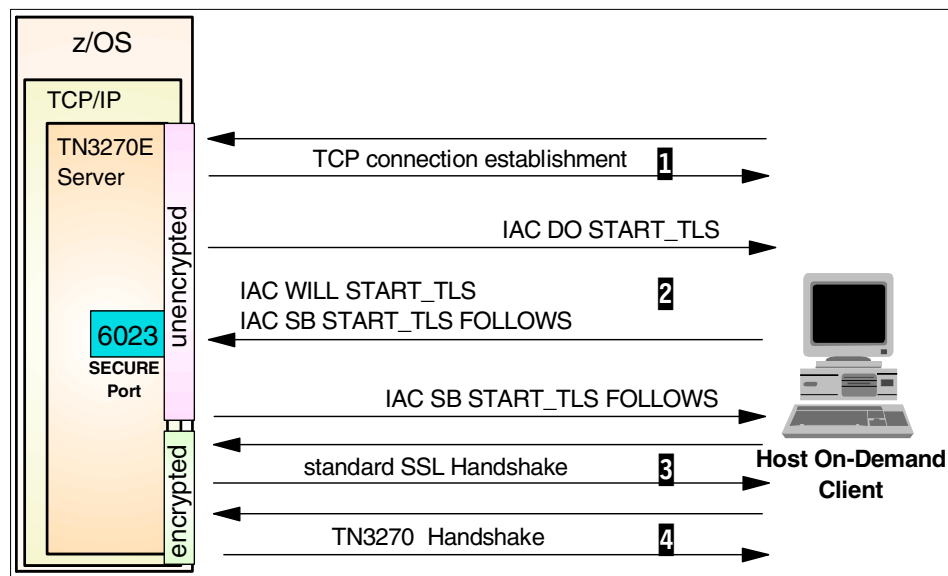


Figure A-23 Telnet-negotiated security session negotiation

1. IP connection establishment.
2. The Telnet server sends the IAC DO START_TLS command to the client to verify if it wants to perform the SSL negotiation.
3. If a positive response is received, then Telnet begins a normal SSL handshake.
4. If no positive response is received, the connection will be dropped.

The IAC DO START_TLS Telnet command, sent from the server, activates TLS at the beginning of a Telnet connection. The client can respond to this command by sending the IAC WILL START_TLS command, if the negotiation of a TLS connection is required. With the IAC DONT START_TLS command, the client can refuse the TLS connection negotiation. Sending the IAC SB START_TLS FOLLOWS IAC SE command initiates a TLS negotiation. When this sub-command has been sent and received, the TLS negotiation will begin.

If Enable Security (SSL) is Yes and Telnet-negotiated is Yes, then the Telnet connection will be started normally without SSL. However, the 3270 session will not start until the SSL negotiation completes successfully. If the server WONT STARTTLS, then the session will not start, and an error message will be issued stating Security was requested, but the server does not support security.

If Enable Security (SSL) is No and the server requests to start the session using Telnet-negotiated security, Host On-Demand will not start the session and an error message will be displayed on the status bar stating The server requested security, but Security is not enabled.

Session configuration

In order to implement Telnet-negotiated security you must first select **Enable Security (SSL)** to activate the Telnet-negotiated radio button, then select **Yes** to the Telnet-negotiated radio button as shown in Figure A-24.

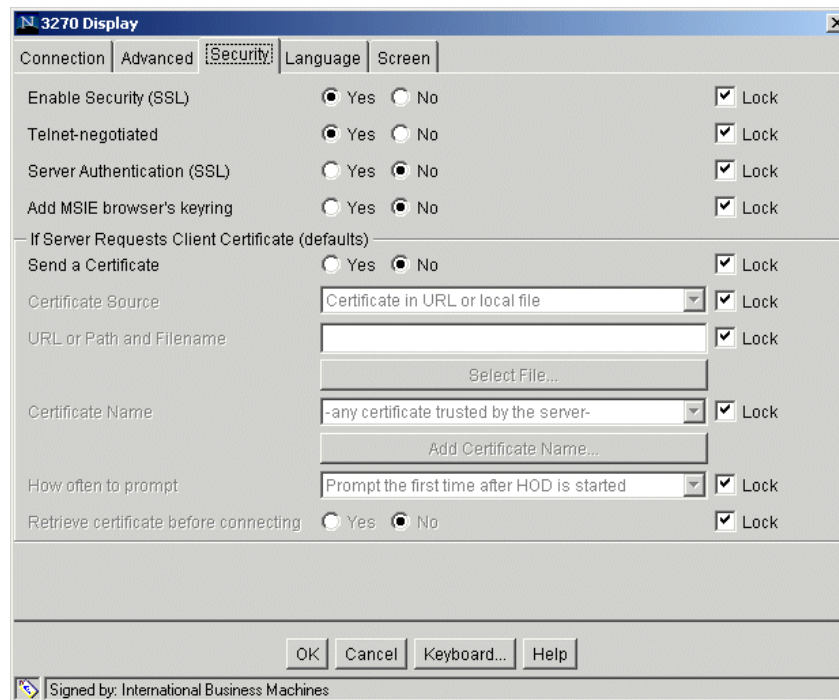


Figure A-24 Enable Telnet-negotiated security

Selecting Telnet-negotiated determines if the SSL negotiation between the client and the server is done on the Telnet connection or on an SSL connection prior to the Telnet negotiations. The other SSL options are valid regardless of whether the Telnet-negotiation is enabled.

If **Yes** is selected, then the Telnet protocol will be used to negotiate the SSL security after the Telnet connection is established. This support is only applicable with a Telnet server which supports Telnet-negotiated Security. Communications Server for OS/390 V2R10, or above, is the only IBM Telnet Server at this time which supports this function.

If **No** is selected, the traditional SSL negotiations will be done on an SSL connection with the server, and subsequently the Telnet negotiations with the server will be done. The default is **No**.

There will be no migration considerations since this is not supported by Personal Communications Manager, and the default is **No**.

The Communications Server for OS/390 documentation refers to this feature as “negotiable SSL”.

**B**

Service Location Protocol

Service Location Protocol (SLP) is defined in Request for Comments (RFC) 2165. It is a service-discovery method for TCP/IP-based communications, providing a simple and lightweight protocol for automatic advertisement and maintenance of intranet services and minimizing the use of broadcast and multicast in the network. SLP uses multicast, which targets a group of nodes, unlike broadcast, which targets all nodes. The benefit of multicast is that it sends one packet that all members of the group receive but that only the intended recipients read. A multicast packet is not isolated to a local segment; routers can forward it to whatever subnets are attached.

Specialized components called *agents* perform tasks and support services:

User Agent	Support for service query functions. It acquires/requests service information for user applications.
Service Agent	Service registration and service advertisement.
Directory Agent	Collects service information from Service Agents that is later requested by User Agents in intranets.

Services are described by the configuration of attributes associated with a type of service. A User Agent can select an appropriate service by specifying the attribute that it needs, in a service request. When the service request is returned, it contains a Uniform Resource Locator (URL) pointing to the service desired, and other information needed by the User Agent.

The Host On-Demand client is the User Agent; Communications Server for Windows NT and Windows 2000 or Communications Server for AIX is the Service Agent.

SLP can reduce overall network traffic by using scopes to manage client service requests. A scope is essentially a grouping method to organize servers into named groups. Scope values are defined by a network administrator, and may represent departments, regions or organizations. If desired, different scopes can be assigned for different services provided on the server.

Load balancing

Communications Server for Windows NT and Windows 2000 or Communications Server for AIX provide information about the server load, using SLP, by calculating the percentage of available resources. For LUA sessions, such as 3270 sessions, the load percentage is the number of active application connections divided by the total number of LUs available.

The Host On-Demand client gets the load percentage through SLP, determines which server is the least loaded, and attempts a connection to that server.

In Figure B-1, the client has three servers available for the client connection in the named scope, hodscope. The HOD client will attempt to connect to Server A because, according to the load values returned, it is the least loaded of the three. If the connection is successful, the load of Server A has increased. If this increase means that Server A is no longer the least loaded, the next SLP client will realize that and connect to a different server.

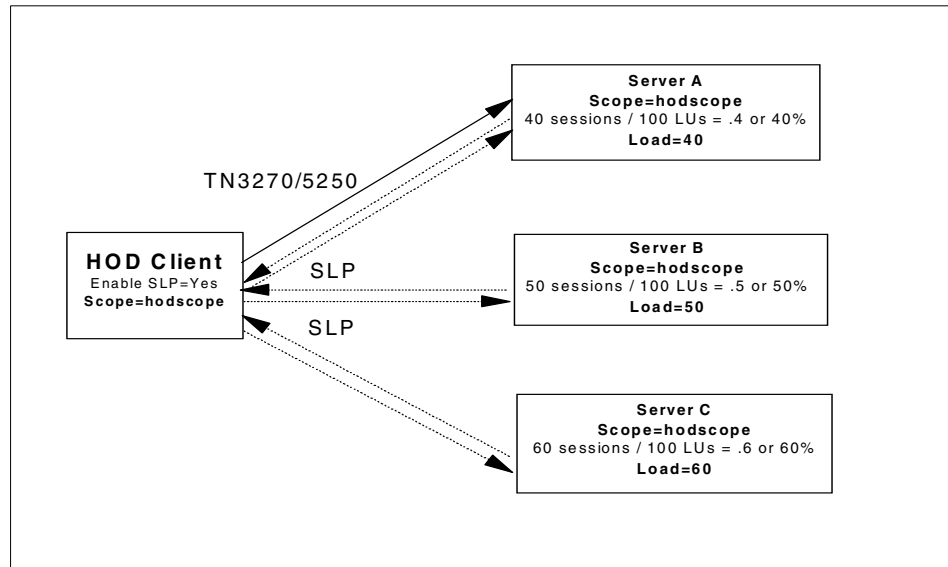


Figure B-1 Host On-Demand load balancing with SLP

Warm standby

Warm standby is available for TN3270 and TN5250 sessions through SLP. If the current connection fails, and if the client's Auto-Reconnect option is set to Yes, the client queries the servers again and connects to the least loaded.

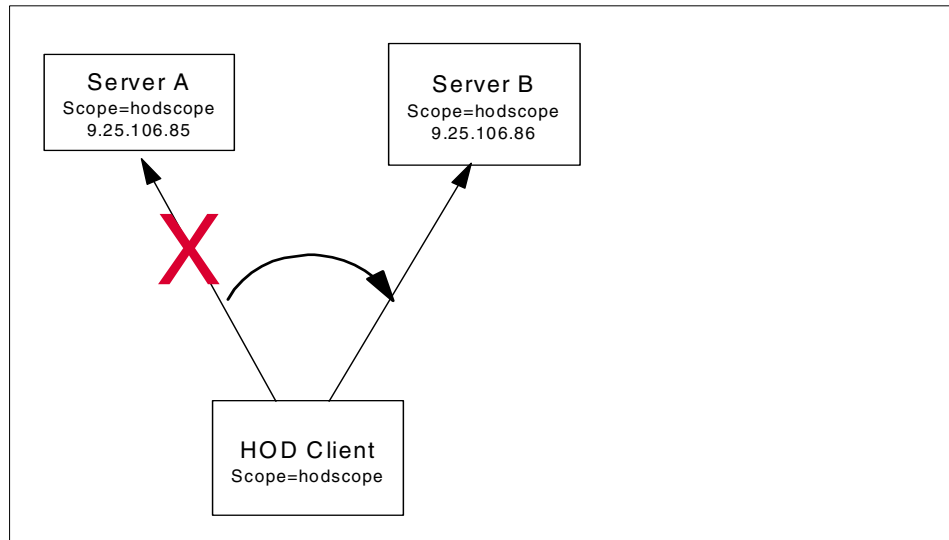
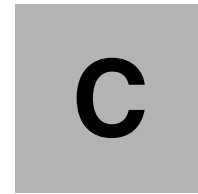


Figure B-2 Host On-Demand warm standby with SLP



An example of MacroIOProvider

```
//*****
// MacroIOProvider implementation...
//
// For this demo, the IO Provider will just maintain the macros in memory.
// Macros will be alive only for the duration of the applet. Use this code
// as a base to write your macros out to your server or local disk. Please
// pay attention to what file privileges you have when doing so.
//
// MacroManager provides a default MacroIOProvider that saves the macros to
// wherever the class was loaded from. This works well if the bean is used
// in an application, or if it used in an applet with Navigator 4.x
// and Internet Explorer 4.x, which should have the right privileges enabled.
//*****

/*****
 * Saves a macro to persistent storage. The macro is supplied in the form of
 * a property which contains the macro name, description, and source code
text.
 *
 * @param prop A properties object representing the macro to be saved.
 * @see #getMacro(String)
 */
public void putMacro(Properties p)
{
```

```

Properties props = null;
String nameToSave = (String)p.get(Macro.NAME);

// Search through list and remove if already exists
for (int i = 0; i < macroList.size(); i++) {
    props = (Properties)macroList.elementAt(i);
    if (((String)props.get(Macro.NAME)).equals(nameToSave)) {
        macroList.removeElement(props);
        break;
    }
}

// Add the element
macroList.addElement(p);
}

/*****
 * Retrieves a macro from persistent storage. The macro name is supplied and
 * the macro is returned in the form of a properties object which
 * contains the macro name, description, and source code text.
 *
 * @param name String containing the name of the macro to be retrieved.
 * @see #putMacro(Properties)
 */
public Properties getMacro(String name)
{
    Properties retProps = null;

    // Search through list for name and return if there, null by default
    for (int i = 0; i < macroList.size(); i++) {
        retProps = (Properties)macroList.elementAt(i);
        if (((String)retProps.get(Macro.NAME)).equals(name))
            return retProps;
    }

    return null;
}

/*****
 * Deletes a macro from persistent storage.
 *
 * @param name String containing the name of the macro to be deleted.
 */
public void removeMacro(String name)
{
    Properties retProps = null;

    // Search through list and remove the macro
    for (int i = 0; i < macroList.size(); i++) {

```

```

        retProps = (Properties)macroList.elementAt(i);
        if (((String)retProps.get(Macro.NAME)).equals(name)) {
            macroList.removeElement(retProps);
            break;
        }
    }
}

/*****
 * Returns a list of all the macros in persistent storage. The returned
 * Vector should contain a set of Properties object, each of which contains
 * (at a minimum) a macro name and description. The properties objects
 * do not need to (but may) contain the macro source code text. The
 * MacroManager will issue a getMacro() call to retrieve the macro source
 * text when required.
 *
 * @see #getMacro(String)
 */
public Vector listMacros()
{
    return macroList;
}

```




Additional material

This redbook refers to additional material that can be downloaded from the Internet as described below.

Locating the Web material

The Web material associated with this redbook is available in softcopy on the Internet from the IBM Redbooks Web server. Point your Web browser to:

<ftp://www.redbooks.ibm.com/redbooks/SG246182>

Alternatively, you can go to the IBM Redbooks Web site at:

ibm.com/redbooks

Select the **Additional materials** and open the directory that corresponds with the redbook form number, SG246182.

Using the Web material

The additional Web material that accompanies this redbook includes two separate, working demonstrations of using the Session Manager API to enhance a host application. One demo is for a Java 1 browser environment while the other is for Java 2. There are two separate ZIP files:

<i>File name</i>	<i>Description</i>
JSDEmoEJ1.zip	Zipped code samples for a Java 1 environment
JSDEmoEJ2.zip	Zipped code samples for a Java 2 environment

Java 1 demo (JSDEmoEJ1.zip)

The following files are in the hostondemand\HOD directory:

JSAPIDemoJ1.html	file created by the Deployment Wizard; the administrator has selected Java 1 as the client Java type
JSDEmoHelp.html	this file opens in the bottom frame when users click the Demo Help button; it shows you how to run the demonstration and to learn more about how it works
JSDEmoIEJ1.html	the main Web site; it divides the page into three frames, with 36% allocated to the first frame, 64% allocated to the second frame, and 0% allocated to the third frame
JSNavigation.html	this is what displays in the first frame of the main Web site; it contains all the navigation buttons
JSWelcome.html	this file opens in the bottom frame when users click the Welcome button; it provides a simple scenario that demonstrates how to store data in the Web page and send it to the Host On-Demand applet

The following files are in the hostondemand\HOD\HODData\JSAPIDemoJ1 directory:

Note: These files contain the session configuration information created by the administrator. Remember that when you create HTML files using the Deployment Wizard, a corresponding subdirectory is created with the same name under \HODData. In this case, the Deployment Wizard file is called JSAPIDemoJ1.html, so the corresponding subdirectory is also called \JSAPIDemoJ1.

cfg0.cf	contains configuration information for one of the three sessions defined by the administrator
cfg1.cf	contains configuration information for one of the three sessions defined by the administrator
cfg2.cf	contains configuration information for one of the three sessions defined by the administrator
params.txt	contains Host On-Demand configuration parameters
policy.obj	contains information about the Disabled Functions

preloads.obj	contains information about the objects to preload as defined on the Preload Options window
udparams.txt	user-defined HTML parameters
wInfo.txt	contains the responses to each window in the Deployment Wizard; used only by the Deployment Wizard

In addition, the JSDemoEJ1.zip file contains various .gif files that make up the images included in the demonstration files.

Java 2 demo (JSDemoEJ2.zip)

The following files are in the hostondemand\HOD directory:

JSAPIDemoJ2.html	file created by the Deployment Wizard; the administrator has selected Java 2 as the client Java type
JSDemoHelp.html	this file opens in the bottom frame when users click the Demo Help button; it shows you how to run the demonstration and to learn more about how it works
JSDemoIEJ2.html	the main Web site; it divides the page into three frames, with 36% allocated to the first frame, 64% allocated to the second frame, and 0% allocated to the third frame
JSNavigation.html	this is what displays in the first frame of the main Web site; it contains all the navigation buttons
JSWelcome.html	this file opens in the bottom frame when users click the Welcome button; it provides a simple scenario that demonstrates how to store data in the Web page and send it to the Host On-Demand applet
z_JSAPIDemoJ2.html	this file is generated for Java2 or Auto selection in Deployment Wizard; it is the final page that gets displayed after the Java2 and Auto detection is done

The following files are in the hostondemand\HOD\HODData\JSAPIDemoJ2 directory:

Note: These files contain the session configuration information created by the administrator. Remember that when you create HTML files using the Deployment Wizard, a corresponding subdirectory is created with the same name under \HODData. In this case, the Deployment Wizard file is called JSAPIDemoJ2.html, so the corresponding subdirectory is also called \JSAPIDemoJ2.

cfg0.cf	contains configuration information for one of the three sessions defined by the administrator
----------------	---

cfg1.cf	contains configuration information for one of the three sessions defined by the administrator
cfg2.cf	contains configuration information for one of the three sessions defined by the administrator
params.txt	contains Host On-Demand configuration parameters
policy.obj	contains information about the Disabled Functions
preloads.obj	contains information about the objects to preload as defined on the Preload Options window
udparams.txt	user-defined HTML parameters
wInfo.txt	contains the responses to each window in the Deployment Wizard; used only by the Deployment Wizard

In addition, the JSDemoEJ2.zip file contains various .gif files that make up the images included in the demonstration files.

System requirements for downloading the Web material

The following system configuration is recommended:

Hard disk space: 250KB minimum
Operating System: Windows 2000 with Internet Explorer

How to use the Web material

You can extract the contents of the demonstration zip files (JSDemoEJ1.zip and JSDemoEJ21.zip) either directly into the Host On-Demand publish directory, or you can extract them into a separate directory and manually copy the files into the publish directory. The files must be in the publish directory in order for the demonstration to work properly.

The Java 1 and Java 2 demonstrations can co-exist in the same directory. Note that the two demonstrations have several files in common, so if you unzip both demonstrations into the same directory, you will be asked if you want to overwrite these files. You can accept the overwrite.

Related publications

The publications listed in this section are considered particularly suitable for a more detailed discussion of the topics covered in this redbook.

IBM Redbooks

For information on ordering these publications, see “How to get IBM Redbooks” on page 1044.

- ▶ *IBM Host Integration in a Secure Network: A Practical Approach*, SG24-5988
- ▶ *Programming with the Host Access APIs*, SG24-5856
- ▶ *Personal Communications Version 5.6 Version 4.3 for Windows 95,98 and NT*, SG24-4689
- ▶ *Java 2 Network Security* by M. Pistoia et al., June 1999, IBM Form Number: SG24-2109-01, ISBN: 0-130-15592-6
- ▶ *IBM Communications Server for OS/390 TCP/IP 2000 Update Technical Presentation Guide*, SG24-6162
- ▶ *IBM Communication Server for OS/390 V2R10 TCP/IP Implementation Guide Volume 1*, SG24-5227
- ▶ *V4 TCP/IP for AS/400: More Cool Things Than Ever Redbook*, SG24-5190
- ▶ *AS/400 HTTP Server Performance and Capacity Planning Redbook*, SG24-5645
- ▶ *Understanding LDAP*, SG24-4986
- ▶ *LDAP Implementation Cookbook*, SG24-5110
- ▶ *Using LDAP for Directory Integration: A Look at IBM SecureWay Directory, Active Directory, and Domino*, SG24-6163
- ▶ *Deploying a Public Key Infrastructure*, SG24-5512
- ▶ *WebSphere V3.5 Handbook*, SG24-6161
- ▶ *TCP/IP Tutorial and Technical Overview*, GG24-3376
- ▶ *A Comprehensive Guide to Virtual Private Networks, Volume I: IBM Firewall, Server and Client Solutions*, SG24-5201
- ▶ *Secure e-business in TCP/IP Networks on OS/390 and z/OS*, SG24-5383

Other resources

These publications are also relevant as further information sources:

- ▶ *OS/390 V2R10 IBM Communication Server for IP Configuration Reference*, SC31-8726
- ▶ *OS/390 SecureWay Security Server LDAP Client Application Development Guide and Reference*, SC24-5878
- ▶ *OS/390 SecureWay Security Server LDAP Server Administration and Usage Guide*, SC24-5861
- ▶ *z/OS SecureWay Security Server LDAP Server Administration and Usage Guide*, SC24-5923
- ▶ *z/OS V1R2 IBM Communication Server for IP Configuration Reference*, SC31-8776
- ▶ *z/OS V1R1.0 CS: IP Migration*, SC31-8773
- ▶ *OS/390 SMP/E User's Guide*, SC28-1740
- ▶ *OS/390 SMP/E Reference*, SC28-1806
- ▶ *MVS Initialization and Tuning Reference*, SA22-7592
- ▶ *OS/390 eNetwork Communications Server IP Planning and Migration Guide Version 2 Release 6*, SC31-8512-01
- ▶ *OS/390 eNetwork Communications Server IP Planning and Migration Guide Version 2 Release 7*, SC31-8512-02
- ▶ *OS/390 SecureWay Communications Server IP Planning and Migration Guide Version 2 Release 8*, SC31-8512-03
- ▶ *OS/390 V2R10.0 IBM Communications Server IP Migration Guide*, SC31-8512-06
- ▶ Host On-Demand online documentation:
 - Host Printing Reference*
 - Getting Started*
 - ReadMe*
- ▶ DB2 online documentation
 - Application Building Guide*
- ▶ *DEC VT220 Programmer Reference Manual*
- ▶ Personal Communications online documentation:
 - Quick Beginnings*
 - CD-ROM Guide to Installation*

Administrator Guide and Reference

Host Access Class Library

Client/Server Communications Programming

Emulator Programming

- ▶ *Cracking DES: Secrets of Encryption Research, Wiretap Politics & Chip Design*, by Electronic Frontier Foundation, John Gilmore (Editor), 1988
- ▶ *The Oakley Key Determination Protocol*, by H. Orman; RFC 2412, November 1998
- ▶ *SKEME: A Versatile Secure Key Exchange Mechanism for the Internet*, by H. Krawczyk; IEEE Proceedings, 1996
- ▶ *Java Security: 2nd Edition* by Scott Oaks, May 2001, O'Reilly & Associates, Inc. ISBN: 0-596-00157-6

Referenced Web sites

These Web sites are also relevant as further information sources:

- ▶ Host On-Demand Internet home page
<http://www.ibm.com/software/webservers/hostondemand>
- ▶ Host On-Demand EHLLAPI Bridge
<http://www.ibm.com/software/webservers/hostondemand/downloads/ehllapi/hodv5.html>
- ▶ IBM Express Logon whitepaper
<ftp://ftp.software.ibm.com/software/network/library/whitepapers/elf.pdf>
- ▶ The IBM public "IBM developer kit porting" Web page:
<http://www.ibm.com/developerworks/java/jdk/?dwzone=java>
- ▶ IBM Developerworks, Java home page
<http://www.ibm.com/developerworks/java/>
- ▶ IBM Developerworks, Java developer kits
<http://www.ibm.com/developerworks/tools.nsf/dw/java-devkits-byname>
- ▶ Setting up the Remote Abstract Window Toolkit for Java on a remote display
<http://publib.boulder.ibm.com/pubs/html/as400/v4r4/ic2924/info/java/rzaha/devkit.htm>
- ▶ The IBM Personal Communications Web page:

- <http://www.software.ibm.com/network/pcomm/>
- ▶ Netscape Preference Wrangler
<http://developer.netscape.com/library/technote/security/prefwrangler.html>
- ▶ Netscape Security Preferences for Communicator
<http://developer.netscape.com/library/technote/security/sectn3.html>
- ▶ Novell Developer home page
<http://www.developer.novell.com>
- ▶ IBM Performance Management for IBM @server iSeries
<http://www.ibm.com/servers/eserver/series/perfmgmt/resource.htm>
- ▶ IBM @server Support
<http://techsupport.services.ibm.com/eserver/fixes>
- ▶ IBM - Fixes for Netscape Communicator on AIX
<http://techsupport.services.ibm.com/aix/efixes/netscape/>
- ▶ IBM Product Publications for AS/400
<http://as400bks.rochester.ibm.com>
- ▶ IBM Software Internet Delivery
<http://www6.software.ibm.com/enetwork/isd/home.html>
- ▶ IBM Software Networking and Communications Support home pager
<http://www.ibm.com/software/network/support>
- ▶ IBM WebSphere Software Support Bulletin
<http://www.ibm.com/software/network/support/alert>
- ▶ IBM WebSphere Host On-Demand Support
<http://www.ibm.com/software/webservers/hostondemand/support.html>
- ▶ Host On-Demand documentation library
<http://www.ibm.com/software/webservers/hostondemand/library/>
- ▶ Verisign - Introduction to Cryptography
<http://www.verisign.com/client/about/introCryp.html>
- ▶ IBM Communications Server online library
<http://www.ibm.com/software/network/commserver/library/>
- ▶ Communications Server for OS/390 white papers

<http://www.ibm.com/software/network/commserver/library/whitepapers/cos390.html>

- ▶ IBM HTTP Server home page

<http://www.ibm.com/servers/eserver/series/software/http/services/apache.htm>

- ▶ IBM AS/400 Support for Windows Network Neighborhood API Mini-Guide

<http://www.ibm.com/servers/eserver/series/netserver/apiminiguide.htm>

- ▶ Troubleshooting AS/400 SSL Enabled Telnet server

http://www.as400.ibm.com/tstudio/tech_ref/tcp/telntssl/Index.htm

- ▶ IBM AS/400 TCP/IP Reference

http://www.as400.ibm.com/tstudio/tech_ref/tcp/indexfr

- ▶ IBM AS/400 Download TELNET exit program files

http://www.as400.ibm.com/tstudio/tech_ref/tcp/telex/telexdwn.htm

- ▶ IBM iSeries online documentation What's New

<http://publib.boulder.ibm.com/pubs/html/series/online/chgfrm.htm>

- ▶ IBM SecureWay Directory home page

<http://www.ibm.com/software/network/directory/>

- ▶ Netscape's SSL home page

<http://home.netscape.com/eng/ssl3/ssl-toc.html>

- ▶ Netscape Directory server

<http://enterprise.netscape.com/products/identsvcs/directory.html>

- ▶ Netscape Object Signing

<http://developer.netscape.com/docs/manuals/signedobj/trust/index.htm>

- ▶ IBM Screen Customizer support

<http://www.ibm.com/software/network/screencustomizer/support/>

- ▶ IBM Screen Customizer product information site:

<http://www.ibm.com/software/network/screencustomizer/>

- ▶ IBM Screen Customizer documentation library

<http://www.ibm.com/software/network/screencustomizer/library/>

- ▶ The Sun Microsystems product home page

<http://java.sun.com/products/>

- ▶ Java 2 Plug-in tutorial

<http://java.sun.com/products/plugin/>

- ▶ Java Policy File Creation and Management Tool

<http://java.sun.com/j2se/1.3/docs/tool docs/win32/policytool.html>

- ▶ Code Signing for Java Applets

http://www.suitable.com/Doc_CodeSigning.shtml

- ▶ The Swing Connection: Mixing Heavy and Light Components

<http://java.sun.com/products/jfc/tsc/articles/mixing/index.html>

- ▶ Cryptanalysis of MD5 Compress

<http://www.cs.ucsd.edu/users/bsy/dobbertin.ps>

- ▶ How Internet Explorer Java virtual machine searches for classes

<http://support.microsoft.com/default.aspx?scid=kb;EN-US;q177168>

- ▶ The Internet Engineering Task Force home page

<http://www.ietf.org>

How to get IBM Redbooks

Search for additional Redbooks or drafts, view, download, or order hardcopy from the Redbooks Web site:

ibm.com/redbooks

Also download additional materials (code samples or diskette/CD-ROM images) from this Redbooks site.

Drafts are Redbooks in progress; not all Redbooks become Drafts and sometimes just a few chapters will be published this way. The intent is to get the information out much quicker than the formal publishing process allows.

IBM Redbooks collections

Redbooks are also available on CD-ROMs. Click the CD-ROMs button on the Redbooks Web site for information about all the CD-ROMs offered, as well as updates and formats.

Special notices

References in this publication to IBM products, programs or services do not imply that IBM intends to make these available in all countries in which IBM operates. Any reference to an IBM product, program, or service is not intended to state or imply that only IBM's product, program, or service may be used. Any functionally equivalent program that does not infringe any of IBM's intellectual property rights may be used instead of the IBM product, program or service.

Information in this book was developed in conjunction with use of the equipment specified, and is limited in application to those specific hardware and software products and levels.

IBM may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to the IBM Director of Licensing, IBM Corporation, North Castle Drive, Armonk, NY 10504-1785.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact IBM Corporation, Dept. 600A, Mail Drop 1329, Somers, NY 10589 USA.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The information contained in this document has not been submitted to any formal IBM test and is distributed AS IS. The use of this information or the implementation of any of these techniques is a customer responsibility and depends on the customer's ability to evaluate and integrate them into the customer's operational environment. While each item may have been reviewed by IBM for accuracy in a specific situation, there is no guarantee that the same or similar results will be obtained elsewhere. Customers attempting to adapt these techniques to their own environments do so at their own risk.

Any pointers in this publication to external Web sites are provided for convenience only and do not in any manner serve as an endorsement of these Web sites.

The following terms are trademarks of other companies:

C-bus is a trademark of Corollary, Inc. in the United States and/or other countries.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Sun Microsystems, Inc. in the United States and/or other countries.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States and/or other countries.

PC Direct is a trademark of Ziff Communications Company in the United States and/or other countries and is used by IBM Corporation under license.

ActionMedia, LANDesk, MMX, Pentium and ProShare are trademarks of Intel Corporation in the United States and/or other countries.

UNIX is a registered trademark in the United States and other countries licensed exclusively through The Open Group.

SET, SET Secure Electronic Transaction, and the SET Logo are trademarks owned by SET Secure Electronic Transaction LLC.

Other company, product, and service names may be trademarks or service marks of others

Abbreviations and acronyms

ACPI	Advanced Configuration and Power Interface	DCAS	Digital Certificate Access Server
AES	Advanced Encryption Standard	DDE	dynamic data exchange
AFP	Advanced Function Printer	DEC	Digital Equipment Corporation
APAR	authorized program analysis report	DES	Data Encryption Standard
API	application programming interface	DLUR	dependent LU requestor
APPN	Advanced Peer-to-Peer Networking	DMZ	demilitarized zone
ATI	automatic transaction initiation	DN	distinguished name
AWT	Abstract Windowing Toolkit	DPI	dots per inch
BMS	basic mapping support	EBCDIC	extended binary coded data interchange code
CA	Certificate Authority	ECLPS	Emulator Class Library Presentation Services
CAB	cabinet	EHLLAPI	enhanced high level language application programming interface
CAPI	cryptographic application programming interface	ELF	Express Logon Feature
CBPDO	Custom Built Product Delivery Offering	ELP	enhanced local preferences
CD	compact disc	EPI	External Presentation Interface
CECI	Command Level Interpreter Transaction	ESP	Encapsulating Security Payload
CICS	Customer Information Control System	FTP	file transfer protocol
CSD	corrective service delivery, corrective service distribution, corrective service diskette	GDI	graphical device interface
CSI	Consolidated Software Inventory	GID	Group Identification Number
DASD	Direct Access Storage Device	GUI	Graphical User Interface
DBCS	double byte character set	HABJ	Host Access Beans for Java
DCAR	Digital Certificate Access Requestor	HACL	Host Access Class Library
		HACLJ	Host Access Class Library for Java
		HFS	Hierarchical File System
		HLLAPI	high level language application programming interface

HPR	high performance routing	PDF	printer definition file
HPT	host print transform	PDT	printer definition table
HTML	Hypertext Markup Language	PKCS	Public Key Cryptographic Standard
HTTP	Hypertext Transfer Protocol	PKI	public key infrastructure
IBM	International Business Machines Corporation	PPDS	personal printer definition stream
IDEA	International Data Encryption Algorithm	PPP	Point-to-Point Protocol
IETF	Internet Engineering Task Force	PSP	Preventive Service Planning
IPSec	IP security	PTF	Program Temporary Fix
ITSO	International Technical Support Organization	PWS	programmable workstation
JAR	Java archive	RACF	Resource Access Control Facility
JCL	job control language	RMI	remote method invocation
JDBC	Java database connectivity	SCCI	Screen Customizer Component Interface
JDK	Java developer kit	SCS	SNA character string
JRE	Java runtime environment	SDLC	synchronous data link control
JSP	JavaServer Pages	SET	secure electronic transactions
JVM	Java virtual machine	SLP	service location protocol
L2TP	Layer 2 Tunnel Protocol	SMP/E	System Modification Program/Extended
LDAP	Lightweight Directory Access Protocol	SMS	Systems Management Server
LLC2	link level control 2	SNA	Systems Network Architecture
LU	logical unit	SSL	Secure Socket Layer
MAC	message authentication code	TSO	Time Sharing Option
MPA	multi-protocol adapter	UDC	user-defined characters
MVS	Multiple Virtual Storage	UID	User Identification Number
NAT	Network Address Translation	URL	Uniform Resource Locator
NDIS	network driver interface specification	USB	universal serial bus
NLS	National Language Support	USSMSG10	Unformatted Systems Services, message number 10
NPT	non-programmable terminal	VM	virtual machine
ODBC	Open Database Connectivity	VPN	virtual private network
OHIO	Open Host Interface Objects	VT	virtual terminal
OIA	operator information area		
PCL	printer control language		

VTAM	Virtual Telecommunications Access Method
XML	Extensible Markup Language

Index

Symbols

!pxs 837
 &COMPn 160
 &USERn 160
)PSS.WD(460, 896
)USR.ID(459, 896
 .acg 871
 .adu 872
 .bar 872
 .bch 835–838, 852–853, 872
 .cert 872
 .der 872
 .ini 872
 .kbd 872
 .kbt 968
 .kmp 872
 .mac 872, 899
 .mlg 871, 912–913
 .mmp 872
 .msi 842
 .msp 842–844
 .ndc 872
 .pmp 872
 .scm 939, 979
 .srl 872
 .sth 872
 .tfr 872
 .tlg 871
 .tpl 939, 941, 973, 979
 .trc 871, 915
 .tto 872
 .upr 872
 .vbs 842
 .ws 835–836, 853–854, 872, 875, 902
 .xld 872
 .xlt 872
 .xml 899
 <\$stoprange 953
 _BPX_SETIBMOPT_TRANSPORT 94
 _BPXK_SETIBMOPT_TRANSPORT 142

Numerics

3270 associated printer 682–686

3270/5250 security
 configuration 304–308
 5250 associated printer 686–690
 5250 file transfer 423
 5250 Workstation ID 160
 5648-D76 934
 5722-AC2 423
 5722-AC3 423
 5722-CE2 424
 5722-CE3 424
 5722-SS1 423
 5769-AC1 (40-bit) 423
 5769-AC2 (56-bit) 423
 5769-AC3 (128-bit) 423
 5769-CE1 (40-bit) 424
 5769-CE2 (56-bit) 424
 5769-CE3 (128-bit) 424
 5769-SS1 423

A

Advanced Configuration and Power Interface 859
 AES 992
 AFP 687
 AH 1012
 Answer Back Message 321
 APAR 838
 Application Data Location 835, 867
 All Users Common Application Data Folder 872
 Users Application Data Folder 868
 application ID 457
 application-level gateway 1007
 AS/400 subfiles 929
 AS/400 table support 929
 AS/400 Toolbox for Java 254
 associated printer session 290
 asymmetric encryption algorithms 993–994
 Diffie-Hellman 994
 elliptic curve 993–994
 RSA 993
 AWT 753

B

Backup

private directory 82
 basic authentication 440, 448, 453
 BestFit 864

C

CA 113, 116, 119, 128–129, 134, 1000–1004
 See also Certificate Authority
 public 127
 root certificate 122
 trusted 111, 113, 116
 unknown 106
 Capture view ??–853, 866
 CBPDO 85
 CDMF 991
 CECI 176
 certificate 1022
 Common Name 442–443
 self-signed 454
 Certificate Authority 429
 See also CA
 certificates 127
 unknown 112, 129
 certificate management
 Host On-Demand 487–511
 using RACF 126
 Certificate Management Utility
 Host On-Demand 11, 492–505
 Personal Communications 893
 Certificate Wizard 119, 506–509
 CFGHODSVM 148
 cfgsrvlt.jar 398, 405
 Check for CSDs 839, 844–846
 CICS 3–4, 176
 CICS Gateway 14, 16, 176
 CICS Gateway client 4, 14, 176, 928
 Citrix Metaframe 528
 Classic Private Directory 870
 CLASSPATH 86
 Client Access 444, 971
 client authentication 443, 449
 ELF 130, 455–456, 464
 enabling
 OS/390 124
 Host On-Demand 440
 MSIE browser keyring 448
 OS/390 79, 106, 125
 IP services 125
 levels 133

Redirector 453
 using gskkyman 109
 client certificate 430, 456
 CLIENTAUTH 130, 134
 ColorRemap 761
 ColorRemapEvent 761
 ColorRemapListeners 761
 combined model
 Deployment Wizard
 configuration 556
 Directory Utility 367
 files stored on local machine 563
 planning 276
 precedence for changes 559
 user preferences 557
 Committing an APAR 839, 841
 Common Application Data Folder 869
 Communications Server for AIX 445, 456, 666, 1028
 Communications Server for OS/2 666
 Communications Server for OS/2 Warp 456
 Communications Server for OS/390 452
 Communications Server for Windows NT and Windows 2000 456, 666, 1028
 componentization 168, 172
 config.properties
 combined model 544
 Configuration Server-based model 553
 Configuration Servlet 411
 Novell NetWare 45
 OS/390 101
 migration 91
 UNIX 50
 config.properties.ascii 91, 99, 101–102, 104–105
 configuration server-based model 553
 ConfigServer 408, 414
 OS/390 105
 ConfigServerPort 57–58, 91, 100, 553
 ConfigServerURL 101–102, 105, 411
 ConfigServletURL 553
 Configuration Server 9, 14, 557
 changing the port 56
 OS/390 103
 Configuration Servlet 397, 408–409, 411, 455
 Deployment Wizard
 combined model 556–557, 559
 Directory Utility 367
 firewall 466
 installation 47–48

- OS/390 58
- integrated Windows domain logon 476, 479, 481–482
- Native Authentication 468–470
- port 467
 - changing 57
 - default 47
- security 427
- Configuration Server-based model 549–555
 - Deployment Wizard 549
 - Directory Utility 367
 - LDAP directory 381
 - planning 276
- Configuration Servlet 7, 9–10, 397–418
 - AS/400 155
 - classpath 404
 - config.properties.ascii 101
 - Deployment Wizard 554
 - Configuration Server-based model 553
 - HTTPS 412–413, 415
 - implementation scenarios 417
 - load balancing 417
 - installation 48
 - iSeries 155
 - OS/390 79, 105
 - WebSpher Application Server 81
 - parameters 408
 - planning 58
 - problem determination 418
 - referencing 412
 - security 455
 - selecting the application 405
 - setup 99–105
 - WebSphere Application Server
 - adding the servlet 406
- Convert macro to XML 853
- CRTJVAPGM 159
- Cryptographic Access Provider 423
- CSD and APAR tool 838
- CSDs 77, 838
- cstrace 847, 914–919
- CUSACCT 937
- CUSAPPLY 937
- CUSDDDEF 937
- CUSHFS 937
- CUSISMKD 937
- CUSRECVE 937
- Custom Terminal Bean 984–985
 - documentation 986
- customadmin 945
- Customization Studio 929–931, 953–973
 - add an image to a screen 961
 - add buttons to a screen 962
 - change screen color 958
 - converting text field to choice box 958
 - Create a label 964
 - create a Web link button 963
 - fine tuning 965
 - global customization 969
 - hiding fields 954
 - light pen support 969
 - macro button functions 967
 - close window 967
 - file transfer 968
 - new session 968
 - run applet 968
 - run macro file 968
 - modifying fields 954
 - modifying host function keys 959
 - moving fields 957
 - saving a screen as a map 953
 - simplified screen capture 969
 - template development 973
 - global template 974
 - Web link improvements 969
 - working with fields 954
- CustomizedCAs.class 106, 119–121, 124, 433
 - Certificate Management Utility 488, 492
 - make available to clients 490–491
 - unknown CA 490
 - certificate search 441
 - finding certificates 448
 - gskkyman 106
 - installation location 431
 - Java class files 430–431, 488
 - Keyring utility 509–511
 - locally installed client 431, 506
 - make available to clients 505
 - Microsoft cryptographic database 432, 491
 - OS/390 119
 - Redirector 454
 - server authentication 443
 - SSL support 430
 - SSLLight key database 505

D

data encryption

- OS/390 105
- data integrity 995
- Data Key 893
- Database On-Demand 4, 14, 247
 - common access problems 269
 - cannot establish the connection 270
 - no suitable driver 270
 - security exception 271
 - OS/400 Proxy 419, 423, 455
 - ports used 466
 - registering other JDBC drivers 266
 - SSL support 446
- DB2 267, 269, 271
- DBCS 18, 691
- DCAR 133–135, 463, 465
- DCAS 130, 132–136, 461, 463–465
- DDE 906–907
- default servlet engine 415
- deployment models 276
- Deployment Wizard 7–8, 284, 476, 529–566
 - cached client
 - Internet support 170
 - configuration parameters 481
 - Configuration Server 102
 - customizing toolbar 179
 - default model 516
 - deployment options
 - combined model 518
 - deployment strategy
 - HTML-based model 515
 - emulator client support 171
 - integrated Windows domain logon 476, 480–481
 - introduction 10
 - keyboard
 - custom functions 185–186
 - OS/390 considerations 79, 96
 - problem determination clients 171
 - smart caching 174
 - storing preferences
 - locally 452
 - transfer files to OS/390 98
- DES 81, 107–108, 991–992, 994, 997, 1041
- DFT 911
- Diffie-Hellman 994
- digital certificate 428
 - See also* X.509 certificate
 - basic concepts 990
 - description and purposes 1001

- digital signatures 1000
- ELF 455
- identity protection 1014
- security 1002
- simplified layout 1002
- SSL 1019
- Digital Certificate Access Requestor
 - See* DCAR
- Digital Certificate Access Server
 - See* DCAS
- digital signature 994, 998–1003, 1013, 1020
- Directory Service 273, 357–360
- Directory Utility 275, 367–368, 379
 - example 374
 - running 368
 - XML file syntax 369
- dirInfo.active 394
- Disable Functions 345, 537
- DisableSubfiles 156
- DWunzip 6

E

- ECLAppletInterface 752
- ECLConnection 747
- ECLConstants 747
- ECLCustomRecoEvent 751, 753
- ECLCustomRecoListener 751, 753
- ECLErr 751
- ECLField 747
- ECLFieldList 747
- ECLIOA 747
- ECLIOANotify 747
- ECLPS 747, 752
- ECLPSEvent 747, 753–754
- ECLPSListener 747
- ECLRecoDebugEvent 751
- ECLRecoDebugListener 751
- ECLRecoNotify 751
- ECLScreenDesc 751
- ECLScreenReco 751, 753
- ECLSDAttrib 751
- ECLSDBlock 751
- ECLSDCursor 751
- ECLSDCustom 751, 753
- ECLSDFields 751
- ECLSDInputFields 751
- ECLSDIOA 751
- ECLSDScreenDescriptor 751

ECLSDString 751
 ECLSession 746, 752, 754
 ECLTraceEvent 752
 ECLTraceListener 752
 ECLTraceProducer 752
 EHLLAPI 11, 297, 897, 906
 Personal Communications 904–908
 EHLLAPI Bridge 739, 780, 784
 ELF 79, 130, 135, 137, 455–457, 901
 See also Express Logon Feature
 macro recording 894
 three-tier network design 132
 TN3270 server 459
 two-tier network design 130
 elliptic curve
 performance 994
 encryption algorithms
 asymmetric 993
 Diffie-Hellman 994
 Elliptic Curve 994
 RSA 993
 symmetric 990
 AES 992
 CDMF 991
 IDEA 992
 RC2 991
 RC4 992
 ENDHODSVM 148, 156, 934
 ENDTCPSVR 56
 Enhanced Non-Programmable Terminal User Inter-
 face
 See ENPTUI
 ENPTUI 8, 297
 EPI 176
 ESP 1012
 Express Logon Feature 7
 See also ELF
 Personal Communications 863, 894–896

F

firewall 466, 1016–1017
 application level gateway 1009
 categories 1006–1009
 application-level gateway 1007
 concepts 989, 1004–1010
 application-level gateway 1007–1008
 configuration port 516
 Configuration Servlet 417, 455

DMZ 1009
 general guidelines 1005–1006
 hardening 1009–1010
 Host On-Demand 466
 logs 1006
 rules
 OS/400 Proxy 426
 SSL considerations 1023
 technologies 1006, 1008
 FMID 81, 85, 107
 HCUT206 935
 Host On-Demand V5
 HHOF500 80
 Host On-Demand V6
 HHOH600 80, 85, 107
 OS/390 encryption
 HIP6120 108
 HTCP380 108
 HTCP50A 108
 HTCP52A 108
 HTCP53A 108
 JIP612K 108
 JTCP382 108
 JTCP383 108
 JTCP38K 108
 JTCP5KA 108
 Fortezza Module 893
 FTP 2
 FTP client 4, 183
 FTP proxy server 1008

G

GemPlus/GemSoft Smartcard 893
 Get-to-the-Point 976
 global templates 941
 Global Variable extensions 976
 gskkyman 79, 106, 114
 certificate management 492
 create key database 109
 create self-signed certificate 117
 SSL configuration 109
 store a CA certificate 113

H

HABJ 739, 744, 747, 752, 755, 769–771, 778,
 786–787
 HACL 11, 297, 897, 908
 HACLJ 739, 744–747, 752–754, 769–771, 778,

- 786–787
- HACLJ functions 747
 - API Error Event 751
 - File Transfer 748
 - Important Global Constants 747
 - operator information area (OIA) 747
 - Outboard Function Execution 752
 - Presentation Space 747
 - Screen Recognition 751
 - Session 747
 - ECLCommNotify 747
 - ECLSession 747
 - Trace Facility 752
 - ECLTrace 752
- hibernation 859
- HLLAPI 904
- HMAC 997–998
- hod60mvs.sh 80, 82, 85, 140
- HODRAPD 141–142, 144–145
 - OS/390 140, 142
- HODSEDLINK-UNIX 938
- HODServerKeyDb.kdb 453–454, 488–489, 492–493, 497–498, 500, 503, 507
- HODServerKeyDb.sth 453, 488, 493, 500
- HODSRV 92
- HODUserMustExist 482
- HOMHFS 84
- HOMISMKD 85
- HOMSERVER 85–87, 89
- HOMSRV 89
- Host Access Beans for Java 3–4, 8, 12, 739, 742, 745, 754–756, 766, 787–788
 - ColorRemap 757, 760
 - FileTransfer 756, 759
 - Keypad 756, 758
 - KeyRemap 756, 758
 - Macro 756, 759, 766, 768–769
 - MacroManager 756, 768
 - Screen 756–758, 760
 - Session 756–758, 760
 - Terminal 756, 760, 787
- Host Access Class Library 4, 8
 - See also* HACL 11
- Host Access Class Library for Java 739, 745, 752
- Host File Transfer 291
- Host On-Demand
 - administration 273–379
 - Directory Service 357
 - Disabling License-Use Count 365
 - Filter option 284
 - OS/400 Proxy Server 361
 - administration clients 166
 - cached clients 167–171
 - across the Internet 169
 - certificate management 487–511
 - Certificate Wizard 506–509
 - client authentication 449
 - CSDs 77
 - Database On-Demand 446
 - SSL 446
 - default clients 164
 - deployment strategies 513
 - download clients 167
 - firewall
 - ports 467
 - FTP client 444–445
 - function on-demand client 172
 - host printing 661–695
 - installation 32
 - AIX 47
 - AIX graphical interface 47
 - AIX silent mode 48
 - local client 62
 - Novell NetWare 45
 - OS/2 43
 - OS/390 or z/OS 56
 - OS/400 52
 - UNIX 50
 - Windows 32
 - Windows InstallShield 32
 - Windows silent mode 38
 - integrated Windows domain logon 475–483
 - Java class files
 - CustomizedCAs.class 430
 - WellKnownTrustedCAs.class 430
 - migration considerations 66
 - client 73
 - server 67
 - OS/390
 - FMID 80
 - implementation 79
 - OS/400 Proxy server 455
 - ports 467
 - problem determination 801
 - OS/390 104
 - problem determination clients 171
 - PTFs 77
 - Redirector

See also Telnet Redirector
 removing 75
 secure Telnet
 defining 447
 security 427–486, 522–524, 526
 server authentication 448
 Service Manager 48
 service updates 77
 SSL
 implementations 440
 server authentication 441
 support 430
 Telnet-negotiated session 448
 TN3270 client 444
 TN5250 client 444
 utility clients 174
 new user client 175
 remove cached client 175
 VT client 445
 Host On-Demand Toolkit 984
 Host Print Transform 687–688
 HTML-based model 276, 515, 534, 557, 559, 563
 HTTP caching 158
 HTTP(S) 455
 httpd.conf 37, 99–100, 104
 httpd.envvars 99, 101, 104

I
 IBM Domino Go Webserver 48, 398
 IBM Key Management Utility 453–454
 IBM Netfinity PSG Chip 893
 IBM SecureWay Smartcard 893
 IBMPCOMM Plus Module 847
 IKE 1012–1015
 IME. *See* Input Method Editor
 IND\$FILE 292, 444
 Information Bundler 914
 Input Method Editor 5
 Integrated Windows domain logon 475, 551
 parameters 481
 Internet Key Exchange
 See IKE
 IPMonitor 983
 IPSec 989, 993, 1001, 1010, 1012, 1015–1016,
 1018
 ISAKMP 1012–1013

J
 J2EE Connector 739, 746, 795–798
 Java 2 744, 769–780, 795
 Java 2 Plug-In 9, 221–246
 Java Database Connectivity
 See JDBC
 Java file interface mode 662, 667
 Java keyring utility 119–120
 JAVA_HOME 86
 JDBC 2, 247, 254, 266
 DB2 driver 267
 JDBC driver 247

K
 keyring utility 509–511
 keyrng.class 106

L
 L2TP 1017–1018
 Layer 2 Tunnel Protocol
 See L2TP
 LDAP 137, 139
 See also LDAP directory
 LDAP client 1019
 LDAP directory 277, 358, 381–395
 Directory Utility 367, 372–373
 enablement 359
 Host On-Demand
 enable 358
 load balancing 417
 manage groups 276–277
 manage users 277–279
 manage users and groups 274
 Native Authentication 281
 OS/390 140
 Native Authentication 143
 schema installation 384
 OS/390 138
 Screen Customizer
 admin ID 945
 LDAP directory server 80, 138, 1019
 load balancing 417
 OS/390 79, 81, 137–138
 LIBPATH 86
 License Use Management 9, 14
 light pen 928
 Screen Customizer 928
 Load balancing 1028

locally stored preferences 171
 LOG0001 393
 LOG0002 104
 Lotus Domino Go Webserver 48, 397–398, 417
 LU Type 1 666
 LU Type 3 667

M

MAC 997–998
 Macro 752, 754–755, 766
 MacroIOPProvider 760
 MacroIOPProvider example 1031
 MacroManager 752
 MacroManager Bean 760
 MD2 996
 MD5 466, 996, 998
 MergeDB 981
 message authentication codes 997
 HMAC 997
 message digest 995–999, 1002
 message digest algorithms 995–996
 MD2 996
 MD5 996
 SHA-1 996
 SHA-256 996
 SHA-512 996
 Microsoft cryptographic database 119, 307, 430–441, 490–491
 adding a personal certificate 434
 basic SSL 441
 certificate name 449
 certificate source 449
 Host On-Demand
 session configuration 448
 mask 450–451
 prompting 452
 recommendation 440
 server authentication 442
 viewing certificates 432
 Microsoft SNA Server 666
 Microsoft Windows Terminal Services 528
 mkkf 106
 MSIE browser keyring
 See Microsoft cryptographic database
 msixexec 844
 multicast 1027
 multiple session object 343–344

N

NAT 1008
 National Language Support 16
 Native Authentication 7, 9, 79, 142, 280–283, 373, 375, 378, 381, 417–418, 523, 525–527
 OS/390 140–145
 problem determination
 OS/390 145
 Netscape IPlanet 10
 Network Address Translation 420
 new user client 175
 Novell JDK 45
 NSMprop 45, 50, 57, 83, 90–91, 283
 NSTRC.DAT 849, 919
 NVT 911

O

OHIO 7
 OIA 177–178, 183–184, 894, 910, 912
 OLE 897
 operator information area
 See OIA
 OS/390 417, 457
 multiple TCP/IP stacks 94
 OS/390 sample jobs 84
 HOMALLOC 84
 HOMCOPY 84
 HOMDDCLN 84
 HOMDDDEF 84
 HOMHFS 84
 HOMISMKD 84
 HOMRECVE 84
 HOMSERVR 84
 OS/400 147
 Apache HTTP server 149
 Configuration Servlet 155
 Domino HTTP Server 152
 JVM
 SF63319 148
 SF63322 148
 SF99067 148
 SF99068 148
 SF99069 148
 JVM level 148
 network drive 162
 performance tips 158
 printing 161
 subfiles 156

Telnet dropout 161
 OS/400 file transfer 455
 OS/400 HTTP server
 configuration 55
 OS/400 Proxy 9, 94–95, 254, 361–362, 445–446,
 455, 466, 516
 port 94, 467
 OW45575 81
 OW45791 81

P

P12 file 893
 packet filtering 1006–1007
 categories
 packet filtering 1007
 router 1007–1009
 params.txt 481
 PassTicket 460
 PATH 86
 PCL 687
 pcommaparinfo.txt 841
 PCommClientKeyDb.kdb 890
 pcseclj.jar 745
 pcsmig.log 873
 PCSWS046 860
 PCWMSG.MLG 912–913
 PDF 663, 691
 PDF. *See* Portable Document Format
 PDT 157, 662, 691
 personal certificate 434–435
 Personal Communications
 EHLLAPI 904–908
 Express Logon Feature 894–896
 HACL 908
 macros 898
 converting to XML 898–904
 migration
 during installation 868
 Migration Utility 873
 Service Bundler 913
 Session Manager 834, 852
 smart card 890–893
 smart card drivers 893
 Tivoli ??–851
 tracing 913–917
 Personal Communications
 macros 904
 PKCS

See asymmetric encryption algorithms
 PKCS#11 890–891, 893
 PKCS#11 driver name 892
 PKCS12 file 430
 PKI 1000–1001
 Portable Document Format 5
 Postscript printer 662
 PPDS 687
 printer definition files
 See PDF
 printer definition table
 See PDT
 private key 453, 993–994, 997–1000, 1002–1003
 Product Update Tool ??–846
 PSDebugger 746
 PTF 80–81, 85, 936
 PTFs 77
 PTKTDATA 130–131, 135, 461, 894
 public certificate 430
 public key 453, 993–994, 997, 999–1003, 1020
 public key encryption 994, 1000, 1013
 Public Key Infrastructure 428
 See PKI
 Public-Key Cryptography Standards
 See asymmetric encryption algorithms

Q

QEJBSBS 155
 QSECOFR 52
 QSYSWRK 54, 56

R

RAC
 level 3 106
 RACDCERT 128–129, 134
 RACF 79, 135
 application 457
 authentication 524
 CA certificate 129
 certificate management 126, 492
 Communications Server for OS/390
 keyring repository 106
 SSL 105
 deployment strategy 524
 digital certificate 455
 ELF 455, 462, 465, 894
 DCAS 463–464
 Host On-Demand

- installation 86–87
- key database 109
- keyring support 107
- level 2 106, 134
- level 3 134
- Native Authentication 143, 418, 526
- Passticket profile 130
- profile 134
- profile names 129
- SAFCERT 106
- SAFKEYRING 136
- Secured Signon services 461, 463–464
- security management 522
- Security Server 85
- SERVAUTH 135
- system 461
- user ID 455
- Rainbow lkey 1000 893
- RC2 991–992
- RC4 992
- RDBM 137
- Redbooks Web site 1044
 - Contact us xxii
- Redirector 9, 11, 523
 - certificates 453
 - client authentication 443
 - client configuration 288–289
 - Communications Server for AIX 14
 - configuration 351
 - Host On-Demand 14
 - key database file 453, 488
 - Keyring utility 509
 - OS/400 Proxy 419, 422
 - security 452
 - SSL 452
 - both 453
 - client-side 452
 - host-side 452
 - pass-through 453
 - SSL connections 105
 - SSL support 428
 - traffic volumes 524
 - using certificates 488
 - VT client 445
 - VT sessions 14, 523
- Remove an APAR 839–840
- RFC 1319 996
- RFC 1321 996
- RFC 1572 462, 465

- RFC 2104 997
- RFC 2246 483, 1019, 1023
- RFC 2253 1002
- RFC 2412 1013, 1041
- root certificate 453
- RSA encryption 993–994, 998, 1018, 1020
- RSTLICPGM 148, 934
- RUN command 837

S

- sc_global.tpl 974
- SCCI 929, 984–985
 - documentation 986
- Schlumberger Cryptoflex 893
- Screen 754
- Screen Customizer 156, 171
 - Administrator 924, 929, 942
 - capturing a screen 945
 - global customization 943
 - screen capture 943
 - built-in screen ID 945
 - Components 929–930
 - Customization Studio 924
 - See Customization Studio
 - deployment 978
 - development 942–981
 - developing a Template 975
 - template hierarchy 975
 - installation 931–941
 - Administrator and Customization Studio 931
 - AIX runtime 933
 - iSeries runtime 934
 - Novell NetWare runtime 934
 - OS/2 runtime 935
 - runtime 933
 - UNIX runtime 933
 - Windows runtime 933
 - migration 941
 - Planning 930
 - Runtime Client 930
 - screen structure 945
 - Service Bundler 929, 981
 - tags 945
- Screen Customizer Administrator 929, 931
- Screen Customizer Beans 929
- Screen Customizer Component Interface 984
 - See also SCCI

- Screen Customizer Runtime
 - OS/390 935–940
 - deployment considerations 939
 - installation 937
 - Java 1.3 consideration 938
 - planning 935
 - post SMP/E processing 938
 - silent installation 940
 - SMP/E Processing 937
- Screen Customizer/LE 12
- screen ID 930, 942, 945, 951, 954
- screen.db 942, 981
- ScreenMouseEvents 761
- SCW PKCS 3GI 3-G International 893
- Secure Sockets Layer 877
 - See also* SSL
 - certificate 1022
 - generate encryption key 1022
 - introduction 989
- self-signed certificate 497
 - authenticating DCAR 134
 - authenticating DCAS 133
 - Certificate Management 488
 - Certificate Management Utility 492
 - Certificate Wizard 506–507
 - creating 500–503
 - using gskkyman 117–118
 - using RACF 127, 129
 - Java keyring utility 122, 511
 - make available to clients 119, 490, 503, 505
 - OS/390 111
 - Redirector 453
 - server authentication 443
 - using 489–490
 - using Microsoft cryptographic database 491
- SERVAUTH 134–135
- server authentication
 - client configuration 307, 445
 - defining secure Telnet session 448
 - operations 442
 - OS/390 113, 119, 122–124
- Service Bundler
 - Screen Customizer 929
- service key 844
- Service Location Protocol 1027–1030
 - See also* SLP
- Service Manager
 - error starting
 - OS/390 95
- Host On-Demand 86
 - OS/390 91
 - start 85
 - starting 89
 - installation
 - OS/390 83
 - LDAP
 - initialization 389
 - OS/390 394
 - revert to private data store 394
 - startup 393
 - switching 392
 - OS/400
 - status 54
 - OS/400 Proxy
 - OS/390 95
 - restart
 - OS/390 93
 - starting
 - OS/390 79
 - stopping
 - OS/390 92
- ServiceManager.sh 92–93, 394
- Session 755
- Session Manager
 - Personal Communications ??–838
- SHA 466
- SHA-1 996, 998
- SHA-256 996
- SHA-512 996
- ShowStats 100, 102–105, 408
- signed applet 429
- Sleep Permission 859
- SLP 289, 298
- smart caching 172, 174
- smart card 432, 449
 - digital certificate 443
 - Host On-Demand 8
 - Microsoft cryptographic database 432
 - Personal Communications 864, 890–893
 - SMP/E 80–81, 83–84, 95, 140, 935–936
 - SOCKS server 1008
 - SQL 247–249, 251–255, 262–266
 - SQL Assist Exception 269
 - SSL 108, 998, 1001, 1012, 1018–1020, 1022–1025
 - 3270 client 444
 - 5250 client 444
 - basic authentication 441
 - certificate management

- self-signed certificate 490
- using certificates 488
- Certificate Management Utility 488
- CICS Gateway client 176
- Communications Server for OS/390 105–106
- concepts 989
- Database On-Demand 254
- ELF 130, 455
 - client setup 894
 - DCAR 133
 - DCAS 132–133, 464
 - DCAS customization 135
 - session setup 456
- enabling
 - Telnet 448
- FTP 444–445
- Handshake Protocol 1020
- Host On-Demand 430
 - implementation 440
 - session security 4
- Java keyring utility 120
- Keyring utility 510
- OS/390 79, 106
 - configuring server 125
 - overview 107
 - server authentication 122
- OS/400 Proxy 423–424
- pass-through 453
- PKI 1001
- protocol
 - general information 109
- Record Protocol 1020, 1022
- Redirector 428, 452
 - both 453
 - certificates 453
 - client-side 452
 - host-side 452
- RSA 1020
- self-signed certificate 454, 1023
- server authentication
 - client configuration 123
- server authentication 440
- SSLCERT 106
- Telnet 428
- Telnet-negotiated session 448
 - session configuration 483
- TLS 998
- V3CIPHER 136
- VT 445

- Web server 427
- SSL handshake 106
- SSL indicator
 - Screen Customizer 928
- standby 859
- status bar 912
- STRHODSVM 148, 156
- Structured Query Language
 - See* SQL
- sub files 157
- Subclassing 787
- subfiles 948, 971–972
- symmetric encryption algorithms 990–992
 - AES 992
 - CDMF 991
 - DES 991–992, 994, 997, 1041
 - IDEA 992
 - RC2 991
 - RC4 992
- symmetric encryption key 1020
- SYS1.LINKLIB 140
- SYS1.PARMLIB 95

T

- TDBM 137, 383
- Telnet 107
 - enabling SSL 448
- Telnet printer association 854–858
- Telnet proxy
 - See* Redirector
- Telnet-negotiated security 448, 523
 - IETF Internet draft 448
 - session negotiation 453
- Telnet-negotiated session 107, 306
- Terminal 754
- Test an APAR 839
- TLS 483, 989, 998, 1019, 1023, 1025
- TLS. *See* Transport Layer Security
- TMR server 848
- TN3270
 - host file transfer 444
 - FTP 444
 - IND\$FILE 444
- TN5250
 - host file transfer 444, 466
 - FTP 444
- Transport Layer Security 306
 - See* TLS

introduction 989
 Triple-DES 81, 108, 126, 991–992
 trusted applet
 See also signed applet 429
 trusted CA 430, 443

U

UDC 692
 UID uniqueness filter 389
 Unicode 19
 UNIX crypt 466
 unknown CA 432, 453
 UseHostColors 156
 User-Defined Characters
 See UDC
 Users Application Data Folder 868
 UseWindowsDomain 481
 USSMSG10 458
 UW72476 81
 UW73147 81

V

vbhllapi.exe 904
 VBSCRIPT 898
 VeriSign 1001
 virtual private network
 See VPN
 VPN 989, 1005, 1008, 1010, 1015–1017
 VT 445
 VT client 928
 VT Host printing 693
 VT100 321
 VT220 321
 VT320 321
 VT420 321
 VT52 321
 VTAM 107, 458, 911
 VTAM APPLID 894–895

W

warm standby 1029
 was.conf 99–100, 104
 WDHODGroup 482
 Web application server 398
 Web server
 alias 37
 pass rules 88

timeout directives 95
 WebSphere Application Server 10, 81
 configuration
 alias 402–403
 configuration files
 OS/390 99–102
 Configuration Servlet 397
 installation 398
 configuring 401
 export definitions 415
 Host On-Demand
 installation 48
 load balancing 417
 OS/390 99
 well-known CA 429, 453–454
 WellKnownTrustedCAs.class 433, 488–491
 certificate search 441, 448
 installation location 431
 Java class files 430–431
 locally installed clients 431
 make available to clients 119
 Redirector 453
 server authentication 443
 trusted CAs 430
 Windows 2000 certified
 Personal Communications 834
 Windows 2000 Power Management 859
 Personal Communications 858
 Windows native printer interface mode 662, 664, 667, 694
 Windows Network Neighborhood 162
 Windows Scripting Host Service 842
 Windows spooler interface mode 662, 664, 667, 694
 Windows XP certified
 Personal Communications 15
 WindowsDomain 482
 workstation ID 290, 521
 WRKHTTPCFG 56, 158
 WRKJOB QHODSVM 148

X

X.509 certificate 428–429, 443, 449, 464, 864, 890
 ELF 461
 XML 898–900
 XMLConfig 401, 415

To determine the spine width of a book, you divide the paper PPI into the number of pages in the book. An example is a 250 page book using Plainfield opaque 50# smooth which has a PPI of 526. Divided 250 by 526 which equals a spine width of .4752". In this case, you would use the .5" spine. Now select the Spine width for the book and hide the others: Special>Conditional Text>Show/Hide>SpineSize(-->Hide)>Set

Draft Document for Review November 24, 2002 6:23 pm

6182spine.fm 1065



IBM Host Access Client Package Update

(1.5" spine)
1.5"<=> 1.998"
789 <=> 1051 pages

To determine the spine width of a book, you divide the paper PPI into the number of pages in the book. An example is a 250 page book using Plainfield opaque 50# smooth which has a PPI of 526. Divided 250 by 526 which equals a spine width of .4752". In this case, you would use the .5" spine. Now select the Spine width for the book and hide the others: Special>Conditional Text>Show/Hide>SpineSize(-->Hide.)>Set

Draft Document for Review November 24, 2002 6:23 pm



Draft Document for Review November 24, 2002 6:24 pm

IBM Host Access Client Package Update



Host On-Demand Portlet

Deployment Wizard enhancements

Enable Host On-Demand for Java 2

The three products in the IBM Host Access Client Package, IBM Personal Communications, IBM WebSphere Host On-Demand, and IBM Screen Customizer, have been enhanced with new features and functions to keep up with current technologies. This book explores the features and functions of each product as it relates to deployment in today's rapidly expanding TCP/IP environment.

The following is an overview of many of the topics found in this book:

- ▶ Improved Host On-Demand for z/OS installation, support, and operational guidelines
- ▶ Improved iSeries tips
- ▶ New portlet support in Host On-Demand Version 7 for WebSphere Portal Server
- ▶ Improved and simplified Host On-Demand support for user administration
- ▶ Personal Communications Windows XP certification
- ▶ Enhanced Personal Communications security
- ▶ Updated Screen Customizer for Host On-Demand Version 7

INTERNATIONAL TECHNICAL SUPPORT ORGANIZATION

BUILDING TECHNICAL INFORMATION BASED ON PRACTICAL EXPERIENCE

IBM Redbooks are developed by the IBM International Technical Support Organization. Experts from IBM, Customers and Partners from around the world create timely technical information based on realistic scenarios. Specific recommendations are provided to help you implement IT solutions more effectively in your environment.

For more information:
ibm.com/redbooks