

Operating System Bull DPS 7000

GCOS 7 System Operator's Guide

47 A2 53US Rev07

Operating System Bull DPS 7000

GCOS 7 System Operator's Guide

Subject: This manual gives the normal operating procedures for DPS 7000 computers running under GCOS 7 Operating System.

Special Instructions: Rev07 cancels and replaces Rev06 for users of GCOS 7 V10 (TS 9910).

Software Supported: GCOS 7 V9-TS 9866
GCOS 7 V10-TS 9870
GCOS 7 V10-TS 9910

Software/Hardware required:

Date: May 2003

Bull S.A.
CEDOC
Atelier de reprographie
357, Avenue Patton BP 20845
49008 ANGERS Cedex 01
FRANCE

Copyright © Bull S.A., 1997-2003

Bull acknowledges the rights of proprietors of trademarks mentioned herein.

Your suggestions and criticisms concerning the form, contents and presentation of this manual are invited. A form is provided at the end of this manual for this purpose.

No part of this publication may be reproduced, stored in a retrieval system or transmitted in any form or by any means, electronic, mechanical or otherwise without the prior written permission of the publisher.

Bull disclaims the implied warranties of merchantability and fitness for a particular purpose and makes no express warranties except as may be stated in its written agreement with and for its customer. In no event is Bull liable to anyone for any indirect, special, or consequential damages.

The information and specifications in this document are subject to change without notice.

Consult your Bull Marketing Representative for product or service availability.



Preface

Manual objectives	This manual gives the normal operating procedures for DPS 7000 computers running under the GCOS 7 operating system.	
Intended readers	Main console and station operators should read this manual.	
Structure of the manual	Section 1	is a general introduction for the GCOS 7 main operator.
	Section 2	is a practical guide to hardware initialization.
	Section 3	is a guide to software loading, starting and terminating a session.
	Section 4	deals with operations under GCOS 7.
	Section 5	deals with facilities for running user jobs.
	Section 6	deals with facilities for using peripheral devices.
	Section 7	is an alphabetical presentation of GCL commands most used by the operator.
	Section 8	is a continuation of the alphabetical presentation begun in Section 7.
	Section 9	describes the Remote Multiplexed Operator Support (RMOS) facility.
	Section 10	describes static and dynamic hardware reconfiguration.
	Appendix A	Operator-System Dialog on a Console
	Appendix B	Invoking System Utilities
	Appendix C	List of Commands by "Object"
	Appendix D	The SYS.URCINIT Printer Control File.
	Appendix E	WHEN, EVERY and RRESTART



Appendix F	describes the device types and classes
Appendix G	describes how GCOS 7 handles dates before and after the year 2000.
Appendix H	describes how to use the Output Writer.

Associated documents

<i>V7000 Operator's Guide</i>	47 A2 74UU
<i>Printers Operator's Guide</i>	47 A2 00UU
<i>GCOS 7 Console Messages Directory</i>	47 A2 61UU
<i>GCOS 7 Messages and Return Codes Directory</i>	47 A2 10UJ
<i>CPU and Operator Station Manual (DPS 7/2x0/3x0 Models)</i>	77 A1 51UU
<i>Operator Reference Manual (DPS 7000/2x0/3x0 Models)</i>	77 A1 52UU
<i>DPS 7000/An Operator's Guide</i>	77 A1 71UU
<i>DPS 7000/5x0/7xx Operator's Guide</i>	77 A1 81UU
<i>DPS 7000/MTxx Operator's Guide</i>	77 A1 65US
<i>GCOS 7-V8 System Installation, Configuration and Updating Guide</i>	47 A2 19US
<i>GCOS 7-V9 System Installation, Configuration and Updating Guide</i>	47 A2 23US
<i>TDS Administrator's Guide</i>	47 A2 32UT
<i>TDS COBOL Programmer's Manual</i>	47 A2 33UT
<i>TDS C Programmer's Manual</i>	47 A2 07UT
<i>High Availability Concepts</i>	47 A2 22UT
<i>High Availability Administrator's Guide</i>	47 A2 23UT
<i>Mirror Disks User's Guide</i>	47 A2 20UF
<i>GCL Programmer's Manual</i>	47 A2 36UJ
<i>IOF Terminal User's Reference Manual - Part 1</i>	47 A2 38UJ
<i>IOF Terminal User's Reference Manual - Part 2</i>	47 A2 39UJ
<i>IOF Terminal User's Reference Manual - Part 3</i>	47 A2 40UJ
<i>GCOS 7-V8/V9 System Administrator's Manual</i>	47 A2 54US
<i>Generalized Terminal Writer User's Guide</i>	47 A2 55UU
<i>Large Memory Cache User's Guide</i>	47 A2 42US
<i>GCOS 7 Network User's Guide</i>	47 A2 94UC
<i>Remote Facilities DPS 7 to DPS 6</i>	47 A2 11UC
<i>UFT User's Guide</i>	47 A2 13UC
<i>DJP User's Guide</i>	47 A2 14UC
<i>DNS-V4 Terminal Management Reference Manual</i>	39 A2 24DN
<i>CNS7-A1 Terminal Management Reference Manual</i>	39 A2 43DM
<i>DSA Documentation Directory</i>	39 A4 9726
<i>ARM User's Guide</i>	47 A2 11US



Administering the Storage Manager47 A2 36UF

Distributed Operator Facility - Programmed Operator User's Guide... 47 A2 80UC

Script Manager User's Guide 47 A2 84UC

File Recovery Facilities User's Guide47 A2 37UF

PRUG00X (PR701/801/901) User's Guide..... 47 A2 00UU

Electronic Documentation

The complete GCOS 7 document set is available on one CD disk. This product is known as CD-DOC. It is delivered for each system and with each software or add-on delivery.

This CD-ROM is updated each time a new GCOS 7 technical status is issued. Between 2 CD-ROM versions, manuals which have been updated are accessible, for reading or downloading, on a specific WEB site (link to this WEB site is available on the CD-DOC itself).

Warning

This manual describes all the GCOS 7 commands, some of them are only available if the hardware platform model supports the functionality. As an example:

The GCOS 7 command `Connect domain` is only available on DPS 7000/MTxx and DPS 7000/Taxxx series.
Mirror disks no more available on DPS 7000/XTA platforms

**Syntax
Notation**

The following conventions are used for presenting GCL (and JCL) command syntax.

ITEM An item in upper case is a literal value, to be specified as shown. The upper case is merely a convention; in practice you can specify the item in upper or lower case.

item An item in lower case is a non-literal, indicating that a user-supplied value is expected.

In most cases it gives the type and maximum length of the value:

- char105** a string of up to 105 alphanumeric characters
- name31** a name of up to 31 characters
- lib78** a library name of up to 78 characters
- file78** a file name of up to 78 characters
- star31** a star name of up to 31 characters

In some cases, it gives the format of the value:

- a** a single alphabetic character



nnn a 3-digit number
hh.mm a time in hours and minutes

In other cases, it is simply descriptive of the value:
 device-class
 condition

<u>ITEM</u>	An underlined item is a default value. It is the value assumed if none is specified.
bool	A boolean value which is either 1 or 0. A boolean parameter can be specified by its keyword alone, optionally prefixed by "N". Specifying the keyword alone always sets the value to 1. Prefixing the keyword with "N" always sets it to 0.
{ }	Braces indicate a choice of values. Only one can be selected.
[]	Square brackets indicate that the enclosed item is optional. An item not enclosed in square brackets is mandatory.
()	Parentheses indicate that a single value or a list of values can be specified. A list of values must be enclosed by parentheses, with each value separated by a comma or a space.
...	Ellipses indicate that the item concerned can be specified more than once.
+ = \$ * / - .	Literal characters to be specified as shown.

EXAMPLE 1:

```
[ { IMMED } ]
[ WHEN = { [dd.mm.yy.] . hh.mm } ]
[ { +nnnn{W|D|H|M} } ]
□
```



This means you can specify:

1. Nothing at all (WHEN=IMMED applies).
2. WHEN = IMMED (the same as nothing at all).
3. WHEN = 22.30 to specify a time (and today's date).
4. WHEN = 10.11.87.22.30 to specify a date and time.
5. WHEN = +0002W to specify 2 weeks from now.
6. WHEN = +0021D to specify 21 days from now.
7. WHEN = +005H to specify 5 hours from now.
8. WHEN = +0123M to specify 123 minutes from now.

EXAMPLE 2:

```
PAGES = (dec4 [-dec4] [,dec4] ...)
```



Indicates that PAGES must be specified. Valid entries are a single value, or a list of values enclosed in parentheses. The list can consist of single values separated by a comma, a range of values separated by a hyphen, or a combination of both. For example: PAGES = (2,4,10-25,33-36,78,83).

EXAMPLE 3:

```
[ REPLACE = { bool | 0 } ]
```



This is a boolean parameter whose default value is zero. You can specify:

1. Nothing at all (REPLACE = 0 applies)
2. REPLACE = 0 or simply NREPLACE
3. REPLACE = 1 or simply REPLACE





Table of Contents

1. Introduction to Running a System under GCOS 7

1.1	Overview	1-1
1.2	DPS 7000 Systems - Hardware Overview	1-2
1.2.1	DPS 7000/2x0 and /3x0 Systems	1-2
1.2.2	DPS 7000/A Systems	1-3
1.2.3	DPS 7000/5x0/7xx/8x0/Dx0/Mx0 Systems	1-4
1.2.3.1	DPS 7000/5x0/7xx Systems	1-4
1.2.3.2	DPS 7000/8xx/Dx0/Mx0 Systems	1-5
1.2.4	The DPS 7000/4xx/Cxx Systems	1-6
1.2.4.1	The DPS 7000/4xx/Cxx Systems	1-6
1.2.4.2	The Subsystems (CSS, IOSS, etc)	1-7
1.2.5	DPS 7000/MTxx/TAxXX/TAxXXC Systems	1-9
1.2.5.1	DPS 7000/MTx1 Non-redundant Models	1-9
1.2.5.2	DPS 7000/MTxx/TAxXX Redundant Models and TAxXXC models... ..	1-10
1.2.6	DPS 7000/XTA systems	1-12
1.2.6.1	DPS 7000/XTA cabinet description	1-12
1.2.6.2	DPS 7000/XTA specific Initialization operations	1-17
1.2.6.3	DPS 7000/XTA specific administrator & operators principles	1-18
1.3	The GCOS 7 Operating System.....	1-20
1.4	A GCOS 7 Session.....	1-21
1.4.1	Initial System Load (ISL).....	1-21
1.4.2	Job Management	1-23
1.5	The Role of the Operator.....	1-24
1.5.1	Main Operator	1-24
1.5.2	Other Operators	1-24
1.5.3	GCOS 7 Control of Certain Operator Tasks	1-25
1.5.4	Unattended Operation.....	1-26
1.5.5	Operating a Large System	1-27
1.5.6	Distributed Systems on a Network.....	1-27



1.6	The Dimensions of GCOS 7.....	1-28
1.6.1	The Batch Dimension.....	1-28
1.6.2	The Interactive and Remote Facility Dimensions	1-28
1.6.3	The TDS Dimension.....	1-29

2. Starting Up Your System

2.1	Introduction	2-1
2.2	The DPS 7000/A	2-2
2.2.1	Using the DPS 7000/A Console.....	2-2
2.2.2	Introducing the DPS 7000/A Console	2-4
2.2.3	Powering On the DPS 7000/A	2-5
2.2.4	Setting the DPS 7000/A to Automatic Mode.....	2-6
2.2.5	Power On or System Reset Command from the Console.....	2-6
2.2.6	Service Commands: Reload (RL) and Restart (SR) on DPS 7000/A.....	2-7
2.2.6.1	Reload Procedure.....	2-7
2.2.6.2	To Restart the System on a DPS 7000/A.....	2-7
2.2.7	Powering Off the DPS 7000/A	2-8
2.2.8	Setting Power On/Off Timers (MDPWR and DPWR Commands).....	2-8
2.3	The DPS 7000/2X0 and /3X0	2-9
2.3.1	Powering On the DPS 7000/2x0 or /3x0.....	2-9
2.3.2	Manual Initialization on the DPS 7000/2X0 and /3X0.....	2-13
2.3.3	ASPA Configuration Parameters on the DPS 7000/2X0 and 3X0.....	2-15
2.3.4	Service Commands: Reload (RL) and Restart (SR) DPS 7000/2X0 and /3X0 Systems.....	2-16
2.3.4.1	Reload procedure (RL).....	2-16
2.3.4.2	System Restart Procedure (SR).....	2-17
2.4	The DPS 7000/4xx/Cxx, DPS 7000/5x0/7xx/8x0/Dx0/Mx0 and DPS 7000/MTxx/Taxxx/TAxxC.....	2-18
2.4.1	Turning On and Off The System.....	2-18
2.4.2	Using the Consoles.....	2-20
2.4.3	Service Commands: Reload (RL), Restart (SR).....	2-22
2.4.4	Setting the Power On/Off Timers (MDPWR & DPWR Commands)	2-22
2.4.5	Powering Off the System	2-23
2.4.6	The DPS 7000/4xx/Cxx/MTx1.....	2-24
2.4.7	The DPS 7000/5x0/7xx.....	2-25
2.4.8	The DPS 7000/8x0/Dx0/Mx0/MT42 and UP/TAxxx/TAxxC	2-26



2.5	The DPS 7000/XTA.....	2-27
2.5.1	Powering On the DPS 7000/XTA.....	2-28
2.5.2	Using the consoles.....	2-30
2.5.3	Powering Off the system.....	2-37
2.5.4	Windows™ and GCOS 7 coexistence	2-38

3. Starting and Terminating a GCOS 7 Session

3.1	Introduction	3-1
3.2	Initial System Load (ISL).....	3-2
3.2.1	Purpose	3-2
3.2.2	Operations.....	3-2
3.3	Operating Modes	3-3
3.3.1	Operating Modes	3-3
3.3.2	Selecting a Console	3-3
3.3.3	Unattended Mode	3-4
3.3.4	Operator Commands Allowed During GCOS 7 Initialization	3-5
3.3.5	Console Failure	3-5
3.3.6	System Startup.....	3-6
3.3.7	System Ready Transition.....	3-7
3.3.8	Loading Operations: Unattended Mode.....	3-7
3.3.8.1	DISPLAY_RESTART_OPTIONS (DRSO) Command.....	3-8
3.3.8.2	MODIFY_RESTART_OPTIONS (MDRSO) Command.....	3-8
3.3.8.3	Automatic Reload	3-10
3.3.8.4	Automatic Restart	3-10
3.3.8.5	Cancelling Auto Mode	3-11
3.3.8.6	Dump Facilities in Automatic Mode	3-11
3.3.8.7	Abnormal Events During First Phase of GCOS 7 Loading (ISL).....	3-12
3.4	Loading Operations: Attended Mode	3-13
3.4.1	System Previous Status.....	3-14
3.4.2	IL01 Options.....	3-15
3.4.3	Answering the IL01 Question.....	3-16
3.4.3.1	BKST	3-18
3.4.3.2	DATE	3-20
3.4.3.3	DISK	3-21
3.4.3.4	DISPLAY.....	3-22
3.4.3.5	DUMP	3-23
3.4.3.6	DUMP_CD.....	3-27
3.4.3.7	HA (High Availability).....	3-27
3.4.3.8	HELP (or ?).....	3-28
3.4.3.9	INIT	3-28
3.4.3.10	LIB	3-29
3.4.3.11	MAM	3-31
3.4.3.12	NSYSSTUP	3-31
3.4.3.13	PAGESIZE.....	3-31



3.4.3.14	PVMF	3-32
3.4.3.15	REFORMAT	3-34
3.4.3.16	REPEAT	3-34
3.4.3.17	RESIDENT	3-35
3.4.3.18	RESTART	3-37
3.4.3.19	RESTART(WARM)	3-37
3.4.3.20	RESTART(COLD)	3-41
3.4.3.21	RESTART(CLEAN)	3-42
3.4.3.22	RESTORE	3-43
3.4.3.23	SHARED	3-44
3.4.3.24	SPOOL	3-46
3.4.3.25	SYSQM	3-49
3.4.3.26	TIMEDEV	3-50
3.4.3.27	TRC	3-51
3.4.3.28	TVMF	3-51
3.4.4	Date and Time Management	3-52
3.4.5	Device Testing and Volume Recognition at GCOS 7 Restart	3-56
3.4.5.1	General	3-56
3.4.5.2	DV43 Question	3-58
3.4.6	System Ready	3-58
3.4.7	Example of Answer to the IL01 OPTIONS? Question	3-59
3.5	Terminating the Session	3-60
3.5.1	Normal Termination	3-60
3.5.2	Abnormal Termination	3-61
3.5.3	Information Left Over After Termination	3-63
3.6	Error Messages Displayed During System Loading	3-63

4. Operating under GCOS 7

4.1	Introduction	4-1
4.2	Description of an Operator Workstation	4-3
4.3	Connection/Disconnection	4-5
4.4	Distributing Tasks	4-6
4.4.1	Command Environment	4-6
4.4.2	Operator Attributes	4-7
4.4.3	Filter Sets	4-7
4.5	System Functions	4-8
4.5.1	Backing Store Files	4-8
4.5.2	Main Operator Console Log	4-9
4.6	Console Message Handling	4-12
4.7	Filtering System Messages	4-13
4.7.1	What is a Filter?	4-13
4.7.2	Filter Set	4-16



4.7.3	Inclusive, Exclusive, and Mandatory Filters	4-17
4.7.4	Priority of a Filter Set	4-19
4.8	Managing the Network	4-21

5. Running User Jobs under GCOS 7

5.1	Introduction	5-1
5.2	User Jobs - General Concepts	5-2
5.2.1	The Job State Concept	5-2
5.3	Multiprogramming	5-6
5.3.1	The Job Class Concept.....	5-7
5.3.2	The Job Class Group Concept.....	5-9
5.3.3	Priorities	5-10
5.3.3.1	Scheduling Priority (SPR).....	5-11
5.3.3.2	Execution Priority (XPR).....	5-12
5.3.3.3	Output Priority.....	5-12
5.3.4	Service Jobs.....	5-13
5.3.5	The Input Reader	5-16
5.3.6	Input Stream Introduction.....	5-17
5.3.7	The Reader Service Jobs	5-17
5.4	Job Introduction	5-20
5.4.1	Data Enclosure Introduction.....	5-20
5.4.2	Operator Dialog During Job Introduction (\$SWINPUT JCL Statement)	5-21
5.4.3	The JCL Translator	5-22
5.4.4	Hold State	5-22
5.5	Job Scheduling	5-24
5.6	Job Execution	5-26
5.6.1	Events Occurring During Job Execution	5-26
5.6.2	Step Execution	5-27
5.6.3	Step Job Repeat	5-30
5.6.4	Pre-initialized Load Modules.....	5-32
5.6.5	SYSOUT Overflow	5-33
5.6.6	Suspended Jobs	5-34
5.6.7	Before Journal.....	5-35
5.6.8	Journalization Advanced Service (JAS) and the After Journal	5-37
5.7	Job Termination	5-40



5.8	Output Writer	5-41
5.8.1	Outputs	5-42
5.8.1.1	Standard and Permanent Outputs.....	5-42
5.8.1.2	Output Names and Sequence Numbers	5-42
5.8.1.3	Output States.....	5-44
5.8.1.4	Output Commands	5-45
5.8.1.5	Output Priorities (OPR).....	5-46
5.8.1.6	Output Class (OC).....	5-46
5.8.1.7	Output Selection	5-48
5.8.1.8	Question OU02.....	5-48
5.8.1.9	Output Writer Incidents.....	5-50
5.9	Automatic Resource Management (ARM)	5-53
5.9.1	What is ARM?	5-53
5.9.2	Basic ARM and Full ARM	5-55
5.9.3	Concepts	5-56
5.9.3.1	Execution Level (XL)	5-56
5.9.3.2	Multiprogramming Limit (MPL)	5-56
5.9.3.3	Execution Level Class (XLC).....	5-57
5.9.4	Predefined Attributes	5-58
5.9.5	ARM Commands.....	5-59
5.9.5.1	Table of FULL ARM commands	5-59
5.9.5.2	Availability of ARM commands.....	5-60
5.9.6	ARM Operator Messages	5-61
5.10	Domain and CDP (Customer Dedicated Processor) Overview.....	5-62
5.10.1	CDP (Customer Dedicated Processor) Overview.....	5-62
5.10.2	Domain Overview.....	5-63

6. Peripheral Operations

6.1	Introduction	6-1
6.2	Device Handling Overview	6-2
6.2.1	Device States	6-4
6.2.2	Device Allocation in Device Mode.....	6-5
6.2.3	Unknown Device	6-5
6.3	Disk and Tape Operations.....	6-6
6.3.1	Volume Characteristics	6-6
6.3.1.1	Disks	6-6
6.3.1.2	Tapes.....	6-8
6.3.2	Premounting, Mounting and Dismounting.....	6-9
6.3.2.1	Premounting	6-9
6.3.2.2	Normal Mounting	6-10
6.3.2.3	Mounting of Non-removable Volumes	6-11
6.3.2.4	Mounting Cartridge Tape Library Volumes.....	6-11
6.3.2.5	Mounting of Tapes in Any Media Mode (media = *).....	6-12
6.3.2.6	Dismounting.....	6-13



6.3.3	Incidents	6-14
6.3.3.1	Input/output Operation Incidents	6-14
6.3.3.2	Transition to STANDBY (or power off)	6-15
6.3.3.3	Write Protect on tapes	6-16
6.4	Printer Operations	6-18
6.4.1	Paper Form Operations	6-19
6.4.1.1	Paper Premounting	6-19
6.4.1.2	Paper Mounting	6-20
6.4.2	Printer Incidents	6-22
6.5	Hardware Incidents	6-23
6.5.1	Device Incidents	6-23
6.5.2	Peripheral Processor Incidents	6-24
6.5.3	Error Logging	6-25
6.6	Monitoring Device Incidents (Device Trace)	6-27

7. Description of the Operator Commands (Part 1)

7.1	Introduction	7-1
7.1.1	GCL Operator Commands	7-1
7.1.2	Network Commands	7-2
7.1.3	TDS Commands	7-3
7.2	Alphabetical List of GCL Commands	7-4
7.3	Using the GCOS 7 Help Mechanism	7-12
7.4	Detailed Description of GCL Commands	7-14
7.4.1	Command Prefixing	7-14
7.4.2	Three Levels of Command Syntax	7-14
7.4.3	Hidden Parameters or Commands	7-14
7.5	"A" To "C" Commands	7-15
7.5.1	ATTACH_CATALOG (ATT) (ATTACH)	7-15
7.5.2	CANCEL_DUMP (CDP)	7-17
7.5.3	CANCEL_JOB (CJ)	7-18
7.5.4	CANCEL_OUTPUT (CO)	7-22
7.5.5	CANCEL_REQUEST (CR)	7-26
7.5.6	CANCEL_TERMINAL_OUTPUT (CTO)	7-27
7.5.7	CANCEL_USER_REQ (CUR)	7-28
7.5.8	CONNECT_DIMENSION (CNDIM)	7-30
7.5.9	CONNECT_FUNCTION (CNFUNC)	7-32
7.5.10	CONNECT_LOAD (CNLD)	7-34
7.5.11	CREATE_COMPLEX_GENERATION (CRCXGEN)	7-36
7.5.12	CREATE_DIMENSION (CRDIM)	7-38
7.5.13	CREATE_FILTER (CRFLT)	7-39



7.5.14	CREATE_FILTERSET (CRFLTST)	7-42
7.6	"D" Commands	7-43
7.6.1	DEASSIGN_MIRLOG_FILE (DASGMIR)	7-43
7.6.2	DELETE_DIMENSION (DLDIM)	7-44
7.6.3	DELETE_FILTER (DLFLT)	7-45
7.6.4	DISCONNECT_DIMENSION (DISDIM)	7-47
7.6.5	DISCONNECT_FUNCTION (DISFUNC)	7-49
7.6.6	DISCONNECT_LOAD (DISLD)	7-51
7.6.7	DISPLAY_ARS (DARS)	7-52
7.6.8	DISPLAY_ASSIGNED_FILES (DASGF)	7-54
7.6.9	DISPLAY_CLC (DCLC)	7-59
7.6.10	DISPLAY_CLX (DCLX)	7-60
7.6.11	DISPLAY_COMPLEX (DCX)	7-63
7.6.12	DISPLAY_CONFIGURATION (DC)	7-64
7.6.13	DISPLAY_CONSOLE_LOG (DCLOG)	7-68
7.6.14	DISPLAY_DEVIATION_TIME (DDTIME)	7-71
7.6.15	DISPLAY_DEVICE_TRACE (DDVTR)	7-72
7.6.16	DISPLAY_DIMENSION (DDIM) - BASIC ARM	7-74
7.6.17	DISPLAY_DIMENSION (DDIM) - FULL ARM	7-76
7.6.18	DISPLAY_DJP (DDJP)	7-80
7.6.19	DISPLAY_DUMP (DDP)	7-83
7.6.20	DISPLAY_DYNSAVE_SVR (DDYNSVR)	7-85
7.6.21	DISPLAY_ERROR_LOG (DERLOG)	7-86
7.6.22	DISPLAY_FUNCTION (DFUNC)	7-89
7.6.23	DISPLAY_GCL_CACHE (DGC)	7-91
7.6.24	DISPLAY_HARDWARE_CONFIGURATION (DHW)	7-92
7.6.25	DISPLAY_IO_CACHE (DIOC)	7-97
7.6.26	DISPLAY_IOF (DIOF)	7-99
7.6.27	DISPLAY_JAS (DJAS)	7-101
7.6.28	DISPLAY_JOB (DJ)	7-103
7.6.29	DISPLAY_LOAD (DLD)	7-107
7.6.30	DISPLAY_MEMBER (DMB)	7-109
7.6.31	DISPLAY_MFT_FACILITIES (DMFTF)	7-111
7.6.32	DISPLAY_MIRROR_MODE (DMIR)	7-113
7.6.33	DISPLAY_MOT (DMOT)	7-114
7.6.34	DISPLAY_OUTPUT (DO)	7-115
7.6.35	DISPLAY_POWER_TIMER (DPWRM) (DPWR)	7-119
7.6.36	DISPLAY_REGULATION_PARAMETER (DRP)	7-120
7.6.37	DISPLAY_REQUEST (DR)	7-121
7.6.38	DISPLAY_RESTART_OPTIONS (DRSO)	7-122



7.6.39	DISPLAY_RMS (DRMS).....	7-126
7.6.40	DISPLAY_SECURITY_OPTIONS (DSO).....	7-127
7.6.41	DISPLAY_SERVICE (DSRV).....	7-131
7.6.42	DISPLAY_SSI (DSSI).....	7-133
7.6.43	DISPLAY_STORAGE_FACILITIES (DSF).....	7-134
7.6.44	DISPLAY_SUBMITTER (DS).....	7-136
7.6.45	DISPLAY_SYSTEM_TRACE (DSYSTR).....	7-141
7.6.46	DISPLAY_TERMINAL_OUTPUT (DTO).....	7-143
7.6.47	DISPLAY_TERMINAL_WRITER (DTW).....	7-145
7.6.48	DISPLAY_TIME (DTM).....	7-147
7.6.49	DISPLAY_USER_REQ (DUR).....	7-148
7.6.50	DISPLAY_XL_CLASS (DXLC).....	7-150
7.6.51	DPRINT (DPR).....	7-152
7.7	"E" Commands	7-157
7.7.1	ENTER_FILETRANS_REQ (EFTR).....	7-157
7.7.2	ENTER_JOB_REQ (EJR) (EJ).....	7-163
7.7.3	ENTER_LIBTRANS_REQ (ELTR).....	7-173
7.7.4	EXEC_CLS_COMMAND (EXCLS).....	7-180
7.7.5	EXEC_CLX_VOLRPT (VOLRPT).....	7-182
7.7.6	EXECUTE_DIRECTIVE (EXDIR).....	7-184
7.8	"F" and "G" Commands.....	7-185
7.8.1	FORCE_JOB (FJ).....	7-185
7.8.2	FORCE_OUTPUT (FO).....	7-187
7.8.3	FORCE_SERVER_TIMEOUT (FSVRTO).....	7-188
7.8.4	FORCE_USER_REQUEST (FUR).....	7-189
7.8.5	GET_CLX_FILE (GTCLXF).....	7-191

8. Description of the Operator Commands (Part 2)

8.1	"H" and "I" Commands	8-1
8.1.1	HOLD_GCL_CACHE (HGC).....	8-1
8.1.2	HOLD_JOB (HJ).....	8-2
8.1.3	HOLD_OUTPUT (HO).....	8-4
8.1.4	HOLD_TERMINAL_OUTPUT (HTO).....	8-8
8.1.5	HOLD_USER_REQ (HUR).....	8-9
8.1.6	INIT_SYSTEM_TRACE (ISYSTR).....	8-10
8.2	"L" Commands	8-11
8.2.1	LIST_DUMP (LSDP).....	8-11
8.2.2	LIST_FILTER (LSFLT).....	8-12



8.3	"M" Commands	8-14
8.3.1	MAIL	8-14
8.3.2	MODIFY_CLC (MDCLC).....	8-17
8.3.3	MODIFY_CLX (MDCLX).....	8-18
8.3.4	MODIFY_CONFIGURATION (MDC) - STATION OPERATOR.....	8-20
8.3.5	MODIFY_CONFIGURATION (MDC) - MAIN OPERATOR	8-22
8.3.6	MODIFY_DEVICE_TRACE (MDDVTR)	8-24
8.3.7	MODIFY_DIMENSION (MDDIM) - BASIC ARM.....	8-26
8.3.8	MODIFY_DIMENSION (MDDIM) - FULL ARM.....	8-27
8.3.9	MODIFY_DJP (MDDJP)	8-29
8.3.10	MODIFY_ERROR_LOG (MDERLOG).....	8-32
8.3.11	MODIFY_FILTER (MDFLT)	8-33
8.3.12	MODIFY_HARDWARE_CONFIGURATION (MDHW)	8-35
8.3.13	MODIFY_INPUT_READER (MDIR).....	8-40
8.3.14	MODIFY_IOF (MDIOF)	8-41
8.3.15	MODIFY_JOB (MDJ)	8-43
8.3.16	MODIFY_LOAD (MDLD).....	8-46
8.3.17	MODIFY_MIRROR_MODE (MDMIR).....	8-48
8.3.18	MODIFY_MOT (MDMOT).....	8-49
8.3.19	MODIFY_OPERATING_MODE (MDOM).....	8-50
8.3.20	MODIFY_OUTPUT (MDO, MO).....	8-51
8.3.21	MODIFY_POWER_TIMER (MDPWRTM) (MDPOWER) (TPW) (MDPWR)....	8-56
8.3.22	MODIFY_RESTART_OPTIONS (MDRSO)	8-58
8.3.23	MODIFY_STATION (MDST).....	8-60
8.3.24	MODIFY_SYSTEM_TRACE (MDSYSTR) (MSYSTR)	8-63
8.3.25	MODIFY_TERMINAL_OUTPUT (MDTO).....	8-65
8.3.26	MODIFY_TERMINAL_WRITER (MDTW).....	8-67
8.3.27	MODIFY_TIME (MDTIME) (MDTM).....	8-70
8.3.28	MODIFY_USER_REQ (MDUR).....	8-72
8.3.29	MODIFY_VOLUME (MDV)	8-75
8.3.30	MODIFY_XL_CLASS (MDXLC).....	8-77
8.3.31	MWENV (Modify Working Environment)	8-79
8.4	"N" and "R" Commands.....	8-80
8.4.1	NAME_VOLUME (NV)	8-80
8.4.2	RELEASE_GCL_CACHE (RGC).....	8-81
8.4.3	RELEASE_JOB (RJ).....	8-82
8.4.4	RELEASE_OUTPUT (RO).....	8-84
8.4.5	RELEASE_TERMINAL_OUTPUT (RTO)	8-88
8.4.6	RESYNC_MIRROR_SHARED (RSCMIR).....	8-90
8.4.7	RELEASE_USER_REQ (RUR)	8-91



8.4.8	REPLY (REP).....	8-93
8.5	"S" Commands	8-94
8.5.1	SEND	8-94
8.5.2	SEND_ERROR_LOG (SNDRLOG).....	8-96
8.5.3	START_ARS (SARS).....	8-97
8.5.4	START_CLC (SCLC).....	8-98
8.5.5	START_CLX (SCLX).....	8-99
8.5.6	START_CMSR (SCMSR).....	8-100
8.5.7	START_CONSOLE_LOG (SCLOG).....	8-101
8.5.8	START_DEVICE (SDV).....	8-102
8.5.9	START_DEVICE_TRACE (SDVTR).....	8-103
8.5.10	START_DYNSAVE_SVR (SDYNSVR).....	8-106
8.5.11	START_ERROR_LOG (SERLOG).....	8-107
8.5.12	START_INPUT_READER (SIR).....	8-108
8.5.13	START_IO_CACHE (SIOC).....	8-109
8.5.14	START_LOAD (SLD).....	8-111
8.5.15	START_MEMBER (SMB).....	8-113
8.5.16	START_OUTPUT_WRITER (SOW) (SO).....	8-114
8.5.17	START_POWER_TIMER (SPWRM).....	8-120
8.5.18	START_RMS (SRMS).....	8-121
8.5.19	START_SERVICE (SSRV).....	8-123
8.5.20	START_SSI (SSSI).....	8-126
8.5.21	START_SYSTEM_TRACE (SSYSTR).....	8-127
8.5.22	START_TERMINAL_OUTPUT (STO).....	8-130
8.5.23	START_TERMINAL_WRITER (STW).....	8-131
8.5.24	SWITCH_MIRROR (SWMIR).....	8-132
8.6	"T" Commands	8-133
8.6.1	TAKEOVER_MEMBER (TKMB).....	8-133
8.6.2	TERMINATE_ARS (TARS).....	8-136
8.6.3	TERMINATE_CLC (TCLC).....	8-137
8.6.4	TERMINATE_CLX (TCLX).....	8-138
8.6.5	TERMINATE_CMSR (TCMSR).....	8-139
8.6.6	TERMINATE_CONSOLE_LOG (TCLOG).....	8-140
8.6.7	TERMINATE_DEVICE_TRACE (TDVTR).....	8-141
8.6.8	TERMINATE_DYNSAVE_SVR (TDYNSVR).....	8-143
8.6.9	TERMINATE_ERROR_LOG (TERLOG).....	8-145
8.6.10	TERMINATE_INPUT_READER (TIR).....	8-146
8.6.11	TERMINATE_IO_CACHE (TIOC).....	8-147
8.6.12	TERMINATE_LOAD (TLD).....	8-148
8.6.13	TERMINATE_MEMBER (TMB).....	8-150



8.6.14	TERMINATE_OUTPUT_WRITER (TOW)	8-152
8.6.15	TERMINATE_POWER_TIMER (TPWRM)	8-155
8.6.16	TERMINATE_RMS (TRMS)	8-156
8.6.17	TERMINATE_SERVICE (TSRV)	8-158
8.6.18	TERMINATE_SSI (TSSI)	8-161
8.6.19	TERMINATE_SYSTEM (TSYS)	8-162
8.6.20	TERMINATE_SYSTEM_TRACE (TSYSTR)	8-164
8.6.21	TERMINATE_TERMINAL_OUTPUT (TTO)	8-166
8.6.22	TERMINATE_TERMINAL_WRITER (TTW)	8-168
8.6.23	TEST_DEVICE (TSTDV)	8-169
8.7	"U" Commands	8-170
8.7.1	UNLOCK_MIRLOG_FILE (UNLMIRF)	8-170
8.7.2	UNLOCK_MIRROR_VOLUME (UNLMIR)	8-172
8.7.3	UNPAIR_MIRROR_VOLUME (UNPMIR)	8-174

9. Remote Multiplexed Operator Support (RMOS)

9.1	Introduction	9-1
9.1.1	Definition of RMOS	9-2
9.1.2	Functions	9-3
9.1.3	Basic Architecture	9-3
9.1.4	RMOS Operator Levels	9-5
9.1.4.1	Command Level	9-5
9.1.4.2	Display Level	9-5
9.1.4.3	Select Level	9-6
9.2	Description of RMOS Commands	9-7
9.2.1	RMOS (ENTER RMOS)	9-8
9.2.2	CTL (CONTROL)	9-10
9.2.3	D (DISPLAY)	9-12
9.2.4	IT (INTERRUPT)	9-14
9.2.5	LK (LINK)	9-15
9.2.6	PRG (PURGE)	9-20
9.2.7	Q (QUIT)	9-22
9.2.8	SEL (SELECT)	9-23
9.2.9	UNLK (UNLINK)	9-25
9.3	Leaving RMOS	9-27
9.4	Description of Screens	9-28
9.4.1	Global Status Screen Layout	9-28
9.4.2	Global Requests Screen Layout	9-31



9.5	RMOS Error Messages	9-34
9.5.1	Initialization Error Messages	9-34
9.5.2	Command Processing Error Messages	9-36
9.5.3	VCAM Error Messages	9-37
9.5.4	File Processing Error Messages	9-40
9.5.5	Miscellaneous Error Messages	9-42

10. Reconfiguration

10.1	DPS 7000 Reconfiguration	10-1
10.1.1	Overview	10-1
10.1.1.1	System Architecture and Resources	10-3
10.1.1.2	When and How to Configure or Reconfigure a System	10-5
10.1.2	Reconfiguration of Central Subsystem	10-6
10.1.2.1	CSS Resources	10-6
10.1.2.2	Resource States	10-9
10.1.2.3	Machine Incidents	10-11
10.1.2.4	GCOS 7 Reconfiguration Commands (MDHW and DHW)	10-12
10.1.2.5	Dynamic Reconfiguration and System Restart	10-12
10.1.3	Reconfiguration of Peripheral Subsystems	10-14
10.1.3.1	Peripheral SubSystem	10-14
10.1.3.2	Physical Connections	10-16
10.1.3.3	Device States and Device Path States	10-23
10.1.3.4	Peripheral Incidents	10-25
10.1.3.5	GCOS 7 Reconfiguration Commands (DHW and MDHW)	10-27
10.1.3.6	Dynamic Reconfiguration and System Restart	10-28
10.2	Reconfiguration on DPS 7000/XTA models	10-30
10.2.1	Overview	10-30
10.2.1.1	When and How to Configure or Reconfigure a System	10-34

A. Operator-System Dialog on a Console

B. Calling System Utilities

B.1	DUMPACT	B-1
B.2	PRLOG	B-3
B.3	PRLOGC	B-3
B.4	TD (Tests and Diagnostics)	B-4
B.5	DP_SAVE (Saving GCOS 7 Anomaly Files)	B-5
B.5.1	Introduction	B-5
B.5.2	Processing	B-7



B.5.3	Syntax of the EXECUTE_DP_SAVE Directive	B-7
B.5.3.1	Parameters Displayed on the Primary Screen	B-7
B.5.3.2	Parameters Displayed on a Secondary Screen	B-8
B.5.3.3	Parameters Displayed on an additional Secondary Screen	B-10
B.5.4	Description of Parameters	B-10
B.5.5	Elements Saved	B-14
B.5.6	Examples	B-17
B.5.6.1	Examples of Commands	B-17
B.5.6.2	Examples of Contexts	B-19
B.6	DP_PRINT (Printing a Dump on a Line Printer)	B-21
B.7	SVLOGF (Saving TDS/UFAS/GAC Logfile)	B-23
B.8	PRINTJOR (Print the JOR of a Job in Execution)	B-24
B.9	V7000 Information Collector	B-26

C. Index of Commands by "Object"

D. The SYS.URCINIT Printer Control File

D.1	Introduction	D-1
D.2	General Programming Considerations	D-2
D.2.1	The SYS.URCINIT File	D-2
D.2.2	The URINIT Utility	D-3
D.2.2.1	The URINIT Statement	D-3
D.2.2.2	JCL Card Arrangement	D-3
D.2.2.3	URINIT Commands	D-4
D.2.2.4	Command Restrictions	D-9
D.2.3	Parameter Overriding	D-11
D.2.4	Protected Character Strings	D-11
D.2.5	Direct Use of *Devices	D-11
D.2.6	Related Information	D-11
D.3	Line Printer	D-12
D.3.1	Device Naming and Allocation	D-12
D.3.2	Printer Attributes	D-14
D.3.3	Printing Procedures	D-16
D.3.3.1	Direct Use of the Printer	D-17
D.3.3.2	Use of a Standard SYSOUT Subfile	D-17
D.3.3.3	Use of a Permanent SYSOUT File	D-19
D.3.4	Index Table - as an Alternative Character Set	D-19
D.3.5	Printer Character Sets	D-22
D.3.5.1	Standard Index Tables	D-27
D.3.5.2	KPRU Parameter	D-27



D.3.6	Belt Definition	D-28
D.3.6.1	BPRU Parameters	D-28
D.3.6.2	IPRU Parameters	D-29
D.3.7	Form Number Encoding.....	D-30
D.3.8	Editing Table - as an Alternative to Form-Name.....	D-31
D.3.9	Form Definition.....	D-31
D.3.9.1	Standard Editing Table	D-31
D.3.9.2	EPRU Parameters	D-32
D.3.9.3	FPRU Parameters	D-33
D.3.10	Vertical Format Definition.....	D-36
D.3.11	Printing Test Definition.....	D-37
D.3.12	Parameter Assignment Restrictions.....	D-39
D.3.13	Parameter Overriding.....	D-40
D.3.14	ASA Data Format.....	D-41
D.3.14.1	General Programming Notes.....	D-41
D.3.14.2	ASA Format Using Standard SYSOUT Subfiles	D-41
D.3.14.3	ASA Format Using Permanent SYSOUT Files.....	D-42

E. WHEN, EVERY and RRESTART

E.1	WHEN	E-1
E.1.1	Absolute Addressing	E-1
E.1.2	Relative Addressing	E-3
E.1.3	Fast Options.....	E-3
E.1.4	FORCE Option.....	E-4
E.2	EVERY	E-5
E.3	RRESTART	E-7

F. Device Types and Classes

G. How GCOS 7 Handles Dates Before and After the Year 2000

G.1	If the Year Is Entered In Two Digits	G-1
G.2	Entering YEARS in GCL	G-2

H. Queued Writer

H.1	Queued Writer Overview.....	H-1
H.2	Definitions of Terms	H-2
H.3	Activation of the Queued Writer	H-3
H.4	Name of the Subfile to be Printed	H-5
H.5	JCL_CONTROL Subfile	H-6



H.6 Termination of the Queued Writer..... H-7

Glossary

Index



Table of Graphics

Figures

Figure 1-1.	DPS 7000/XTA General Architecture.....	1-12
Figure 1-2.	DPS 7000/XTA cabinet	1-13
Figure 1-3.	V7000 Description	1-17
Figure 2-1.	The DPS 7000/A Control Units	2-3
Figure 2-2	The DPS 7000/2X0 Control Elements	2-11
Figure 2-3	The DPS 7000/3X0 Control Elements	2-12
Figure 2-4	ASPA Configuration Screen.....	2-15
Figure 2-5	DPS 7000/4xx/Cxx, DPS 7000/5x0/7xx/8x0/Dx0/Mx0 and DPS 7000/MTxx/TAxxx/TAxxxC Power On Procedure	2-19
Figure 2-6	Screen Presentation.....	2-21
Figure 2-7	The DPS 7000/4xx/Cxx/MTx1 Control Units.....	2-24
Figure 2-8	The DPS 7000/5x0/7xx Control Units	2-25
Figure 2-9	The DPS 7000/8x0/Dx0/Mx0/MTxx/MT42 and up/TAxxx/TAxxxC Control Units.....	2-26
Figure 2-10	DPS 7000/XTA Control Panel.....	2-28
Figure 2-11	DPS 7000/XTA Windows™ Start Menu	2-30
Figure 2-12	V7000 System Control screen image	2-31
Figure 2-13.	V7000 Properties screen image.....	2-32
Figure 2-14	SYC Action menu screen image	2-34
Figure 2-15.	Dialog for automatic start of LN02 GCOS7 console	2-35
Figure 2-16.	V7000 Operation Supervisor screen image	2-35
Figure 2-17	GCOS 7 console screen image.....	2-36
Figure 3-1.	Time Management	3-55
Figure 4-1.	Control Consoles.....	4-2
Figure 4-2.	Using Filters	4-13
Figure 4-3.	Setting up Filter Sets and Filters	4-15
Figure 5-1.	The Job State Concept	5-3
Figure 5-2.	Job State Messages.....	5-5
Figure 5-3.	Job Introduction.....	5-18
Figure 5-4.	Output States	5-43
Figure 5-5.	ARM Mechanism.....	5-54
Figure 9-1.	Screen Format Transitions.....	9-27
Figure 9-2.	Display Global Status.....	9-28
Figure 9-3.	Link Status Transitions.....	9-30
Figure 9-4.	Display Global Requests.....	9-31
Figure 10-1.	System Architecture	10-3



Figure 10-2.	Peripherals - Physical and Logical	10-15
Figure 10-3.	Example of Cartridge Subsystem	10-16
Figure 10-4.	Level of Simultaneity (LOS)	10-18
Figure 10-5.	Access Paths.....	10-20
Figure 10-6.	Device Sharing	10-22
Figure 10-7.	DPS7000 /XTA : GCOS7 System Architecture.....	10-31
Figure D-1.	Item Links	D-6
Figure D-2.	Item Links	D-18
Figure D-3.	Character Set Rules.....	D-22
Figure D-4.	Index Tables Rules	D-23
Figure D-5.	Form Parameters	D-35

Tables

Table 2-1	Functions of the DPS 7000/2X0 and 3X0 System Operator Panel ..	2-14
Table 2-2	LED information	2-29
Table 3-1.	Time and Date Algorithm	3-54
Table 5-1.	Job States	5-4
Table 5-2.	Job Display and Control Commands	5-6
Table 5-3.	Job Classes: Standard Attributes and Recommended Usage	5-8
Table 5-4.	Job Class Commands	5-10
Table 5-5.	Scheduling Priority Commands	5-11
Table 5-6.	Execution Priority Commands	5-12
Table 5-7.	Service Jobs (1/2).....	5-14
Table 5-8.	Service Job Commands	5-16
Table 5-9.	Job Scheduling Commands	5-24
Table 5-10.	Step Abort; Severities and their Causes	5-31
Table 5-11.	Output Commands	5-45
Table 5-12.	Output Priority Commands	5-46
Table 5-13.	Output Class Priority Commands	5-46
Table 5-14.	Output Class Commands	5-47
Table 5-15.	Standard Dimension Default Attributes	5-58
Table 5-16.	FULL ARM Commands	5-59
Table 5-17.	Availability of ARM Commands	5-60
Table 9-1.	Summary Description of RMOS Commands.....	9-1
Table 9-2.	Summary of RMOS Tasks.....	9-6
Table 10-1.	Dynamic Reconfiguration	10-13
Table B-1.	SYSTEM_CRASH, OPEN7 and OTHER_PROBLEM Anomaly Parameters and Elements Saved by DP_SAVE	B-14
Table B-2.	NET_ANOMALY Parameters and Element Saved by DP_SAVE (1/2).....	B-15
Table D-1.	URINIT Items.....	D-5
Table D-2.	Printer Characteristics	D-13
Table D-3.	Printer Attributes	D-14
Table D-4.	Character Sets and their Belt Images	D-26
Table D-5.	Test Format Control Bytes (FORMCTL).....	D-38
Table D-6.	Print Test Overriding.....	D-40



1. Introduction to Running a System under GCOS 7

1.1 Overview

Running a system under GCOS 7 comprises the following three phases:

1. **Initialization** and Power-On of the system, that is:
 - applying power to the hardware;
 - initializing the hardware by loading firmware (micro-programs);
 - loading the operating system, GCOS 7, into memory;
 - initializing this operating system.
2. The **operation** phase proper:
 - starting and monitoring jobs until their output is delivered;
 - handling peripheral devices - printers, disk drives etc.;
 - managing certain system files;
 - correcting software and hardware incidents.
3. **Shutdown** of the system.

These phases are explained in Chapters 2 to 6 of this manual. This chapter provides a general introduction to:

- the visible elements of a DPS 7000 system;
- the role of a main operator;
- the dimensions of GCOS 7.



1.2 DPS 7000 Systems - Hardware Overview

1.2.1 DPS 7000/2x0 and /3x0 Systems

- **The Core Cabinet:** The Core Cabinet is made up of a Power Module, the Central Sub-System (CSS) containing Central Processing Unit(s) (CPU), Main Memory Units (MMU) and Physical Channel Multiplexor Units; (PXU), the Service Processor (SPA), and an optional CNP7 (Communications Network Processor) for non-redundant models only (DPS 7000/2x0).
- **System/Service Console:** The console consists of a screen, keyboard, logic pack and printer. Redundant models (DPS 7000/340/360/380) have two system/service consoles.
- **Peripheral Subsystems:** Peripheral subsystems consist of peripheral processors (or controllers) and the devices they control. The DPS 7000/2x0 and /3x0 have the following range of standard peripheral subsystems:
 - MSS 4102 Mass Storage Subsystem(s);
 - MTS 4101 Magnetic Tape Subsystem(s);
 - DATANET 713X Network Processor(s);
 - MP XXXX Non-impact printing subsystem(s);
 - PRU 41XX Impact printers directly controlled by the SPA.

The Service Processor

The Service Processor (SPA) of the DPS 7000 is an independent unit running under its own Operating System, SPOS. DPS 7000/340/360/380 systems include two SPAs, a master and a back-up. Functions of the SPA include: initialization and testing of the Central Subsystem(s), maintenance and reconfiguration operations, control of the service console, and networking interface. The SPA can be accessed from Bull Technical Centers for remote maintenance purposes.

In addition to the specific controller, connected by a PSI to the system, in which the SPA software can run, the SPA supports:

- The MSCI (Maintenance Serial Channel Interface) to which are connected maintenance stations:
 - the MPA (Maintenance Processor Adapter), which is the maintenance station of the CSS,
 - the Power Monitor, which controls the power system.
- A fixed mini-disk, and a 5.25" diskette unit.
- The console subsystem.
- Printer connections through the PDSI interface.

On-line Tests and Diagnostics (OLTD) for Peripheral Subsystems and devices are controlled by GCOS 7 from the System console.



Operator Controls

These controls are described in more detail in Chapter 2 *Starting Up Your System*.

- The System Operator Panel (SOP) is located on the core cabinet, to switch the system on or off, or to re-initialize.
- The system console controls the operation of the system. On dual-cabinet systems (DPS 7000/340/360/380 only), two consoles are available.
- Maintenance panels, located in various processor cabinets, control the peripherals.

1.2.2 DPS 7000/A Systems

The DPS 7000/A is a compact system designed to enable both processing and peripheral functions to be integrated into a single cabinet, including not only the Central Sub-system and power units, but also 5.25" disks and cartridge tape units. The largest models also have an extension cabinet. Added to this, the following external peripherals can be connected if required:

- MTS 4101 magnetic tape subsystem.
- MP xxxx Non-Impact Printer.
- PRU 41xx Impact Printer(s).

Operator controls

These controls are described in more detail in Section 2 *Starting Up Your System*.

- The System Operator Panel (SOP) on the core cabinet, to send commands such as Power-on, Power-off, and Reset.
- The Local Console is optional; it is implemented on a micro-computer, and presents a window-type interface.



1.2.3 DPS 7000/5x0/7xx/8x0/Dx0/Mx0 Systems

A DPS 7000/5x0/7xx/8x0/Dx0/Mx0 system consists of a Central SubSystem (CSS), one to eight Input/Output SubSystems (IOSS), one or two Service Processors, two remote consoles (Remote Workstation and RSC) and several peripheral subsystems.

These peripheral subsystems include:

- Mass storage subsystems;
- Magnetic tape subsystems;
- Data communication subsystems;
- Printer subsystems.

The multiprocessor architecture of the Central Subsystem enables the DPS 7000/5x0/7xx/8x0/Dx0/Mx0 system to offer a wide range of configurations.

1.2.3.1 DPS 7000/5x0/7xx Systems

Central Subsystem (CSS)

The Central Subsystem (CSS) is located in the CSS cabinet. It includes:

- A System Controller Unit (SCU) which connects the System Bus (ASB) with the Memory Bus (AMB);
- Up to six Central Processing Units (CPU) plugged into the ASB;
- One to eight Memory Units plugged into the AMB;
- One or two Input/Output Units (IOU) plugged into the ASB;
- A Clock and Maintenance Unit (CMU).

Input/Output Subsystem

The Input/Output Subsystems are connected to the Central Subsystem (CSS) through one to four I/O busses (IOB) and to the peripheral subsystems through specific or standard interfaces. There are from one to eight Input/Output Subsystems (IOSS) each with a UMI Interface Adapter (UIA) and three Channel Multiplexer Units (CXU).



Power Subsystem Overview

The power subsystem is based on the Auxiliary Function Service (AFS). There is an AFS in each CSS or IOSS cabinet and its basic components are:

- an AC control and filter module (ACC);
- one or several Power converter Modules (PM500);
- one or two Fan and Service Power modules (FSP);
- a system ENVIRONMENT module (ENVI)
- a LOGistic control module (LOG);
- a Service Operator Panel (SOP).

Optional features:

- a Hold Up Power module (HUP);
- an Electrical Motor Generator (EMG) to give AC power to a Service Processor.

1.2.3.2 DPS 7000/8xx/Dx0/Mx0 Systems

Central Subsystem (CSS)

The Central Subsystem (CSS) is located in the CSS cabinet. It includes:

- A System Controller Unit (SCU) which connects the System Bus (ASB) with the Memory Bus (AMB);
- up to 6 CPU slots, plugged into the ASB. Each slot can have four CPUs (Central Processing Units) that can be configured as IPs or EPs. (EPs are co-processors).
- One to eight Memory Units (MU) plugged into the AMB;
- One or two Input/Output Units (IOU) plugged into the ASB;
- A Clock and Maintenance Unit (CMU).

Input/Output Subsystem (IOSS)

The Input/Output Subsystems are connected to the Central Subsystem (CSS) through one to four I/O busses (IOB) and to the peripheral subsystems through specific or standard interfaces.

The Input/Output Subsystems consist of one to eight IOSSs:

- two MFB-IOSSs (Multi Function Bus) in the CSS cabinet,
- and 4 to 6 MFB IOSSs in the extension cabinets.



Power Subsystem Overview

The AFS (Auxiliary Function Service) provides the electrical power and cooling for DPS 7000 systems is based on the Auxiliary Function Service (AFS). Each cabinet contains one AFS sub-system.

The AFS sub-system includes the following items:

- one AC control and filter module (ACC);
- one or more Power Modules (PM5P/PM700 and PMD);
- one Power module for fan and Standby Power supply (FSP);
- an ENVironment control module (ENVI)
- a LOGistics control module and fan DC power (LOG);
- a Service Operator Panel (SOP).
- a Hold Up Power module (HUP) to provide energy for voltage sag (up to 400 milliseconds);
- a redundant AFSR (except on DPS 7000/Cxx and /8xxS).

1.2.4 The DPS 7000/4xx/Cxx Systems

The DPS 7000/4xx/Cxx incorporates the powerful central processor of the DPS 7000/5x0/7xx systems with the miniaturized internal peripherals first introduced on the DPS 7000/A systems.

1.2.4.1 The DPS 7000/4xx/Cxx Systems

A DPS 7000/4xx/Cxx system includes one or two cabinets:

- The basic cabinet, (which can be a complete configuration on its own). It contains:
 - a Central Sub-System (CSS),
 - two Input/Output Sub-Systems (IOSS),
 - up to nine disks and one or two streamers.
- The extension cabinet. It contains, chiefly, an extra set of up to 15 disks, and two Input/Output Sub-Systems (IOSS).

A Bull DPS 7000/4xx/Cxx system maximum configuration has one Central Sub-System (CSS), up to four Input/Output Sub-systems (IOSS), a Service Processor with a local console, two remote consoles (the RSC console and the remote maintenance console) and several peripheral sub-systems.



1.2.4.2 The Subsystems (CSS, IOSS, etc)

Central Subsystem (CSS)

The Central Subsystem (CSS) is located in the CSS cabinet. It includes:

- A System Controller Unit (SCU) which connects the System Bus (ASB) with the Memory Bus (AMB);
- up to 3 CPUs;
- One to four Memory Units (MU) plugged into the AMB;
- One Input/Output Units (IOU) plugged into the ASB;
- One or two Service Processors;
- A Clock and Maintenance Unit (CMU).

Input/Output Subsystem

The components of the Input/Output Sub-System are listed below. The IOSS is also illustrated by a diagram later in this chapter.

- up to four **Multibus2 Interface Adaptors** (MIA) that connect the IOU of the system to the Subsystem controller.
- two **Multibus2 Maintenance Processors** (MMP) which are common to the four IOSSs in the two Basic and Extension Cabinets.

Different sub-system controllers can be attached to the Multibus2 interface:

- up to six **Mass Storage Processors** (MSP). The ORU name for a SCSI drive and streamer controller is MSPD.
- up to two **Local Network Modules** (LNM) which provide the connection to the CNP7-2 (dual LAN access) and up to forty lines.
- an **FCP7** controller for optical fiber link connection.
- PNM is a **PSI-(N) adaptor** for any peripheral sub-system that requires a PSI interface.
- PSM is a **PSI-(S) adaptor** for any peripheral sub-system that requires a PSI-S interface.
- a **WSP** controller for Wide SCSI disk or cartridge subsystem connections.



- **Printer interfaces** can be either PRM (PRinter Module) or CPM (Centronics Printer Module). These two boards are used to connect printers that comply with:
 - Line Printer Programming Manual
 - Printronix (P series) and IGP (Intelligent Graphic Processor).

Service Processor

The Service Processor is based on a personal computer running specific applications under MS/DOS and WINDOWS™ type environments.

Auxiliary Functions System (AFS)

The AFS (Auxiliary Function Service) provides the electrical power and cooling for DPS 7000 systems. Each cabinet contains one AFS sub-system.

The AFS sub-system includes the following items:

- one AC controller and filter module (ACC),
- one or more power modules (PM5P/PM700 and PMD),
- one power module for fan and Standby Power supply (FSP),
- an environment control module (ENVI),
- a logistics control module and fan DC power (LOG),
- a Service Operation Panel (SOP),
- a Hold Up Power module (HUP) to provide energy for voltage sag (up to 400 milliseconds).
- a back-up battery to protect the system against mains power failure shorter than ten minutes

Peripheral Subsystems

Disk Subsystems:	MSP with 2.5 GB disks LSS shared with an Auriga system MSC4B with FSD500 or HI1000 disks (reconnection)
Tape Subsystems	ATP, CMTS, MTC-G (reconnection)
Data Communications Subsystems	Datanet, CNP7, Extended-CP7, FCP7, OCS.
Unit Record Subsystems	MPC (SDI support for the ATP)



Printers	PDSI interface - PR54, PR88, MP5050, MP6060, MP6090 Centronics interface - PR90 RS 232 and Dataproducts interface - PR800, PR900, PR701, PRU801, PRU901.
----------	--

1.2.5 DPS 7000/MTxx/TAxxx/TAxxxC Systems

1.2.5.1 DPS 7000/MTx1 Non-redundant Models

The central processor subsystem (CPS) of the Bull DPS 7000/MT11, MT21, MT31, MT41, MT51, MT61 non-redundant models comprises the CSS cabinet and one service processor.

The CSS cabinet contains:

- one Central Processor Unit,
- 128 MB of main memory,
- one Input/Output Controller Unit,
- one Multi-Function Bus (MFB) enabling connection of up to 8 MFB channel controllers,
- a power subsystem and power hold-up unit,
- one MSPD controller with one cartridge streamer of 5 GB.

The service processor (without hard-copy) can be located up to 15 meters away from the CSS cabinet.

The basic and extension cabinets contain integrated peripherals (disks, cartridge streamer and network processor).

System configurability limits

The Bull DPS 7000/MTx1 non redundant systems are mono- or (optionally) bi-cabinet configurations with up to three optional Customer Dedicated Processors (CDP), up to 1024 MB of main memory and up to 32 MFB channel controllers.

CPS Options

All current MFB controllers are offered on the Bull DPS 7000/MT non-redundant models.

All non-redundant models can receive one extension cabinet depending on customer needs for reconnection and new peripherals.



Peripheral Offer

The new SDA 7 (Small disk Array 7) offer provides customers with the RAID technology. These disks are connectable to open systems as well.

It is possible to reconnect current peripherals of the Bull DPS 7000/Cxx series to the Bull DPS 7000/MT non-redundant models, except the CDA (Cached Disk Array).

1.2.5.2 DPS 7000/MTxx/TAxix Redundant Models and TAxixC models

The central processor subsystem (CPS) of the Bull DPS 7000/MT42, MT52, MT62, MT72, MT82, MT83, MT84, MT86, MT88, MT92, MT96 TAxix redundant models comprises the CSS cabinet, one I/O cabinet and two service processors.

The CSS cabinet contains:

- two or more Central Processor Units, depending on the model,
- two CDP1 (Type-1 Customer Dedicated Processor),
- one back-up CPU board,
- 256 MB of main memory,
- two Input/Output Controller Units
- one Dual Multi-Function Bus (MFB) enabling connection of up to 16 MFB channel controllers,
- a power subsystem (with redundant power supply) and power hold-up unit.

The I/O cabinet has one Dual Multi-Function Bus (MFB) that enables connection of up to 16 MFB channel controllers

The two service processors (without hard-copy) can be located up to 15 meters away from the CSS cabinet.

System configurability limits

The Bull DPS 7000/MTxx redundant systems are bi- or (optionally) tri- cabinet configurations with two or (optionally) up to eight Customer Dedicated Processors (CDP), up to 2048 MB of main memory (with Bi-System Facility or Millennium Bi-System Test Facility) and up to 64 MFB channel controllers.

The TAxixC is identical but non redundant.



CPS Options

All current MFB controllers are offered on the Bull DPS 7000/MT redundant models.

All DPS 7000/MT and TAxxx redundant models can receive one additional I/O cabinet depending on customer needs for reconnection and new peripherals.

To help resolve year 2000 problems, the Bull DPS 7000/MT/TAxxx redundant models can receive a factory-fitted option named Millennium Bi-System Test Environment which enables customers to test applications of the third millennium without disturbing their production system.

Peripheral Offer

All current peripherals of the Bull DPS 7000/8x0/Mx0 series are available with the Bull DPS 7000/MTxx/TAxxx redundant models.

System configurability limits for peripherals are the same as those of the Bull DPS 7000/Mx0 series.

It is possible to reconnect current peripherals of the Bull DPS 7000/8x0/Mx0 series to the Bull DPS 7000/MTxx/TAxxx redundant models and TAxxxC models, except MSC4 disk controllers with 1 GB disks.



1.2.6 DPS 7000/XTA systems

DPS 7000/XTA system architecture is a new concept (eXtended Twin Architecture): the GCOS 7 operating system runs on an Intel™ processor server, concurrently with Windows™ operating system. It is a major evolution of the Twin Architecture (DPS 7000/ TA series).

The figure below describes this new architecture:

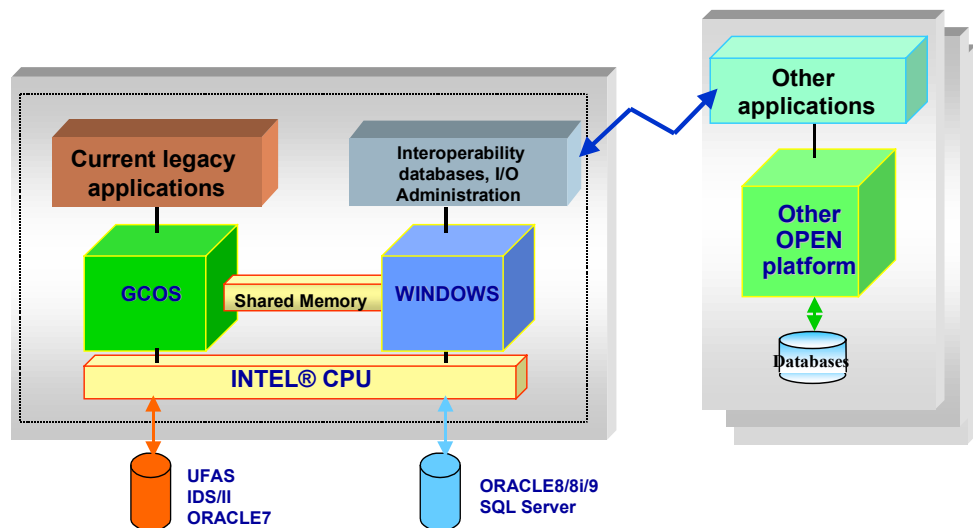


Figure 1-1. DPS 7000/XTA General Architecture

1.2.6.1 DPS 7000/XTA cabinet description

The DPS 7000/XTA is a compact system to enable both processing and disk peripheral functions to be integrated into a single cabinet.

The 19" cabinet (19 U height) includes:

- The DPS 7000 Base System Unit (BSU) server itself.
- A Power Distribution Unit (PDU) to bring electrical power to the cabinet components.
- An optional disk subsystem based on a RAID DAS subsystem. This disk subsystem is configured either as a Data Integrity (DI) configuration or as a High Availability (HA) configuration and may have a disk extension rack.

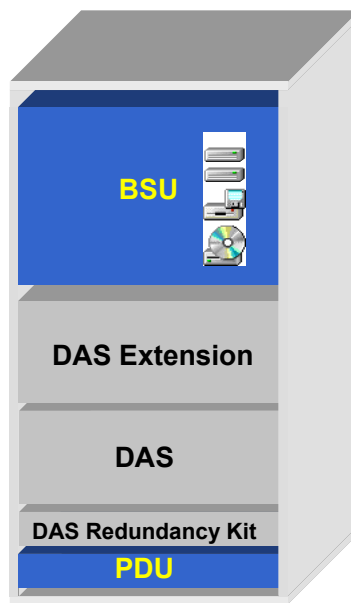


Figure 1-2. DPS 7000/XTA cabinet

External peripheral subsystems can be connected or reconnected if required:

- Raid disk subsystems:
Connected through a FC adapter (for FDA 7 and CDA 7 reconnection).
- Tapes / Cartridges subsystems.
- Telecommunication processors: Mainway or Mainway LE
- Local Printers: Matrix printers (PRUx00, PRUx01, PRUx02, ...)
These printers are connected in serial mode on an USB port through USB/serial port adapter
- Network printers: Mathilde,
Connected through customer LAN (Ethernet HBA).



DPS 7000/XTA server description

The DPS 7000/XTA server is based on a 32 bits Intel™ Architecture server located in a 19" drawer. It is also called Base System Unit (BSU) and is composed of:

- **Central Processor Unit:** Intel™ Xeon Family: up to 8 processors depending on DPS 7000/XTA model.
- **Main Memory** based on DIMM technology with Error Correcting Code (ECC).
- **Input /Output controllers** based on PCI or PCI-X technology: number of slots depend on model.
These controllers are also known under the generic name Host Bus Adapter (HBA).
- The different types of HBA are:
 - 10/100/1000 Mb/s Ethernet adapter for network connection.
 - SCSI adapter to connect tapes/ cartridges devices or disk subsystems.
 - Fiber Channel (FC) adapter to connect disk or tapes subsystems.
- **Internal devices:** Several Devices are integrated in the server itself:
Internal disks: Two disks:
 - 1 to support Windows™, V7000 software and several platform administration softwares. It is the server system disk (the one which is used when powering on or reboot the server).
 - It also contains the GCOS 7 disk generated in the factory and used to install the platform. It should not be considered as a customer reference disk (the customer reference disk or the second production disk has to be installed on external RAID subsystem).
 - The other one is used as a back-up, it is a copy of the Windows™ system disk described above.

This disks are not used by GCOS 7 and customer programs, files or data bases.

- A VDAT 4 mm device is used to Import or export data on the Windows™ part of DPS 7000/XTA.
This drive cannot be used by GCOS 7.
- A CD-ROM and a floppy disk drives to install new software versions.
- **Power subsystem:**
This server is self-powered and cooled.
The power unit is composed of 3 power modules. If one power module fails, the server is still operational and the failed power module can be exchanged without stopping GCOS 7 and the customer production.
The fans are also redundant and can be exchanged without any customer production impact.



Power Distribution Unit description

The PDU is a static component which allows to connect all cabinet components to a single electrical source, itself connected to the customer provided equipment. Only one power cord is required to connect the cabinet.

The system can be connected to a customer provided external UPS (Uninterruptible Power System).

GCOS 7 will be alerted when the power is delivered only by the UPS, and when the UPS will be discharged.

Optional Disk subsystem description

The DPS 7000/XTA cabinet can house an internal disk subsystem. Each model is always used with RAID1 configuration.

Each model can be accessed with a single path (DI configuration) or a dual access (HA configuration).



Console operation

The DPS 7000/XTA console operation is done through:

- A **local console** connected directly to the DPS 7000/XTA server and composed of a standard 17" screen, an international keyboard (QWERTY) and a PS2 type mouse.
- A **remote console**: this remote console may be :
 - a standard workstation running under Windows™ operating system and connected to the customer LAN to remotely operate GCOS 7 and V7000. Specific V7000 software has to be installed on such workstation. Operator can administrate or operate GCOS 7 and V7000 worlds.
 - a workstation or a laptop connected through the customer LAN or a phone line to operate the whole DPS 7000/XTA server. Terminal Service (Windows™ service) is used as connection software. Operator can administrate or operate GCOS 7 and Windows™ worlds.
- **Local Console printer facility**:

There is no direct console printing facility (as it is on DPS 7000/TA platforms). All the system activity is logged into a file and this file can be easily printed on a local printer connected to an USB port (one is systematically available for this function).
This printer is not a part of DPS 7000/XTA offer and has to be provided and installed by the customer.



1.2.6.2 DPS 7000/XTA specific Initialization operations

As you have seen at the beginning of the §1.2.6, the general DPS 7000/XTA architecture is different from the previous DPS 7000 systems series.

The coexistence of the 2 operating systems WINDOWS™ and GCOS 7 has modified the general initialization principle.

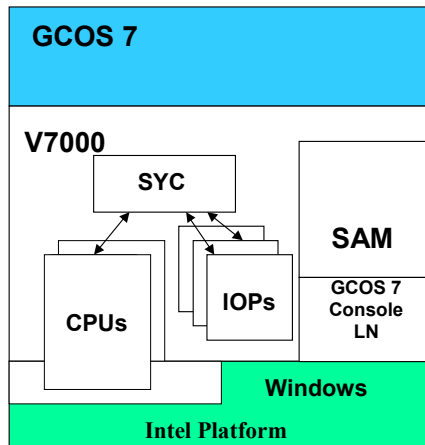


Figure 1-3. V7000 Description

To understand the initialization mechanism, we have to present more in detail the GCOS 7 / V7000 and Windows™ organization:

V7000 is composed of firmware and software parts which allow GCOS 7 to interface with Windows™ and the Intel™ platform.

The **System Control (SYC)** software part initializes, starts and manages the firmware and hardware resources allocated to GCOS 7.

The **System Administration Manager (SAM)** administrates the GCOS 7 configuration, log files, maintenance tools and system utilities.

The **CPU and IOP firmware parts** execute GCOS 7 instruction set and Input / output operations.

The **CPU** part manages the hardware Intel™ processor without using Windows™ services.

The **IOP** part uses Windows™ drivers to access the peripherals.

The **GCOS 7 console** parts give operator access to GCOS 7.

3 GCOS 7 consoles are available (LN02, LN03, LN04 resources).



When the system is powered on or reset the following steps are automatically executed:

- Power on self-tests to ensure that the hardware is ready to work.
- BIOS is started to recognize and configure the hardware components of the server.
- Then the WINDOWS™ operating system is started from the internal disk.
- The SYC service is automatically started.
- If the SYC automatic mode is set then the SYC service starts the ISL function which itself starts GCOS 7 (refer to § 3.3.1).
The GCOS 7 reaches the GCOS 7 READY state without any operator action.
The customer production is started depending on GCOS Start-up procedure.
- If the automatic mode is not set, the operator has to manually launch the GCOS 7 using SYC commands. See following chapters for more details or refer to V7000 Operator's Guide

1.2.6.3 DPS 7000/XTA specific administrator & operators principles

As you have seen in the precedent paragraphs the first operating system to be loaded is Windows™, this peculiarity implies a new dimension in the administrator and operator definition.

Users have to be declared by the Windows™ administrator.

These groups and users will not replace current GCOS 7 access rights (Sysadmin, project, billings, ...). They will be used to manage and protect the Windows™ environment.

V7000 first installation creates automatically 2 user's groups:

- **V7000Operators** which gives control to SYC and normal system operation.
The customer administrator has to create users in this group (people who will have to operate V7000).
Note: Bull users should be created in this group to allow them to initialize, test, use the system as normal operators These users could also authorize Bull people to use the V7000 remotely.
- **V7000BullServices**, this group is reserved for maintenance people and gives control to specific tools (configure V7000, declare GCOS 7 disks, ...)
Note: The customer administrator should not create customer users in this group, untrained persons may induce actions which can damage the customer production.



- Another implicit group is the **Windows™ administrator group**, this group is reserved to the customer people who have the authorization to administrate the whole system.

Note: V7000 and Interoperability software installations need Windows™ administrator access rights. These tasks are normally done by Bull people. Therefore, the customer administrator will have to create (may be temporarily) a user in this group accessible to Bull people.

The GCOS 7 administrator and operator dimensions are not changed (see §1.5).

Refer to V7000 Operator's Guide to have more details.



1.3 The GCOS 7 Operating System

General Comprehensive Operating System (GCOS 7) is the main operating system of the DPS 7000 computers. GCOS 7 is loaded into Main Memory from the System Disk.

The operator communicates with GCOS 7 through an operator console which may be the System console, although not necessarily so.

Once GCOS 7 is loaded, it is in control of the DPS 7000 system. The operator's role is to influence and modify the practical operation of GCOS 7. He can control certain parameters and objects: the behavior of jobs, memory, devices, priorities, queues, classes, and many other elements.

The main tools for the manipulation of these elements are the GCL (GCOS 7 Command Language) operator commands, described in Sections 7 and 8, *Description of the Operator Commands*.



1.4 A GCOS 7 Session

1.4.1 Initial System Load (ISL)

The ISL procedures are important to the system operator for the initial set up of the GCOS 7 system. Initial System Load (ISL) operations:

- complete the loading of the operating system into main memory;
- specify current and following session characteristics.

These operations are selected among the optional replies to the IL01 OPTIONS? question, the initial message to the operator after GCOS 7 is loaded.

The RESTORE, RESTART, and BACKING STORE options are of special interest to the operator. ISL is fully described in Section 3 *Starting and Terminating a GCOS 7 Session*.

Restore

If segments in the SYS.BKST, SYS.TVMF, SYS.PVMF or SYS.LIB files have been corrupted, then these files must be restored from the SYS.SYSTEM file. At the completion of the restore a test is made of the state of all devices. Restore is also mandatory after a SYS.SYSTEM update (CONFIG for example).

Restart

The system segments are correct, so now a particular restart is selected, depending on how the last session was terminated - normally or abnormally. The type of restart affects jobs, output, and characteristics from the previous session.

1. Warm Restarts the session as closely as possible to the last session. Job and output processing is resumed following the same characteristics, e.g. multiprogramming level. If the termination had been abnormal, then some file and job salvaging is required.
2. Cold This restart loses all existing jobs, but not necessarily output, or pre-initialized load modules.
3. Clean A radical restart that loses jobs, pre-initialized load modules, network and session characteristics. However, the operator may request to recover the outputs.



Backing Store (BKST)

Given that the size of main memory is limited, Backing Store files are used to contain permanent or temporary information and temporarily to store executing programs not being processed by the system.

The Backing Store has three main functional elements:

1. System Backing Store

This comprises the static segments of the system contained in either a SYS.BKST or a SYS.BKST0 file.

2. Permanent Backing Store

It contains information kept from one session to another, restarted as COLD or WARM.

It includes pre-initialized load modules, sharable modules, permanent Virtual Memory Files (VMF), and checkpoints.

The permanent backing store is held on the SYS.LIB file and can be extended from SYS.LIB1 to SYS.LIB15.

The permanent VMF backing store is held on the SYS.PVMF file and can be extended from SYS.PVMF1 to SYS.PVMF15.

3. Temporary Backing Store

It contains session information, either non-repeatable user programs, or temporary system segments.

The temporary backing store is held on the SYS.BKST files and can be extended from SYS.BKST1 to SYS.BKST15.

The temporary VMF backing store is held on the SYS.TVMF file and can be extended from SYS.TVMF1 to SYS.TVMF15.

Backing Store Files

There are four types of backing store file:

- SYS.LIB which may exist as up to 16 permanent paging backing store files (named SYS.LIB, SYS.LIB1 through SYS.LIB15) which contain pre-initialized load modules, sharable module libraries, and checkpoints;
- SYS.PVMF which may exist as up to 16 permanent VMF backing store files;
- SYS.TVMF which may exist as up to 16 temporary VMF backing store files;
- SYS.BKST which may exist as up to 16 backing store files containing both permanent and temporary information, such as segments of the system, non-repeatable user programs.



A list of the volumes containing backing store files is retained from session to session; the operator can change this list at ISL time. Other characteristics which can be modified at ISL time include:

- the list of resident volumes,
- the list of volumes shared between two systems.

1.4.2 Job Management

At System Initialization one final aspect of the system has to be selected - job management. This concerns the running of jobs, their introduction, scheduling, execution and output, and also the quantities or "load" of jobs that the system can know and handle at any one time.

A key concept is **multiprogramming**. Given that more than one job can be executing at the same time, the maximum number of jobs executing has to be established. This load includes user and service jobs. It is an operator's responsibility first to select this maximum level and then to monitor it. This is a dynamic task that involves using the GCL commands.

Similar kinds of operations can be established for scheduling jobs, where priority and priority class are used, and for outputting jobs, where levels of output are selected and then monitored, again using the dynamic control provided by the GCL commands.

The **Automatic Resource Management** facility (ARM) optimizes system response time and throughput. It is based on a feedback concept which keeps the utilization of resources (such as the CPU, and the main memory) within specified ranges.

ARM is described in Section 5 *Running User Jobs Under GCOS 7*, and in more detail in the *ARM User's Guide*.



1.5 The Role of the Operator

Refer to §1.2.6.3 to see the specific administrator and operator dimensions on the DPS 7000/XTA platforms.

1.5.1 Main Operator

The various capabilities of a main operator to establish and monitor the system that were introduced in the preceding paragraph, are defined in the project established by the System Administrator. A short list of these includes:

- The exclusive use of GCL commands that modify the functioning of the system (commands like `MODIFY_LOAD` or `START_LOAD`).
- The management of hardware devices: printers, disks, etc.
- Output management.

1.5.2 Other Operators

An operator can also assume other roles. He can become the network operator, with all the network control at his disposal. Or he can become the Master User of a TDS, with all the TDS control commands at his disposal. He can also become a RBF (Remote Batch Facility) station operator, with a special set of control commands.

For a very general introduction to these roles, see Section 1.6 *The Dimensions of GCOS 7*.

For further information on operator commands management, see the *System Administrator's Guide*.



1.5.3 GCOS 7 Control of Certain Operator Tasks

Increasingly, the task of organizing work is being transferred from the operator to the machine.

The trend is to reduce operator intervention, for the following reasons:

1. The operator may not be at the console when a request is issued.
2. The operator may not take the best action.
3. High volume of traffic.
4. Multisite supervision by one operator.

Operating systems often work 24 hours a day. Facilities intended to minimize operator intervention include:

- on-line media,
- remote printers at end-user sites,
- automatic reloading after a crash,
- user-defined applications for submitting jobs,
- programmed operator applications.

The function of operators can be divided into two categories:

1. Local operators manually operating devices such as tape drives, removable media, local printers.
2. Operators responsible for different aspects of the system, for example TDS master console, network control, and system tuning.



1.5.4 Unattended Operation

The automatic features of GCOS 7 include:

- Automatic Volume Recognition (AVR),
- Dynamic file extension,
- Starting jobs at predefined dates or times.
- Automatic initialization,
- Optional system console,
- Predefined replies to questions,
- Remote main operator,
- Programmed operator applications.

Automatic initialization means that the system is initialized without any on-site manual operator intervention. Software loading may be automatically set up at the end of a previous session by use of the REPEAT option. In addition, the restart may take place in AUTO and SILENT or NSILENT mode. The identifier of the device from which the system is to be loaded at the next session may also be given while the system is running. These various possibilities are explained in Section 3.2 *Initial System Load (ISL)* and the IL01 Options.

The Optional System Console means that the system, when initialized, can execute without the need for any operator log-on. The system continues to operate in unattended mode, provided that no operator action such as a recovery or a mount operation is required.

Predefined Replies to Questions means that the System Administrator provides default replies to questions likely to be asked during initialization.

The Remote Main Operator feature enables the main operator to log on through any terminal in the network, either on site or remote, instead of logging onto a local system console.



1.5.5 Operating a Large System

Several main operators may log on simultaneously using the multi-console feature. They may have either the same or different access rights to system commands. The system administrator decides which commands a particular operator may use.

On large systems, an individual operator may be responsible for a particular activity, for example:

- device operations;
- batch operations;
- incidents affecting the system;
- file handling.

With any of the above responsibilities, a network operator becomes one of the main operators.

1.5.6 Distributed Systems on a Network

Here the emphasis is on complementing local subsystems by providing software and operator services which cannot be easily operated on a smaller computer or which must be centralized for other reasons, for example administrative reasons. At the system control center, applications for execution on the satellites can be prepared.

A main operator can control satellite systems using:

- an asynchronous line of the switched network;
- any terminal on the network if a link is made to the satellite;
- a synchronous terminal with the Remote Multiplexed Operator Support (RMOS) facility which lets you simultaneously control applications on several remote sites. For more information on RMOS, see Section 9 *Remote Multiplexed Operator Support (RMOS)*.



1.6 The Dimensions of GCOS 7

The current release of GCOS 7 has two notable tendencies towards:

- **non-batch activities**, that is, increasing interactive and transactional activities.
- the **increasing integration of any one system into a network of systems**, with all of the distributed interaction that networking involves.

1.6.1 The Batch Dimension

Jobs submitted in batches must be monitored in the system. This may require manipulation of disks or tapes, which must be mounted or dismounted as required, and also responses by a main operator to the various questions, requests, errors, and processing levels that batch monitoring involves.

1.6.2 The Interactive and Remote Facility Dimensions

GCOS 7 supports interactive processing. This means that a user interactively accesses given domains, commands, and files, by which he gives work to the system.

One way of looking at Remote Facility job processing is to regard it as interactive, but distributed over a network. Consider, for instance, the activities involved in Distributed Job Processing (DJP), in which:

- jobs are submitted, monitored, and output on any machine in a network;
- files can be transferred between machines in the network;
- operator commands can control a remote site;
- work can be re-routed from one machine to another in the network;
- the station, at which outputs are printed, can be defined and modified.

This concerns a main operator in two ways. First, he can assume the role of an interactive user, or a network and remote facility operator as required, with all their attendant facilities, commands, and responsibilities.

Secondly, as main operator, he must ensure that these facilities are operated correctly. A main operator can monitor the load of jobs submitted and output by interactive users by using the following commands: CJ, MDJ, RJ, HJ, CO, HO, RO, and MDO.



1.6.3 The TDS Dimension

When a system is transaction driven, programs known as Transaction Processing Routines (TPRs) are written by the user for his specific requirements. TPRs are triggered by appropriate data being introduced into this system.

An operator, when required to assume the role of Master User of a TDS, has sole access to a set of privileged commands. These can start and stop the TDS, admit or remove users, open or close files, lock transactions and so on.

A TDS job must not be submitted in the system start-up (see Section 3.3.6 *System Startup*) as this causes unexpected TDS behavior.

For further information, refer to the *TDS Administrator's Guide* and the other manuals in the TDS documentation set (see list in the Preface).





2. Starting Up Your System

2.1 Introduction

The exact startup procedure depends on the type of DPS 7000 system.

DPS 7000/A DPS 7000/A power-on and initialization operations are described in Section 2.2 below.

If you need to modify the system parameters by running an Initial Storage Load (ISL) and answering the IL01 OPTIONS question, refer to Chapter 3 *Starting and Terminating a GCOS 7 Session*.

DPS 7000/2x0 and /3x0 DPS 7000/2x0 and /3x0 operations are described in Section 2.3 below.

If you need to modify the system parameters by running an Initial Storage Load (ISL) and answering the IL01 OPTIONS question, refer to Chapter 3 *Starting and Terminating a GCOS 7 Session*.

DPS 7000/4xx/Cxx/MTx1 DPS 7000/4xx/Cxx operations are described in Section 2.4. A diagram giving specific components of the DPS 7000/4xx/Cxx is in Section 2.4.6. Termination of the GCOS 7 session is described in Chapter 3 *Starting and Terminating a GCOS 7 Session*.

DPS 7000/5x0/7xx/8x0/Dx0/Mx0/MTxx (≥ 42)/TAxxx/TAxxxC DPS 7000/5x0/7xx/8x0/Dx0/Mx0 operations are described in Section 2.4. A diagram giving specific components of the DPS 7000/5x0/7xx/8x0/Dx0/Mx0 is in Section 2.4.7. Termination of the GCOS 7 session is described in Chapter 3 *Starting and Terminating a GCOS 7 Session*.



DPS 7000/XTA

DPS 7000/XTA operations are described in Section 2.5. A diagram giving specific components of the DPS 7000/XTA is in Section 2.5.7. Termination of the GCOS 7 session is described in Chapter 3 *Starting and Terminating a GCOS 7 Session*.

2.2 The DPS 7000/A

The control units of the DPS 7000/A are shown in Figure 2-1.

Three things to remember about the DPS 7000/A:

1. The DPS 7000/A can be switched on or off by operator command

Both the Local Console and the SOP panel (see Figure 2-1) offer POWER ON, POWER OFF, or SYSTEM RESET commands. These are operational only if the AC power switch is physically on.

2. The normal mode for powering on is Automatic mode

When the system is in Automatic mode, hardware is initialized and the operating system loaded without any intervention from the operator. The complete procedure is described in the following pages.

Of course, it is also possible to power on or initialize in manual mode, but this goes beyond the scope of this manual. For documentation relative to these procedures, refer to the *DPS 7000/A Operator's Guide*.

3. The system must never be powered off without being terminated first

System Termination is described in Section 3. Note that it is possible to leave the system powered on 24 hours a day.

2.2.1 Using the DPS 7000/A Console

A complete explanation of the console is found in the *Operator Reference Manual*.

Below, there is a brief presentation of the console interface, and a set of procedures for doing the following:

- To power on or off the DPS 7000/A from the console without using a hardware switch.
- To send a reset command to the DPS 7000/A.
- To modify parameters such as automatic initialization or the disk from which the system is loaded.

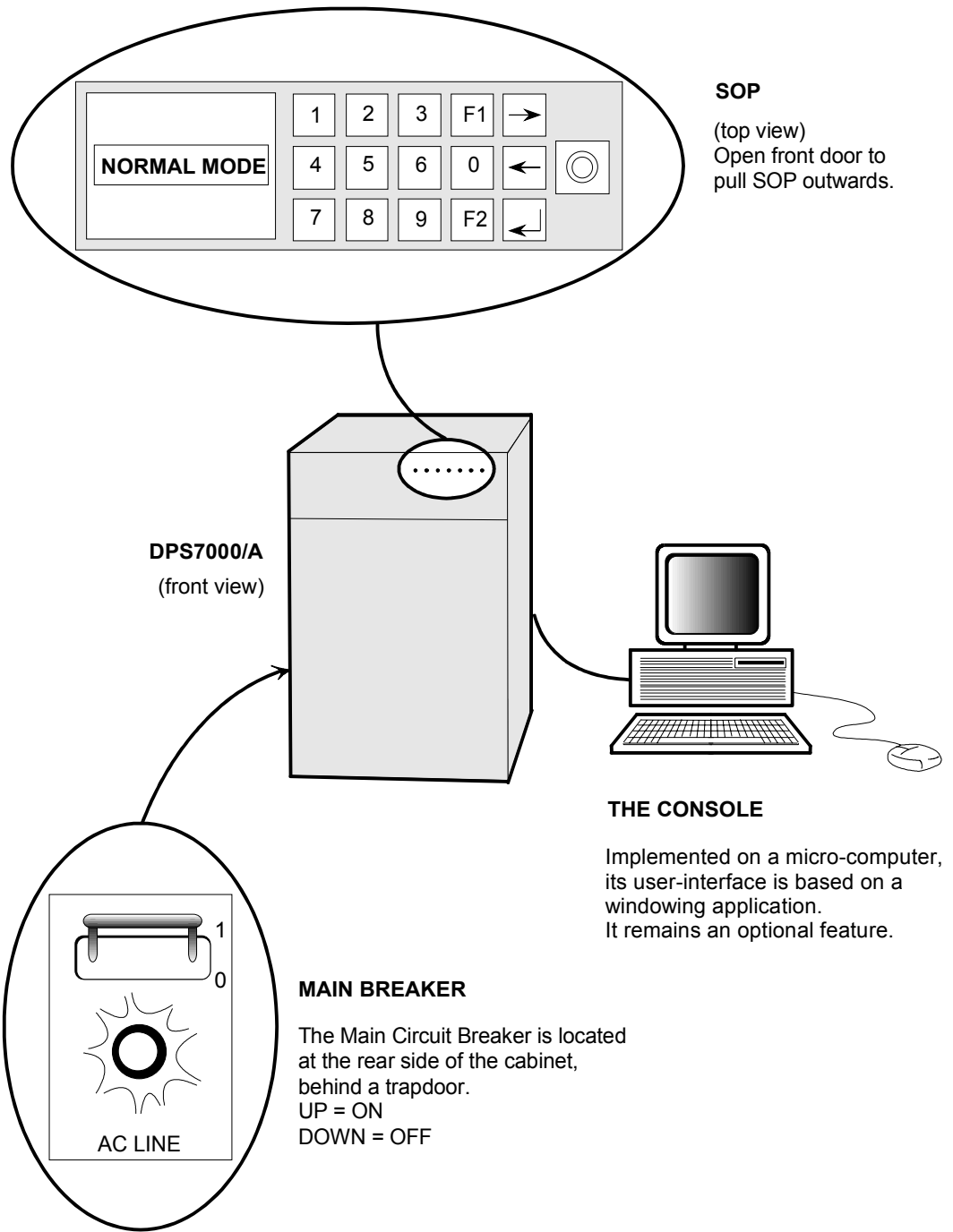
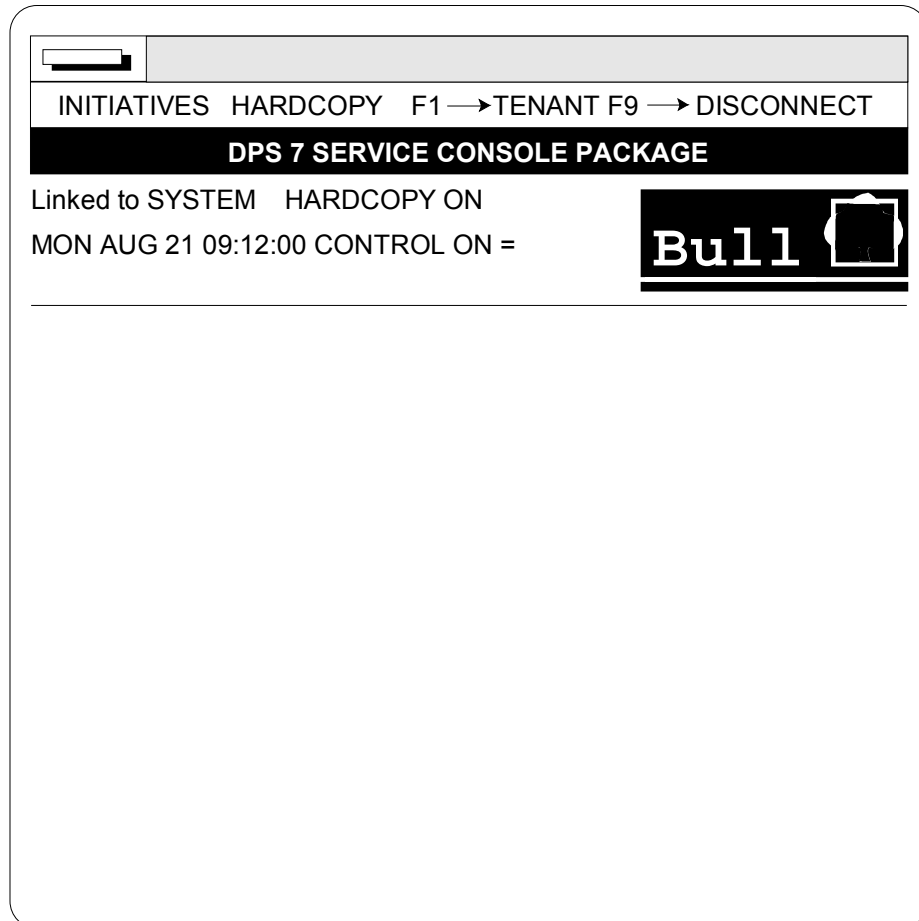


Figure 2-1. The DPS 7000/A Control Units



2.2.2 Introducing the DPS 7000/A Console

The first screen visible (the Control Screen) on the DPS 7000/A console is shown below. If you are at the console, and other screens are displayed, it is always possible to return to this one by clicking on the CTRL icon.



To enter a command from the Initiatives menu:

1. Press <F2> function key to obtain the CONTROL window.
2. Click on the INITIATIVES menu.
3. Click on the required command.

Initiatives commands include Power On, Power-off, System Reset, etc. They are described in detail in the *DPS 7000/A Operator's Guide*.



2.2.3 Powering On the DPS 7000/A

The normal operating mode of the DPS 7000/A is Automatic. Manual initialization is not used on the DPS 7000/A when in normal operation.

1. Ensure that the console is powered on. If not, flip the console switch.
2. Ensure that the AC power switch is ON. (See Figure 2-1. The AC power switch is located at the rear of the system. It should be up, with AC lamp lit.)
3. Activate the power-on command on the console (see procedure given further on).
4. If you are in Automatic mode, Power On and Initialization occur without any further intervention (see below if that is not the case).
5. Once the power-on and initialization phases are over, wait for the following message:

```
hh.mm GCOS 7 RELEASE V6xx xxxxx READY mmm dd, yyyy
```

And if that does not work...

Last message displayed on the console: What to do:

I-MID xx READY	You are in MANUAL mode. In this case, 1. Set automatic mode (See procedure below). 2. Enter a SYSTEM RESET command (See procedure below.).
IL01 OPTIONS? I:	Refer to Chapter 3 <i>Starting and Terminating a GCOS 7 Session</i> to answer the IL01 OPTIONS question. (or Press <Enter> key to load with default options).
*M INIT69 PR01 RECOVER.REPLY (UD, CR)	You have the choice between: - taking care of the printer PR01 (mounting paper, for instance) and answering UD (Use Device), - or doing nothing, answering CR (Cancel Request), and doing without the printer. Refer to Chapter 6 <i>Peripheral Operations</i> for indications relative to printers.



2.2.4 Setting the DPS 7000/A to Automatic Mode

At the console:

1. CONTROL window Press the F2 function key to obtain the CONTROL window.
2. INITIATIVES menu Click on the INITIATIVES menu, or press Alt I.
3. MODIFY INIT PARAMETERS command
Click on the MODIFY INIT PARAMETERS command. A dialog box appears.
4. Enter <1> Press <1> after the blinking cursor, to select Automatic mode, then press <Enter>.
5. Press <Enter> several times
Press <Enter> several times, to accept the values of the other parameters.

2.2.5 Power On or System Reset Command from the Console

To enter POWER-ON or RESET commands from the console:

1. CTRL window Press F2 to obtain the CONTROL window.
2. INITIATIVES menu Click on the INITIATIVES menu (or press Alt I).
3. POWER ON or SYSTEM RESET commands
Click on POWER ON or SYSTEM RESET commands of the initiatives menu.

In the case of a SYSTEM RESET, a dialog box appears, with two options: YES and NO. Click on YES (or enter Y) to validate the command.

The POWER ON command has no effect at all if the system is already powered on.

If a COMMAND COMPLETED message appears, click on OK (or press <Enter>) to acknowledge it.



2.2.6 Service Commands: Reload (RL) and Restart (SR) on DPS 7000/A

2.2.6.1 Reload Procedure

This command is used after a TSYSGCOS (TERMINATE_SYSTEM) command or after a System Crash.

NOTE:

The RL command is only necessary if the REPEAT parameter is not in force; see Section 3.3.8.4 *Automatic Restart*.

To reload the system on a DPS 7000/A

1. Obtain Service Console Call the INITIATIVES menu, then click the initiative, SWITCH TO SERVICE. (<Ctrl C> on the keyboard also works.)
2. <Insert> Press <Insert> to visualize the SP: prompt of SPOS operating system.
3. SP: RL To reload GCOS 7

2.2.6.2 To Restart the System on a DPS 7000/A

This command is used to force a system crash and terminate the system cleanly in the case of a system deadlock, for example.

1. Obtain Service Console Call the INITIATIVES menu, then select the initiative SWITCH TO SERVICE. (<Ctrl-C> on the keyboard also works.)
2. <Insert> Press <Insert> to visualize the SP: prompt.
3. SP: SR To restart GCOS 7.



2.2.7 Powering Off the DPS 7000/A

This procedure works if GCOS 7 is running. (If GCOS 7 is not running, use the POWER-OFF initiative: <F2>, <ALT-I>, <Enter> keys.)

1. Obtain S: prompt in the SYSTEM window
Press F1, then insert. If that does not work, read the appropriate console help text.
2. TSYS GCOS PWROFF This command terminates the system, then powers off.
3. Wait for jobs to terminate
Termination time depends on what jobs are running.
Refer to Section 3.5 *Terminating The Session*.

NOTE:

It may be necessary to terminate the telecommunications job, using the TTSVR TNS command.

2.2.8 Setting Power On/Off Timers (MDPWR and DPWR Commands)

Use MODIFY_POWER_TIMER (MDPWR) and DISPLAY_POWER_TIMER (DPWR) to control the Power ON timer and Power OFF timer. Refer to both commands in Chapter 7 *Description of the Operator Commands* for further information.



2.3 The DPS 7000/2X0 and /3X0

The control elements of the DPS 7000/2x0 and /3x0 are represented in Figures 2-2 and 2-3.

Two things to remember about the DPS 7000/2x0 and /3x0:

1. The normal mode for powering on is Automatic mode

When the system is in Automatic mode, hardware is initialized and the operating system loaded without any intervention from the operator. The complete procedure is described below.

Of course, it is also possible to power on or initialize in manual mode. The procedure is described briefly below on the following pages. For more documentation relative to these procedures, refer to the *Operator Reference Manual*.

2. The system must never be powered off without being terminated first

System Termination is described in Section 3. Note that it is possible to leave the system powered on 24 hours a day.

2.3.1 Powering On the DPS 7000/2x0 or /3x0

- | | |
|--|---|
| On the Main Distributor Board | 1. Ensure that the power supply from the Main Distributor Board is on. |
| On the System Operator Panel (SOP) | 2. Check that the AC LINE indicators are ON, indicating that the system power is ON.
3. Check that the SOP Switch is OFF (Figure 2-3).
4. Set the Diskette unit switches to ON. |
| On the control panels of the peripheral controllers | 5. Check that the Mains Circuit Breaker(s) are set to ON.
6. Set the LOCAL/SYSTEM switches to SYSTEM, or set the ON LINE / OFF LINE switches to ON LINE. |
| On the console(s) | 7. Set the ON/OFF switch to ON. |
| On the SOP | 8. Set the Mains Circuit Breaker(s) to ON. (On a DPS 7000/3x0, set both Mains Breakers to ON.)
9. Set the SOP switch to ON. |



The DPS 7000/2x0 or 3x0 then outputs several pages of messages, without any operator intervention required. Finally, the following message appears:

```
hh.mm GCOS 7 RELEASE V6xx xxxxx READY mmm.dd,yyyy
*****
```

You can now press the <Transmit> key to log on, entering your user-name and password.

And if that does not work...

Last message displayed on the console:

What to do:

```
I-MID xx READY
```

You are in MANUAL mode. In this case, key in:

```
##MP AUTO 1 (to set automatic mode)
##FS INIT 1 (to reinitialize)
```

```
IL01 OPTIONS?
```

```
I :
```

Refer to Chapter 3 *Starting and Terminating a GCOS 7 Session* to answer the IL01 OPTIONS question. (Press <Transmit> key to load with default options).

```
*M INIT69 PR01
```

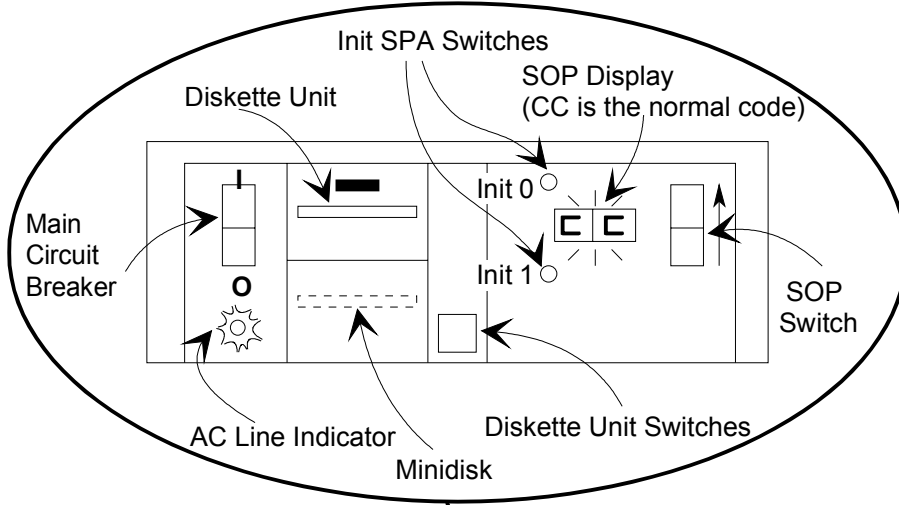
```
RECOVER.REPLY
```

```
(UD, CR)
```

You have the choice between seeing to the printer (mounting paper, for instance) and answering UD (Use Device), or else doing nothing and answering CR (Cancel Request).



The SOP Panel



Warning: Keep the cover closed to avoid accidents.

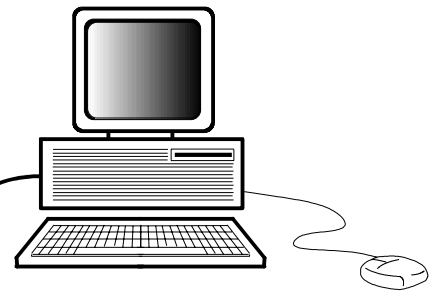
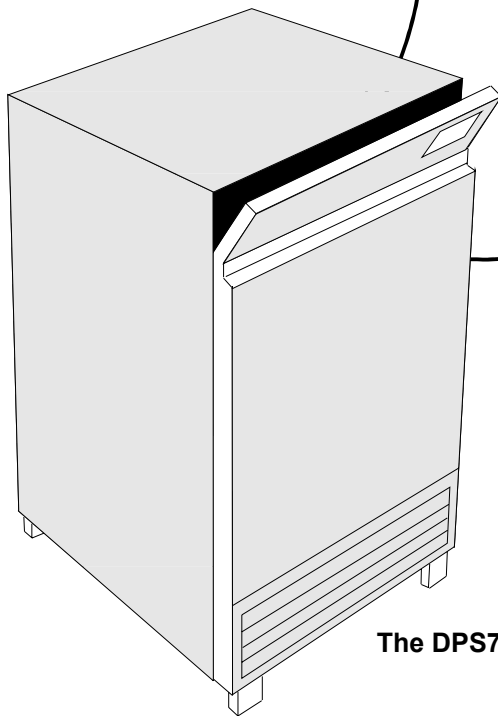
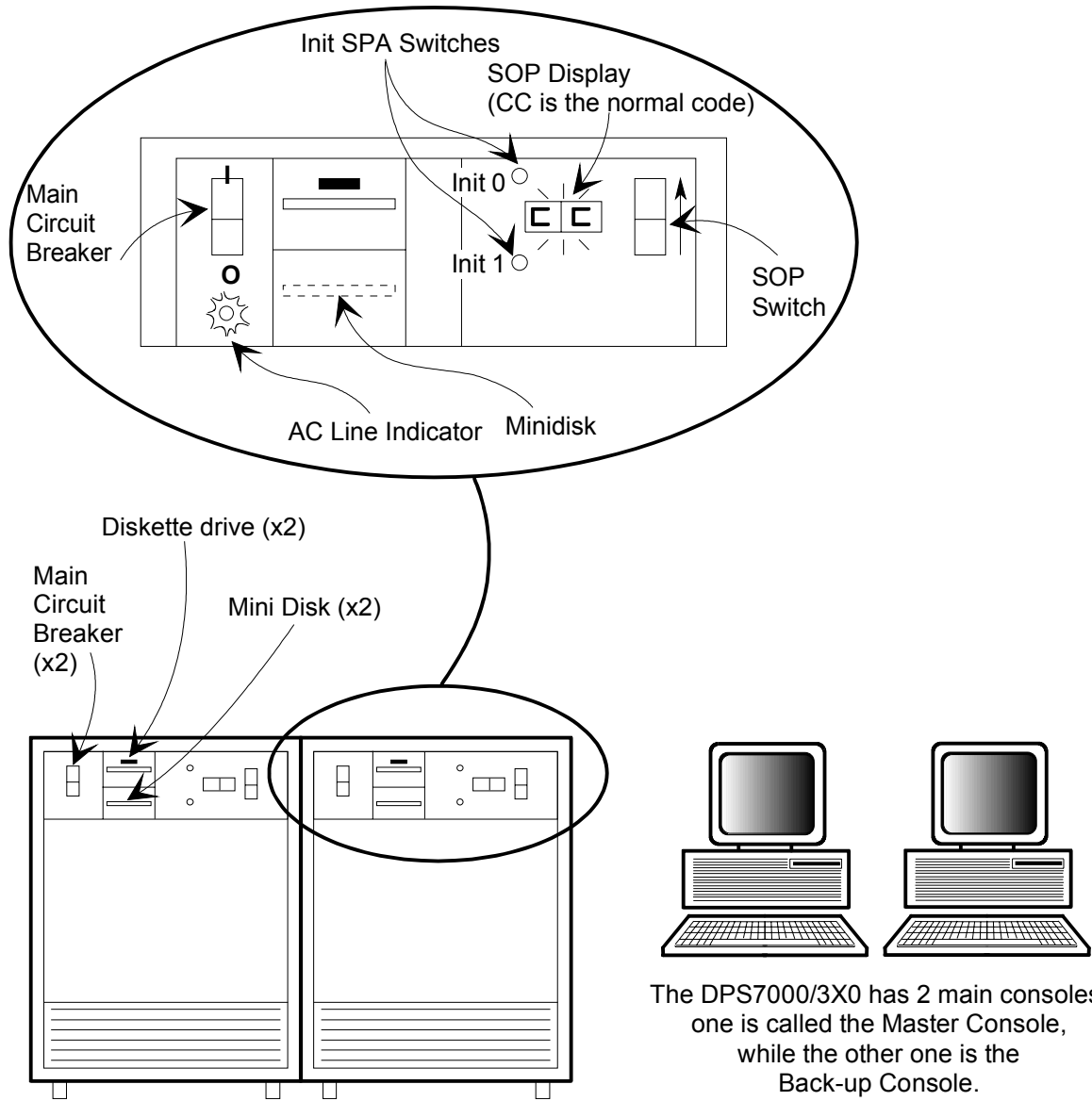


Figure 2-2 The DPS 7000/2X0 Control Elements



The DPS7000/3X0

Figure 2-3 The DPS 7000/3X0 Control Elements



2.3.2 Manual Initialization on the DPS 7000/2X0 and /3X0

This is not the normal initialization procedure.

This procedure is not necessary in the context of normal operation. But it may be of some use in the case of looping crashes, when the system enters a crash, reloads automatically, crashes again, in such a way that it is not possible to regain control.

Commands keyed in by the Operator are given in **bold type**. Note that ## commands can be entered at any time, even if at the same time, the console is outputting a flood of messages.

Procedure	Comments
##MP AUTO 0 ##COMMAND COMPLETED	To set Manual mode
##FS INIT 1 ##COMMAND COMPLETED	To initialize
I-MID xx READY <Break> I:LOAD0	To load SPI from the Minidisk (LOAD2 to load from diskettes)
...	
>>>SPI01 DEVICE LOADER (0/2): 0	To load SPOS from the Minidisk (<2> to load from diskettes)
...	
M SP0100 NO JOBS RUNNING IN SERVICE PROCESSOR ##FM ##MID OFF CALL SP:IN	SP: IN MS03 if you want to load GCOS 7 from MS03 disk
* IL01 OPTIONS ? I:	When you have reached this level, you can modify the IL01 OPTIONS, as indicated in section 3. Otherwise, you can enter <Transmit> to load with default options.
##MP AUTO 1	When you have finished, it is necessary to set the system back to automatic mode.



Table 2-1 Functions of the DPS 7000/2X0 and 3X0 System Operator Panel

Name	Function
MAIN CIRCUIT BREAKER	When this switch is ON (set at 1), the Central Subsystem Units are powered on.
AC LINE INDICATOR DPS 7000/X0 and 3x0	When LIT, it indicates that system power supply is ON at the main distributor board.
INIT SPA PUSHBUTTON	<p>The INIT 0 switch initializes SPA 0.</p> <p>The INIT 1 switch initializes SPA 1.</p> <p>This can also be done by typing in a command at the console.</p>
SOP DISPLAY	<p>Indicates the current status of the SOP by means of a two-digit display.</p> <p><i>See Operator Reference Manual, Section 3, for details.</i></p>
SOP SWITCH	When this switch is ON (set at 1), it activates the Central Subsystem(s). It must not be switched on before the Main Circuit Breaker.
DISKETTE DRIVE	The Diskette drive is used if diskettes are to be used for manual utilization.
DK UNIT SWITCH	When this Switch is ON (set at 1), it activates the mini-disk normally used for initialization.



2.3.3 ASPA Configuration Parameters on the DPS 7000/2X0 and 3X0

The ASPA (Ante Service Processor), which is activated when a DPS 7000/2X0/3X0 system is powered on, has a certain number of configuration parameters. Some of these parameters directly affect initialization mode and bootload devices. The values of these parameters are normally defined at installation time.

The ASPA Configuration screen is illustrated in Figure 2-4.

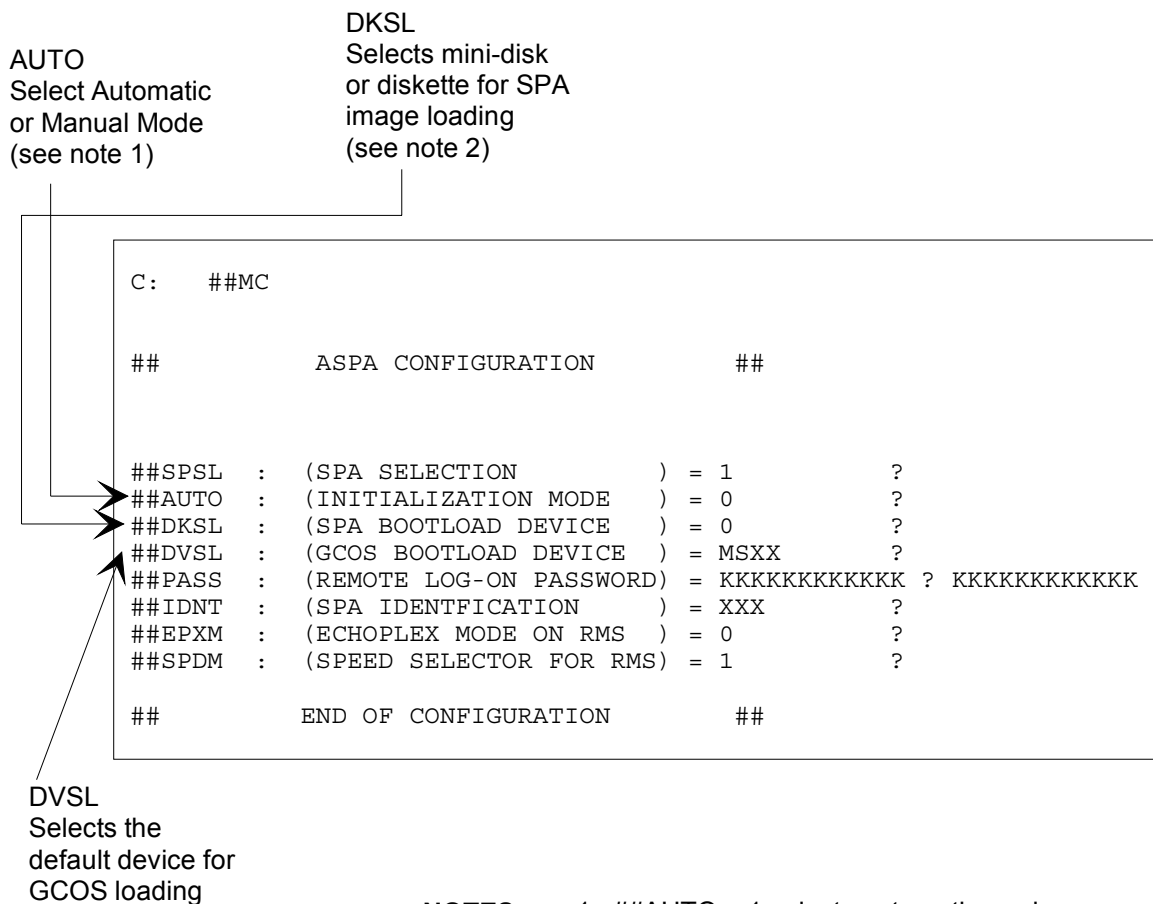


Figure 2-4 ASPA Configuration Screen



1. To display this screen and check the current values of the parameters:
Type in ##MC
2. If it is necessary to modify a parameter:
 - Press <Transmit> to move to next line of screen.
 - Type in a new value.
 - Press <Transmit> to confirm.For example, to select MS03 as the default GCOS 7 bootload device:
Press <Transmit> 3 times
Type in MS03
 - Press <Transmit> to confirm.

2.3.4 Service Commands: Reload (RL) and Restart (SR) DPS 7000/2X0 and /3X0 Systems

The two commands described below are only meaningful when entered after an SP: prompt, with the console in SERVICE mode.

2.3.4.1 Reload procedure (RL)

This command is used after a TSYS GCOS (TERMINATE_SYSTEM) command or after a system crash.

NOTE:

The RL command is necessary only if the REPEAT option is not in force, see Section 3.3.8.4 *Automatic Restart*.

To reload the system on the DPS 7000/2X0 or /3X0

To reload GCOS 7 (after a TSYS 1 command, or after a crash), proceed as follows:

1. ##AC Enter ##AC to switch from SYSTEM to SERVICE.
2. <Break> Press <Break> to display the SP: prompt of SPOS, the service operating system.
3. SP: **RL** To reload GCOS 7.



2.3.4.2 System Restart Procedure (SR)

This command is used to force a GCOS 7 crash and to terminate the system cleanly, in the case of a system deadlock for example.

To restart the system on the DPS 7000/2X0 and 3X0:

1. **##AC**

Enter **##AC** to switch from SYSTEM to SERVICE.

2. **<Break>**

Press **<Break>** to visualize the SP: prompt of SPOS, the service operating system.

3. SP: **SR**

To force a GCOS 7 crash.



2.4 The DPS 7000/4xx/Cxx, DPS 7000/5x0/7xx/8x0/Dx0/Mx0 and DPS 7000/MTxx/Taxxx/TAxxC

There are three processor families:

- the DPS 7000/4x0/4x5/Cxx/MTx1 and the DPS 7000/5x0/7xx,
- the DPS 7000/8x0/Dx0/Mx0,
- the DPS 7000/MTxx/TAxxx/TAxxC.

The GCOS 7 operating system functions in a very similar manner on all these platforms.

The main differences between the larger and smaller systems are in the peripherals:

- the larger DPS 7000/5x0/7xx/8x0/Dx0/Mx0/MTxx/TAxxx/TAxxC systems can include a vast number of external tapes, cartridges, disks and printers,
- the smaller DPS 7000/4xx/Cxx/MTx1 systems have the miniaturized internal peripherals as well as a more limited number of external peripherals.

2.4.1 Turning On and Off The System

In normal mode the system is powered on by first turning on the power switches of each cabinet and of the Service Processor console. Then, the operator simply pushes the "on" button on the right of the System Operator Panel (SOP) of the BASIC/CSS cabinet. The SOP is the little pull-out panel, behind the front door, at the top of each cabinet.

If the Programmed Operator Facility (POP) for an automatic initialization has been loaded in your service processor, the cabinets are powered on, the hardware is initialized and the operating system is loaded without any intervention by the operator.

Note that it is possible to leave the system powered on 24 hours a day.

The complete procedure, in normal mode, is presented in the flowchart below:

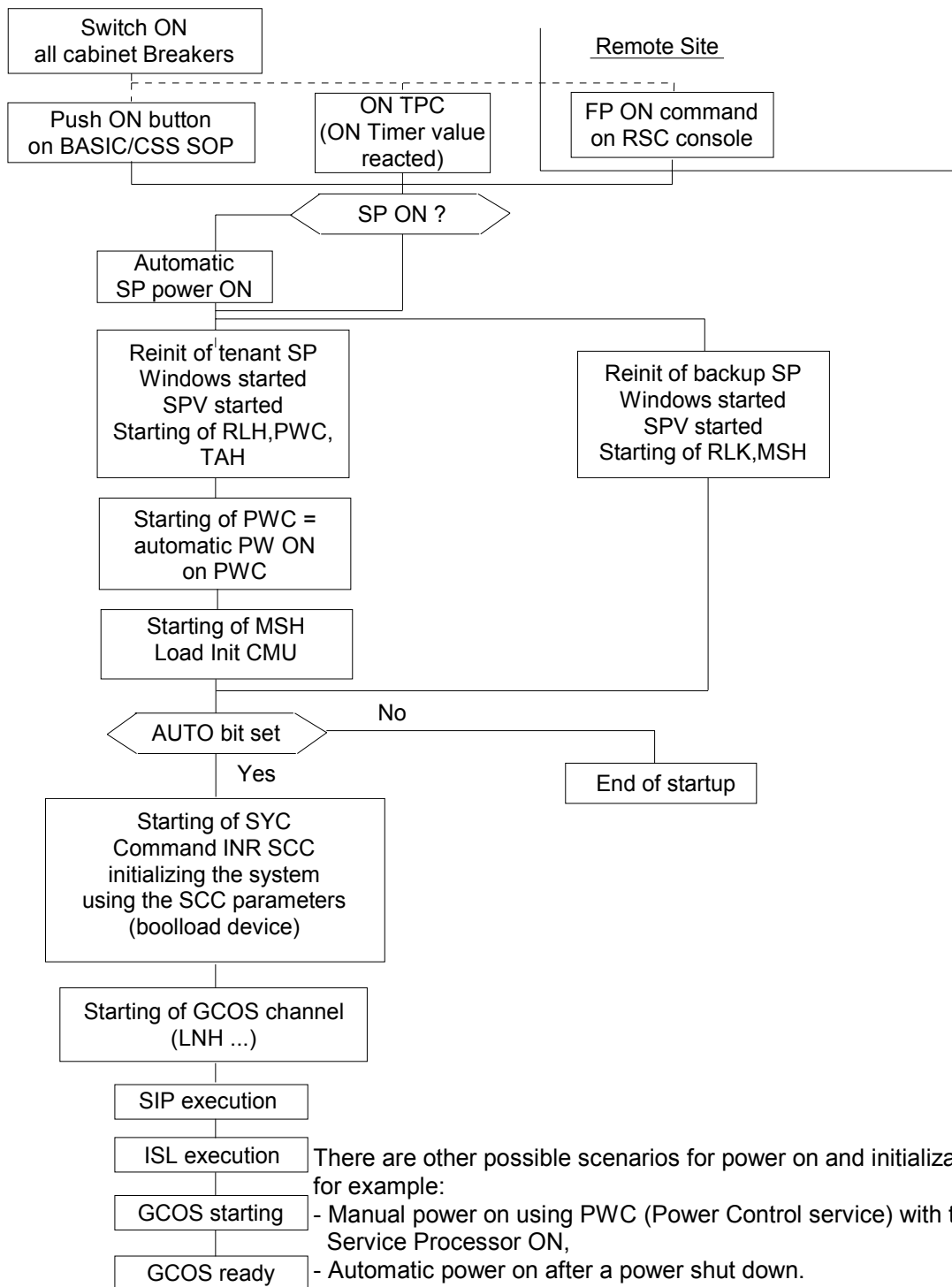


Figure 2-5 DPS 7000/4xx/Cxx, DPS 7000/5x0/7xx/8x0/Dx0/Mx0 and DPS 7000/MTxx/TAxXX/TAxXXC Power On Procedure



2.4.2 Using the Consoles

Information on the Consoles:

The system is delivered with a basic configuration of 4 consoles. The configuration comprises:

1. The Tenant Console of the four consoles available per system, only the tenant console is started automatically at initialization.
2. The Backup console (optional).
The two consoles above are microcomputers that contain the service processor. They are on site.
3. The remote service console (RSC),
4. The Remote Workstation is devoted to maintenance tasks. The operator cannot use this console, but he or she controls it from the Tenant Console.

Information on Services:

The consoles run applications called services.

A service allows you to operate the service processor in window mode or in line mode.

Each service has an appropriate function. For example, SYC and CFG services handle respectively Initialization and Configuration.

The Screen Presentation:

The first screen visible on the tenant console is shown in Figure 2-6.

RLN (Remote Line Handler) service starts automatically at power on together with SPV (SuPerVisor), PWC (PoWer Control) and MSH (Maintenance Station Handler) services.

For more information about the services, refer to the appropriate *Operator's Guide*.

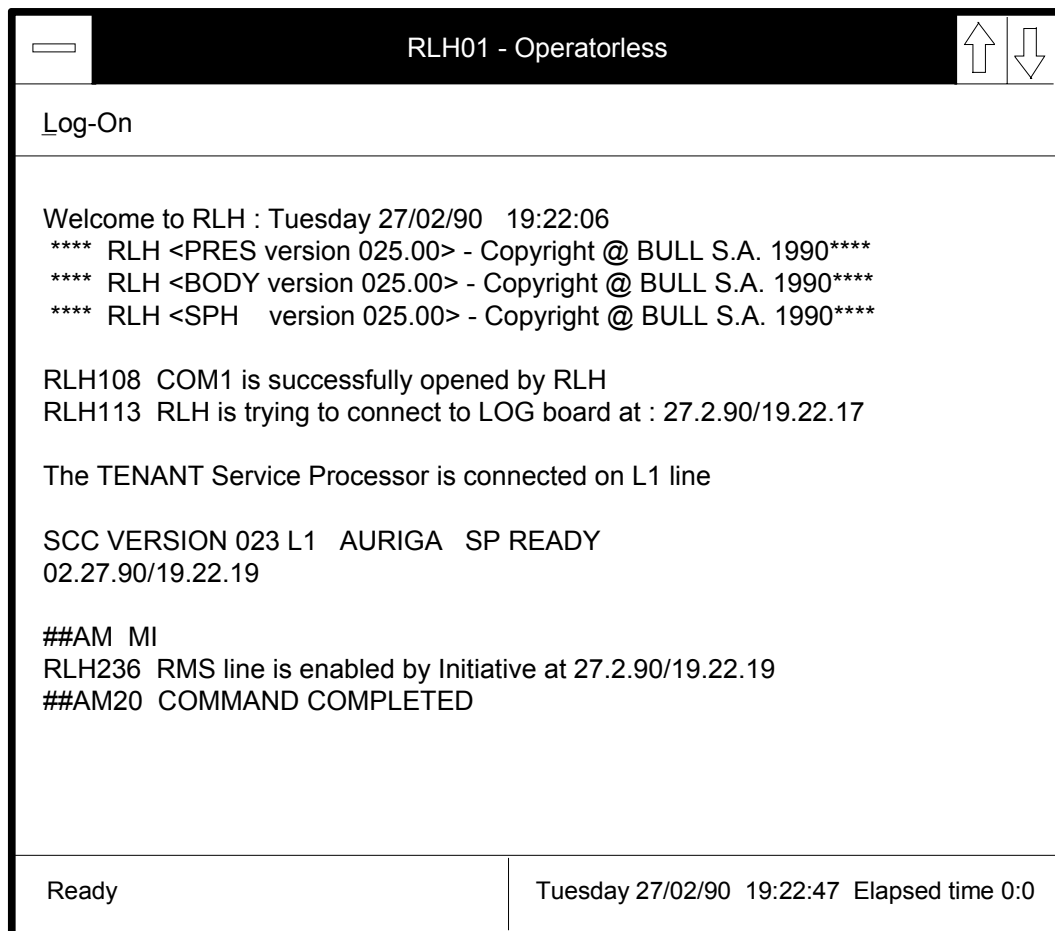


Figure 2-6 Screen Presentation



2.4.3 Service Commands: Reload (RL), Restart (SR)

Reload Procedure (RL)

This command is used after TSYS GCOS (TERMINATE_SYSTEM) or a System Crash.

To reload the system:

1. On the tenant console, call the SYC service by clicking on its icon in window mode, or type, EJR SYC in line mode
2. At the SYC prompt (SYC:), to restart GCOS 7, type this command:

```
RL [S0 | S1]
```

S0 is for system zero and S1 for system one when a bi-system is involved.

Restart Procedure (SR)

This command is used to force a system crash and terminate the system cleanly, in the case of a system deadlock, for example.

1. On the tenant console, call the SYC service by clicking on its icon in window mode, or type: EJR SYC in line mode.
2. At the SYC prompt (SYC:), to restart GCOS 7, type this command:

```
SR [S0 | S1]
```

S0 is for system zero and S1 for system one when a bi-system is involved.

2.4.4 Setting the Power On/Off Timers (MDPWR & DPWR Commands)

Use MODIFY_POWER_TIMER (MDPWR) and DISPLAY_POWER_TIMER (DPWR) to control the Power On Timer. Refer to both commands in Chapter 7 *Description of the Operator Commands* for more information.



2.4.5 Powering Off the System

From the Tenant Console

On the Tenant Console, under the PWC service, enter the command:

PW OF (include the space)

This command shuts down everything except the Tenant Console itself.

To turn off the Tenant Console, enter this command:

SPOF (no space)



IMPORTANT:

Do not enter SPOF from a remote console. With the Tenant off, there would be no way to reconnect without manual intervention at the computer site itself, which could be far away!

From the SOP

On the SOP of the BASIC/CSS cabinet:

- Verify that you are in normal run mode.
- With the keys, select either the command:

NORMAL OFF This sends GCOS 7 a request to switch OFF; the system will switch off only when authorization returns from GCOS 7.

or

NORMAL EMER OFF This switches off the cabinet, but without requesting authorization from GCOS 7. It has the same effect as the POWER OFF initiative in the PWC service window.



2.4.6 The DPS 7000/4xx/Cxx/MTx1

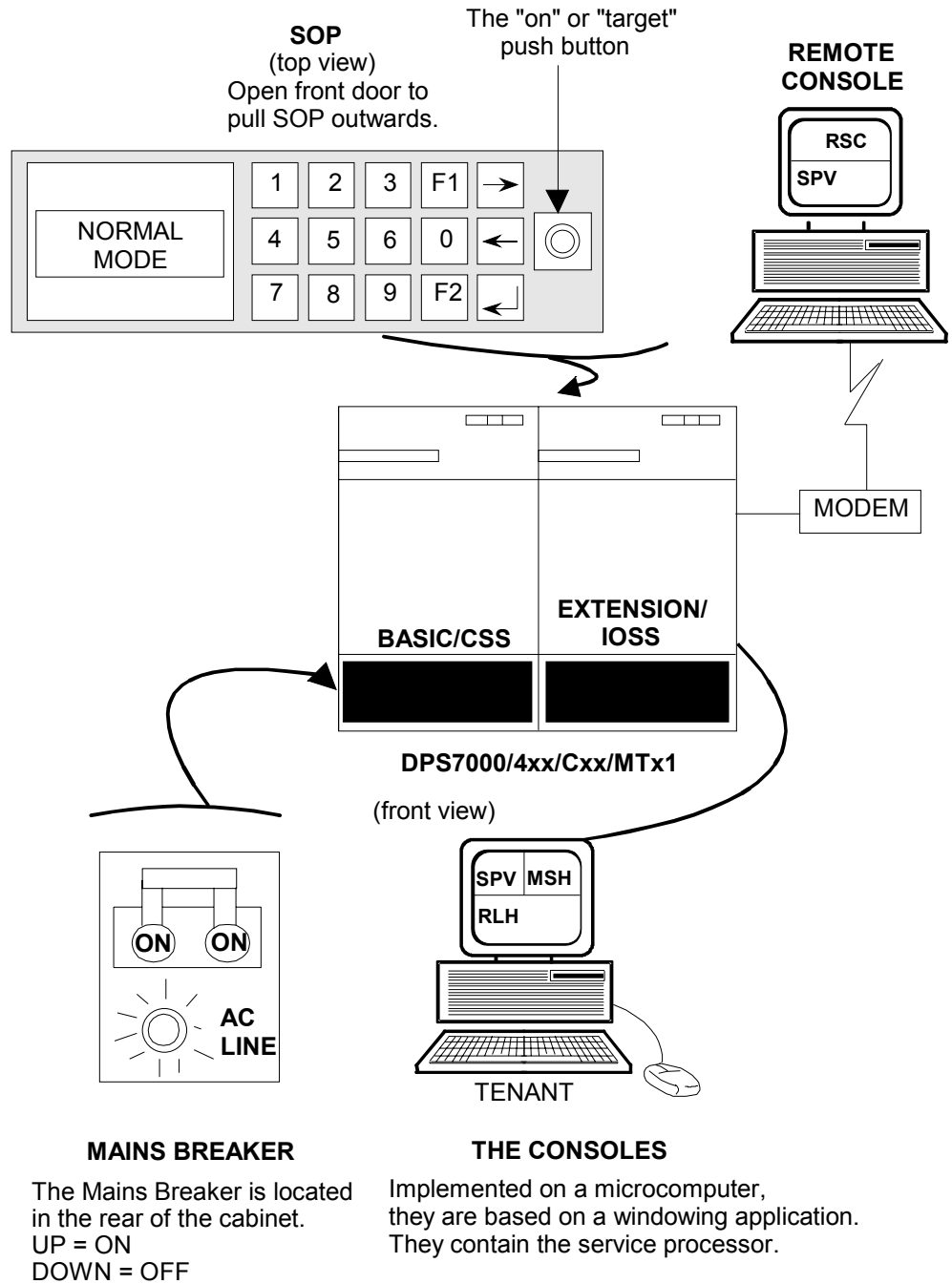


Figure 2-7 The DPS 7000/4xx/Cxx/MTx1 Control Units



2.4.7 The DPS 7000/5x0/7xx

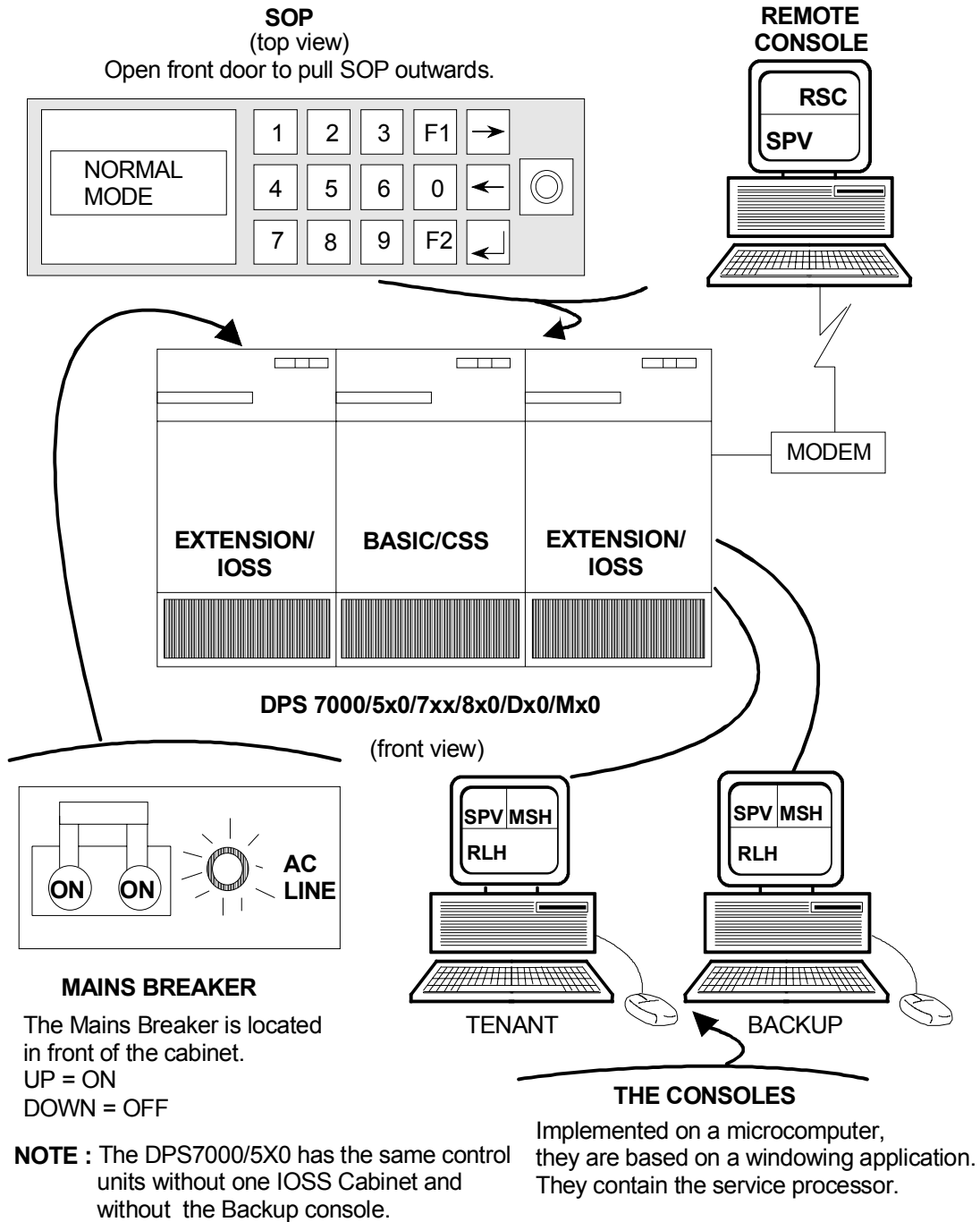


Figure 2-8 The DPS 7000/5x0/7xx Control Units



2.4.8 The DPS 7000/8x0/Dx0/Mx0/MT42 and UP/TAxXX/TAxXXC

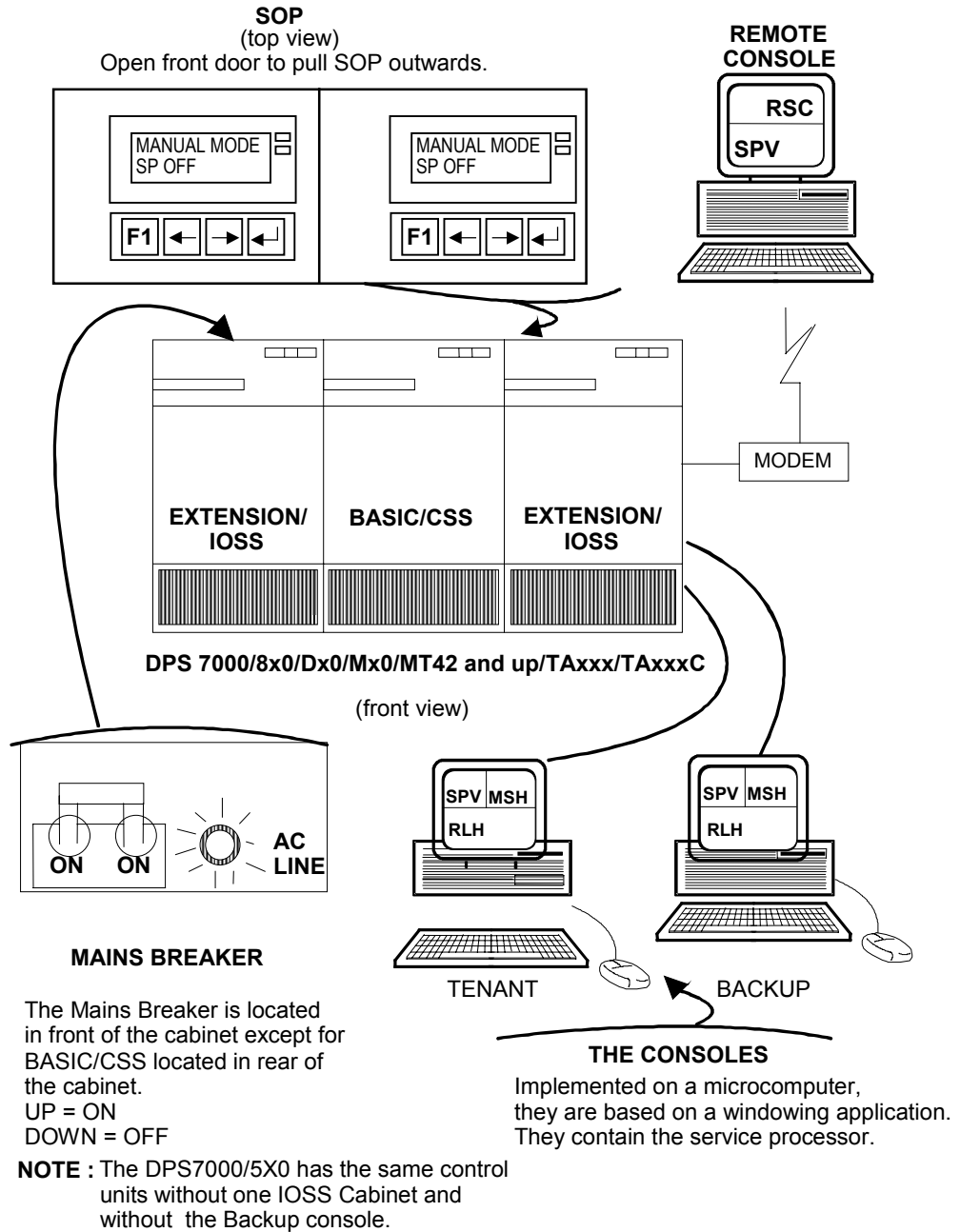


Figure 2-9 The DPS 7000/8x0/Dx0/Mx0/MTxx/MT42 and up/TAxXX/TAxXXC Control Units



2.5 The DPS 7000/XTA

The control unit of the DPS 7000/XTA is shown in §2.5.1

1. **The DPS 7000/XTA system can be switched off by operator command** when terminating the Windows™ session.
Note: before terminating the Windows™ session be sure that GCOS 7 and V7000 software are not running.
See later on in this chapter (§2.5.3) the normal procedure to power off the system.
2. **The Normal mode for powering on is the automatic mode**
When the system is in automatic mode, hardware is initialized and the operating systems (Windows™ first then GCOS 7) are loaded without any intervention from the operator, the complete procedure is described in the following pages.
Of course it is also possible to power on and/or initialize in manual mode: refer to the V7000 operator's guide.
3. **The system must never be powered off without being terminated first**
System termination is described in section 3 (note that it is possible to leave the system powered on 24 hours a day).



2.5.1 Powering On the DPS 7000/XTA

There is no main breaker in the cabinet. The power on/off operations can be done:

- Using the control panel of the DPS 7000/XTA server located on the front side of the server drawer. It is accessible by opening the front door of the cabinet.

The figure below describes switches and lights.

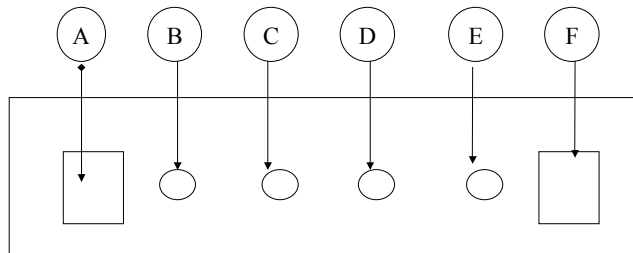


Figure 2-10 DPS 7000/XTA Control Panel

A: Power switch: press to turn the system On or Off

B: Power LED: when green, power is on.

C: Status LED: when green, system is OK.

D: Disks LED: When green, Internal disks are being accessed.)

E: Sleep LED: when lit, the system is in the sleep (power saving) mode.

F: Sleep switch: Press to put the system in the sleep mode (operating system has to support this option).

- Using a remote connection (through LAN and or COM 2 serial port).
When the server is administrated by the ESMPRO software, it is possible to use the Wake up LAN mechanism to power on the system remotely.
This function is accessible to the customer on LAN and to the maintenance people using COM2 serial port.

If the local power sequence is used, the local screen has to be also powered on to be able to follow the initialization process.



Table 2-2 LED information

LED	Status	Description	Response
<i>Power</i>	Off	Power OFF	None required (normal)
	Green	Power ON	None required (normal)
	Red	Power alarm	Replace the power supply.
<i>Status</i>	Off	Power OFF	None required (normal)
	Turns off after lighting in amber	Abnormal	Check condition
	Green		
	(Steady light)	No alarms	None required (normal)
	(Flashing light)	Abnormal	Check condition
	Amber	Abnormal	Check condition
<i>Disk</i>	Off	Not accessing disk drives	None required (normal)
	Amber	Internal disk drive failure	Check disk drive status led
	Green	Accessing disk DRIVES	None required (normal)
<i>Sleep</i>	Off	Sleep mode is off	None required (normal)
	Amber (Lights immediately after power-on until POST begins.)	Power-saving mode	None required (normal)



2.5.2 Using the consoles

The system is delivered with a local system console directly connected to the DPS 7000/XTA server, it consists of:

- A 17" screen
- An international keyboard (US – QWERTY)
- A PS2 type mouse.

There is no intermediate workstation (such as service processors on previous DPS 7000 systems).

How to access V7000 & GCOS 7 services

When using the manual mode, the V7000 services are accessible using the standard START menu from Windows™.

The figure below shows this menu.

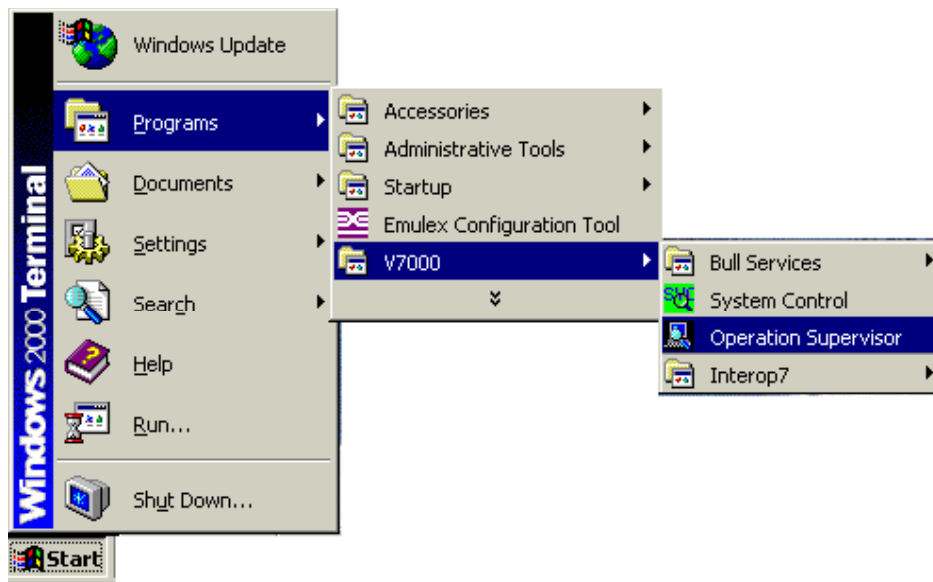


Figure 2-11 DPS 7000/XTA Windows™ Start Menu

The entry point is the V7000 line of the programs menu.



The customer operators may only have access to the following items:

- **System Control** to give access to GCOS 7 world
- **Operation Supervisor** to know what is active, who are the users currently logged, ...
- **Interop7** to administrate interoperability solutions between GCOS 7 and the open world (FTP, ESP7 , DA7,....)

Note: The customer operators and administrator should not execute programs which are in the Bull Services menu, these services are reserved to Maintenance and support people.

The execution of some of those services could have a negative impact on the GCOS 7 production when they are used outside the context for what they have been developed.

SYC operation (overview) The System Control service is used to manage GCOS 7:

- Start, Terminate, Restart, Reload the GCOS 7
- Set operational parameters (Bootload device, Automatic mode, ...)
- Manage the GCOS 7 configuration (Exclusion list, ...)

The paragraph gives an overview, the detailed description is given in the V7000 Operator's Guide.

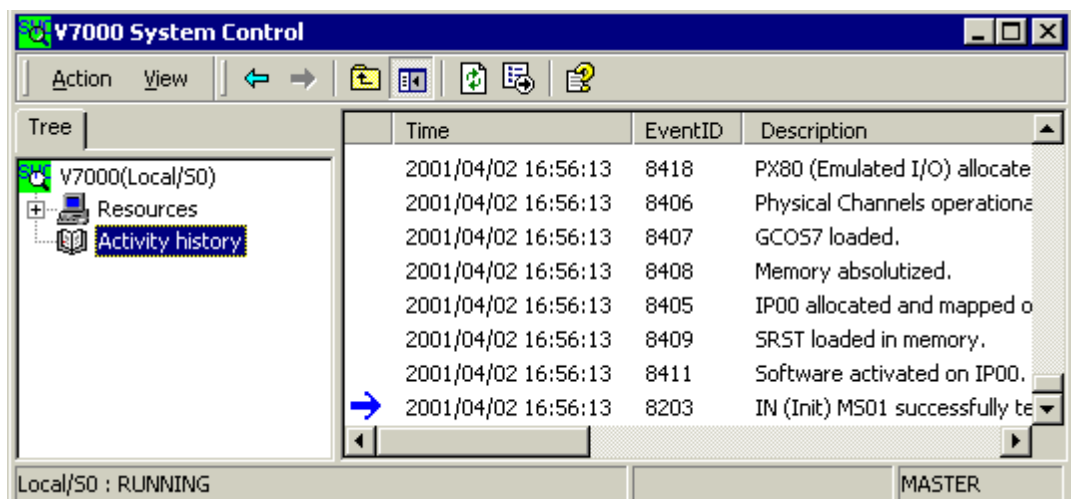


Figure 2-12 V7000 System Control screen image



This is the screen you see when you select the System Control in the Start menu of Windows™.

The **activity history window** lists all the previous actions and messages. The last message is marked with a blue arrow.

How to change SYC parameters The different parameters used by SYC at a given time are accessible through the Properties screen. You have access to these properties by right clicking on the V7000 node (green icon in the above figure).

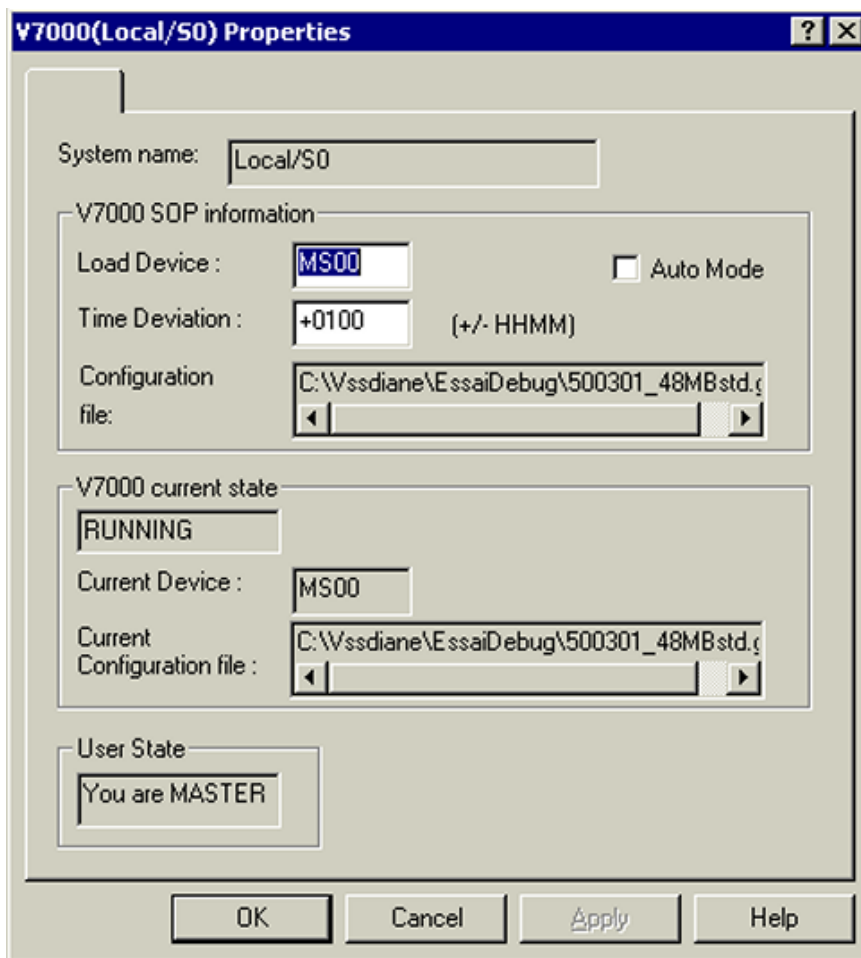


Figure 2-13. V7000 Properties screen image



In the first part of this screen you can select:

- The load device (GCOS 7 system disk): there is no more possibility to have a different bootload and ISL devices.
- The Time deviation.
- The automatic mode.
- The V7000 configuration file to be used at the next GCOS 7 initialization. (this parameter is normally managed by Bull Customer Service people).

You have also an information screen to give you the current state of V7000:

- Is it running or not ?
- If it is running the load device and the configuration which were used at the previous initialization.
- The information concerning your own status regarding SYC control:
MASTER means you have the control of SYC and you can do actions.
SLAVE means that someone else is MASTER and he has the SYC control.
Refer to V7000 operator's guides to get more details.

How to enter SYC actions: The following screen image shows the way you can enter actions. You may proceed in the same way as for accessing the Properties screen, by right clicking the V7000 node.

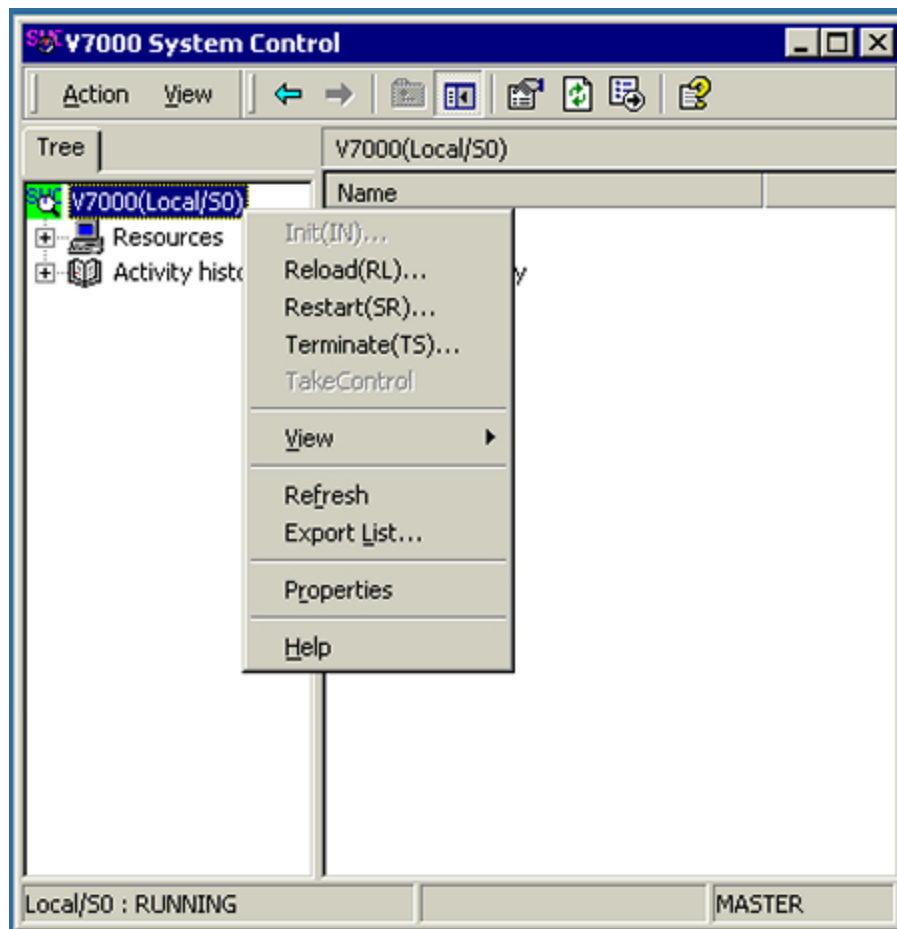


Figure 2-14 SYC Action menu screen image

Instead of selecting Properties, you can choose an action (in the top of the menu). These actions are contextual, which means that only those which are possible are selectable. In the above example the system is running (see status on the last line of the screen): the Init action is Grey. The operator may choose either:

- **Reload command:** it has to be used after a GCOS 7 Terminate System command (TSYS).
- **Restart command:** it has to be used to force a system crash and terminate the system as cleanly as possible, after a system deadlock for example.
- **Terminate command:** it has to be used after a TSYS GCOS 7 command to shut down correctly the GCOS 7 session before a new initialization procedure.



To launch a GCOS 7 session, the operator has to:

- check initialization parameters (V7000 properties screen).
- Enter **Init command** under V7000 to Start ISL operation then GCOS 7 loading.

How to launch a GCOS 7 LN Window

To be able to control GCOS 7 you have to start a LN window (line mode window). 2 methods are available:

- A “Start LN02” dialog box appears to automatically start the **LN02 console**.

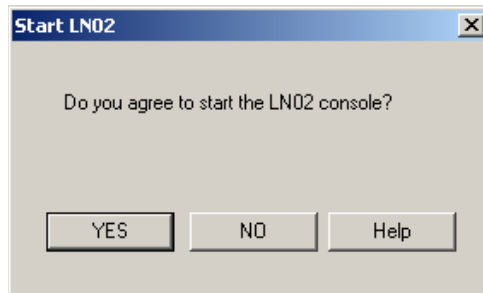


Figure 2-15. Dialog for automatic start of LN02 GCOS7 console

- It is also possible to use the V7000 service: **V7000 Operation Supervisor (VOS)**
This service is accessible in the V7000 item of the START menu of Windows™.

In the right part you have the list of LN consoles and information relating to each of them.
You may start one of them by right clicking on the LN resource itself (client type).

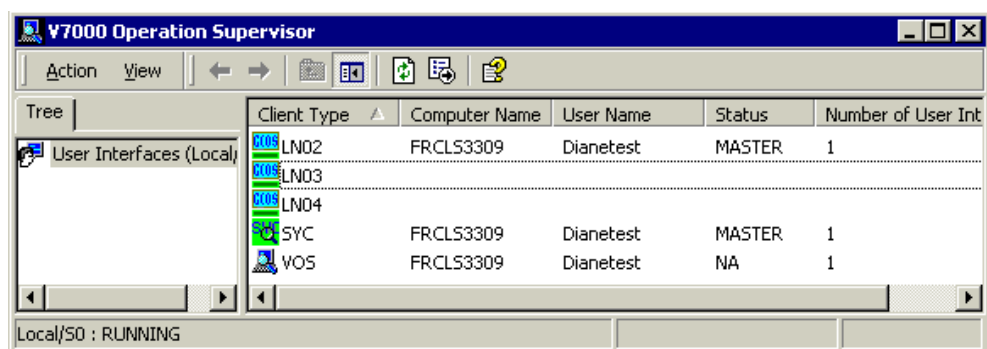


Figure 2-16. V7000 Operation Supervisor screen image



An LN can have several users at the same time, only one of them will be master and can enter commands in the window, the other users will have only the screen visibility, a Slave user can ask the master user to take control.

3 GCOS 7 consoles are available and correspond to the LNO2, LN03 and LN04 GCOS 7 resources.

Each of them can be used locally or remotely: one user is master and other users are slaves.

Refer to V7000 Operator's guide to have more details.

When a LN is started, the window looks like the screen image below:

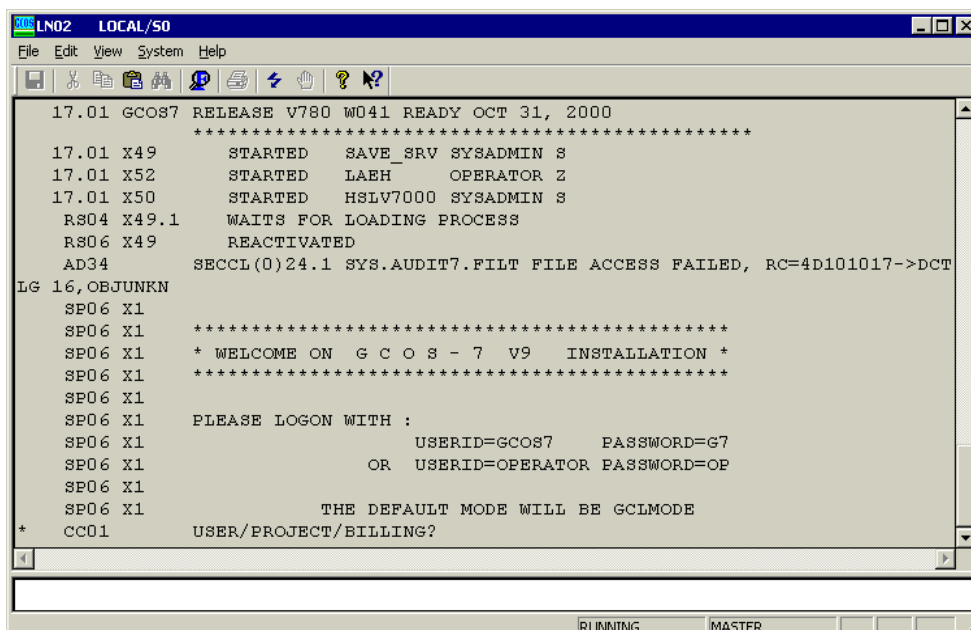


Figure 2-17 GCOS 7 console screen image

In this window you have the usual GCOS 7 visibility of all previous platforms.



Several little changes have been developed:

- When you want to enter a text for GCOS 7 (when asked by GCOS 7 or after entering the break command) you will have to enter the text in a separate window (line at the bottom of the screen). When the text is validated (Enter key) it will be placed in the normal GCOS 7 screen.
This input line is only validated when GCOS 7 waits for an input, generally at the GCOS 7 prompt S:
- Full copy / past features
- Command editor
- Several font are possible
- ... refer to V7000 Operator's Guide to get more details

All GCOS 7 commands are unchanged. (see the following sections of this manual).

2.5.3 Powering Off the system

Before powering Off the system, operator should take care of the following items:

- GCOS 7 session is completely terminated.
<< GCOS end of session message on the operator console (LN window).
If GCOS 7 is still running, enter `TSYS GCOS 7` command and wait for its completion.
- V7000 is terminated: have a look in the SYC Properties window to see the V7000 state.
Refer to Figure 2-24.
If V7000 is still running, enter a `terminate` command under SYC service.
Refer to Figure 2-25.
- Windows™ has to be terminated correctly: use shut down command in the Start menu of Windows™.
When the Windows™ session is completed, and depending of BIOS parameters:
 - a message will indicate that you can manually power off the server:
Press the power switch on the front panel of the server.
 - The Server will be automatically powered off after Windows™ shut down.



2.5.4 Windows™ and GCOS 7 coexistence

A system which uses 2 different operating systems has to be administrated carefully.

V7000 and GCOS 7 software need a specific Windows™ configuration, as well as different independent software to manage the whole DPS 7000/XTA configuration. This configuration (Windows™ and software) is built in Bull factory or on site by Bull people if they need to re-install the Windows™ system disk.

The DPS 7000/XTA platform is mainly dedicated to run GCOS 7 applications, whereas the Windows™ part is used to run:

- GCOS 7 interoperability solutions
- Software validated by Bull (Oracle, SQL net , ...)

In any case, the customer is authorized to run standard Windows™ applications without the Bull agreement.

Any change in:

- ✓ Windows™ configuration (add or uninstall Windows™ components, uninstall independent software, ...)
- ✓ Windows™ parameters (change hardware configuration, network addresses)
- ✓ Windows™ start-up conditions (automatic services, ...)

may lead to unpredictable results regarding the V7000 and GCOS 7 production. All actions resulting in Windows™ registry modifications have to be analyzed in advance and, at least, Windows™ system disk should be saved before such operations.

In case of doubt, contact Bull customer service to secure the operation.



Here are the W2000 options needed by V7000 and GCOS 7.

Group	Function	option	Used in V7000
accessories & utilities			
	accessibility wizard		
	accessories		
		calculator	
		character map	
		clipboard viewer	
		desktop wallpaper	
		document templates	
		mouse pointers	
		object packager	
	communications		YES
		chat	
		hyper terminal	
		phone dialer	YES
	games		
		freecell	
		minesweeper	
		pinball	
		solitaire	



Group	Function	option	Used in V7000
	multimedia		
		CD player	
		media player	
		sample sounds	
		sound recorder	
		utopia sound scheme	
		volume control	
certificate services			
	certificate services CA		
	certificate services Web enrollment support		
indexing services			
IIS			YES
	common files		
	documentation		
	FTP server		YES
	frontpage 2000		
	IIS snap-in		
	Internet services mger (HTML)		
	NNTP service		
	SMTP service		
	visual interdev RAD		
	WWW server		
managt & monitoring tools			YES
	connection manager cpts		
	network monitor tools		
	SNMP		YES
message queueing services			
networking services			



Group	Function	option	Used in V7000
	COM Internet services proxy		
	DNS		
	DHCP		
	Internet authentication		
	QoS admission control		
	simple TCP/IP services		
	site server ILS services		
	WINS		
other network file & print services			
	file services for Macintosh		
	print services for Macintosh		
	print services for Unix		YES
remote installation services			
remote storage			
script debugger			
terminal services			YES
	client creator files		YES
	enable terminal services		YES
terminal services licensing			YES
Windows™ media services			
	Windows™ media services		
	Windows™ media services admin		





3. Starting and Terminating a GCOS 7 Session

3.1 Introduction

This section describes the following operations:

- Initial System Load.
- Operating Modes: attended and unattended.
- Loading Operations: Attended Mode.
- Answering IL01 OPTIONS question.
- Date and Time Management.
- Terminating the Session.
- Error Messages displayed During System Loading.

These phases are the normal continuation from initialization, described in Chapter 2 *Starting Up Your System*.



3.2 Initial System Load (ISL)

3.2.1 Purpose

The Initial System Load (ISL) operation has a triple purpose.

1. Loading the operating system into main memory.
2. Specifying the session characteristics, which are:
 - RESTART, which may be WARM, COLD, CLEAN
 - specification as to which disk volumes are to contain backing store files, which are to be RESIDENT and which are to be SHARED with a coupled system.
 - specification of options such as INIT (AUTO, NAUTO), REPEAT, etc.
This is done by entering commands after the IL01 OPTIONS ? question.
3. Processing information left over from a previous session (e.g. dumps, PLMs, information on jobs and their associated outputs).

3.2.2 Operations

1. The system status at the end of the previous session is displayed.
2. Question IL01 is asked which allows the selection of various options (DUMP, RESTORE, CLEAN, COLD, WARM RESTART, ...) and enables the operator to specify or modify elements of the session identity (resident, backing store or shared disks, etc.). Automatic storage load is described in Section 3.3 *Operating Modes*.
Manual storage load is described in Section 3.4 *Loading Operations: Attended Mode*, including the full set of IL01 options.
3. A DUMP session is executed if required.
4. A RESTORE session is executed if required.
5. The system checks the state of all the devices and starts the initial AVRs (Automatic Volume Recognition).
6. A CLEAN, COLD, or WARM RESTART is executed.
7. A message indicating that GCOS 7 is available is displayed.



3.3 Operating Modes

3.3.1 Operating Modes

Two operating modes are available. These are:

- Manual mode, which requires a dialog between the operator and the system.
- Automatic (or unattended) mode.

At the end of system initialization, a MAIN operator may log on from any console with the required attributes, by entering the proper password or operator authorization when prompted by the system to do so. The operating mode of the session is determined by the INIT option provided at IL01 with the parameters SILENT/NSILENT, AUTO/NAUTO. The parameters NAUTO and NSILENT indicate that the operator is going to perform the session manually whereas the other two, AUTO and SILENT indicate that subsequent sessions will start automatically.

Note that the system catalog of messages is loaded during ISL instead of CONFIG in previous releases. The catalog is automatically reloaded:

- after each RESTORE option,
- if its creation date is different from that used during the previous session.

3.3.2 Selecting a Console

If the system initiates the first exchange, the system selects the console with the highest priority as found in the SRST.

If the operator initiates the first exchange, he should enter a break on the selected console. In this case, the current mode becomes NSILENT and system messages are displayed until the system is ready.

You can automatically switch from the current working console to another by entering a break on the new console.



3.3.3 Unattended Mode

This refers to the ability of GCOS 7 to load itself without manual intervention. If the SILENT mode is set, then loading can take place with the console(s) off-line.

The INIT (AUTO, SILENT), and REPEAT options provided at IL01 are associated with UNATTENDED MODE.

SILENT	only messages requiring the operator to take action or reply to questions are displayed on the console.
AUTO	means that loading options are supplied automatically (on condition that the right subfiles are present; see below).
REPEAT	allows software loading to be done automatically after the TERMINATE_SYSTEM command or after a system crash.

The library members named RELOAD_OPTIONS (in the case of a GCOS 7 reload) or INIT_OPTIONS (in the case of a GCOS 7 reload preceded by a hardware/firmware reinitialization) supply default replies to the IL01 question.

The library member named H_SYS_REPLY supplies default replies to GCOS 7 questions which are asked during initialization; it contains a predefined reply for each question key and subkey.

RELOAD_OPTIONS, INIT_OPTIONS and H_SYS_REPLY are stored in the SYS.HSLLIB default source library, and can be modified on the site. Each record of the file H_SYS_REPLY consists of:

```
KEY = XXnn [SUBKEY = dec2] REPLY = 'reply text'
```

KEY is the question key containing two letters and two digits.

SUBKEY is the question subkey containing two digits.

REPLY introduces the reply itself.

System components test the AUTO option and apply default values to those questions for which no reply was defined.

Of course, manual operator intervention may be needed to answer certain questions, unusual requests such as device recovery or an error message preventing successful loading.



In this unattended mode, no system messages appear on your console but they are stored in the SYS.LOGC file. You may retrieve them later by using the DISPLAY_CONSOLE_LOG (DCLOG) command or by producing a printout. During software loading, all questions request a deferred reply in order to facilitate console configuration. If an incident occurs (for example, the system cannot find a console or is waiting for the operator to take an action), the system then enters a dynamic wait state in which it periodically tries to dialog with the operator.

You may receive pending action messages on a local console. Once the operator takes the required action, unattended loading continues normally.

3.3.4 Operator Commands Allowed During GCOS 7 Initialization

To activate the system, enter a break. During storage load, you can only reference a subset of the GCOS 7 main operator commands:

- START_DEVICE (SDV)
- CANCEL_REQUEST (CR)
- MODIFY_VOLUME (MDV)
- NAME_VOLUME (NV)
- DISPLAY_TIME (DTM)
- DISPLAY_CONSOLE_LOG (DCLOG)
- REPLY (REP)
- MODIFY_CONFIGURATION CLM (MDC CLM)
- DISPLAY_REQUEST (DR)

3.3.5 Console Failure

If the console in use fails or becomes unavailable, for example goes off-line, the operator may interrupt the loading procedure.

To select another console which is to be dynamically reconfigured, see Section 3.3.2 *Selecting a Console*.

If the system sends a message requiring the operator to take action, for example, setting a device on-line or physical exchange of a device, the system re-issues the message until a console becomes available.



3.3.6 System Startup

Once the loading is completed, and just before the system is declared ready, a startup procedure named SYSTEM is executed automatically before any queued job is restarted. The absence of the member SYSTEM in the library SITE.STARTUP does not prevent the system from operating.

The purpose of the SYSTEM startup is to execute a set of installation-specific commands before operators or users log on.

The startup can:

- start telecommunication sessions,
- set up specific parameters, for example, job classes.

For sites without a local console, these facilities allow you to:

- connect a remote operator,
- start a RMOS session (see further on).

Interactive processors cannot be activated on the SYSTEM startup. SYSTEM startup is a set of commands (by default, these commands are in GCL language) with sequences which may depend on switches themselves set by the type of restart:

SW1	is set on for a WARM RESTART.
SW2	is set on for a COLD RESTART.
SW3	is set on for a CLEAN RESTART.
SW4	is set if a dump is present in the SYS.SYSDUMP file.

SW5 and SW6 concern coupled systems configurations:

SW5 = 0 and SW6 = 0	the local system is not configured as part of a coupled system (it is independent).
SW5 = 1 and SW6 = 0	coupled systems configuration: the local system is even.
SW5 = 0 and SW6 = 1	coupled systems configuration: the local system is odd.

NOTES:

1. The SYSTEM startup commands can also be in JCL, in which case the switches are numbered from 0 to 5.
2. A TDS job must not be submitted in the system startup as this can cause unexpected TDS behavior.



3.3.7 System Ready Transition

When the SYSTEM startup is executed, the Operator Message Handler (OMH) is automatically activated in job class Z. This job becomes an integral part of GCOS 7.

During the initialization of the OMH job, system messages are not normally displayed on the operator terminal and breaks are inhibited for a few moments. However, no messages are lost and if you wish, you can retrieve them from the SYS.LOGC file.

If OMH does not start, you cannot start a job. To correct the software error, call the Service Center. See the *Operator's Guide* for the platform concerned for an explanation of the maintenance procedures.

Because you may wish to operate in unattended mode, or do remote operations, the system, when ready, does not request the MAIN operator to log on.

3.3.8 Loading Operations: Unattended Mode

Certain operations can be performed without operator intervention:

- Initial System Load (ISL)
- restart after a crash.

The parameters which affect automatic operations are the AUTO parameter of the INIT option and the REPEAT option provided at IL01.

The AUTO parameter permits an automatic reply to the IL01 OPTIONS question and REPEAT permits software reloading without operator intervention after a crash or after TERMINATE_SYSTEM. The INIT and REPEAT options can be displayed, set or modified during the GCOS 7 session by using the DISPLAY_RESTART_OPTIONS (DRSO) and MODIFY_RESTART_OPTIONS (MDRSO) commands.



3.3.8.1 DISPLAY_RESTART_OPTIONS (DRSO) Command

Displays the following system initialization parameters:

- automatic restart option
- silent restart option
- automatic repeat option
- the initialization device

```

{DISPLAY_RESTART_OPTIONS}
{DRSO                      }

```

EXAMPLE:

DRSO

```

>>16.13 AUTO=0/0 SILENT=0/0 REPEAT=0
        SYSTEM DUMP AVAILABLE
        ANTESP SWITCH IS OFF INIT=MS17

```

□

NOTE:

For a description of the values displayed, refer to the DRSO command in Chapter 7 *Description of the Operator Commands*.

3.3.8.2 MODIFY_RESTART_OPTIONS (MDRSO) Command

Modifies the system initialization options for the next restart only, after a crash or a TERMINATE_SYSTEM command. (See Section 3.3.8.3 *Automatic Reload* for modifying initialization parameters for all subsequent restarts.)

```

{ MODIFY_RESTART_OPTIONS }
{ MDRSO                  }

```

```

[REPEAT = bool]
[AUTO   = bool]
[SILENT = bool]

```

```

[ {DEVICE}          ]
[ {                } = MSxx ]
[ {DV               }      ]

```



Parameters

REPEAT	when 1, the next system shutdown or crash is immediately followed by a restart, reload or re-initialization (depending on the system and the cause of the shutdown) without any operator intervention. This operation takes into account the disk unit identification stored in the service processor through the DEVICE parameter. Consequently, when REPEAT is set to one the operator must check the device identification stored in the service processor. When 0, the next system shutdown will require a manual intervention from an operator to restart the system. In the case of CRASH the system is restarted and reloaded automatically, taking into account the unit that was the system disk in the current session. This operation is necessary to process consistently the DUMP related to the CRASH.
AUTO	when 1, the next system restart (whether with or without operator intervention) is automatic, i.e. does not ask the operator any question, provided that default replies have been stored in the RELOAD_OPTIONS library member, as appropriate. When 0, the next system restart is run by the operator who is asked questions.
SILENT	when 1, the next system restart (with or without operator intervention) results in no message being issued on the operator console; when 0, messages are issued.
DEVICE	this parameter is used only on DPS 7000 systems. It specifies the disk unit from which the operating system is to be initialized. Unlike the other parameters, this option is retained for all subsequent system initializations until another MDRSO command is issued.

Constraints

If AUTO = 0 then SILENT is ignored.

**EXAMPLES:**

MDRSO AUTO SILENT (Next restart asks no question and issues no message.)

MDRSO REPEAT = 0 (Manual intervention is required to initiate the next restart.)



To display the repeat, restart, silent restart, and initialization device options, use the DISPLAY_RESTART_OPTIONS command described in Section 3.3.8.1 above.

3.3.8.3 Automatic Reload

The "IL01 OPTIONS" question is asked at the local console during a Reload (subsequent to a TSYS command or a crash), on condition that INIT (NAUTO) is set. If INIT (AUTO) is set, answers are read from disk. Note that setting INIT (AUTO) at the IL01 OPTIONS question applies to all subsequent reloads, but that the value set by the MDRSO command applies only to the next reload.

When the whole system is (re)initialized in unattended mode (auto-hardware mode) the reloading of GCOS 7 is automatic, even if AUTO was not set at the end of the GCOS 7 session. In this case the INIT_OPTIONS member is used to answer the ILO1 OPTIONS question.

3.3.8.4 Automatic Restart

This is controlled by the REPEAT option provided at IL01 parameter. If repeat is set to OFF a manual restart is needed after the end of a GCOS 7 session.

REPEAT can be set so that an automatic restart is made after the end of session or a crash. One way is to use the MDRSO command, but this is only effective for the next restart. The other way is to set the REPEAT Option, with no parameters, as one of the replies to the IL01 question. You can additionally use the minutes parameter of the REPEAT Option, which allows the user to set a minimum time between automatic restarts of crashes. If two consecutive crashes occur during this time, Auto mode is stopped.



3.3.8.5 Cancelling Auto Mode

There are two ways to cancel Auto mode:

- By using the MDRSO AUTO=0 command, when GCOS 7 is running. After a crash or a shutdown, for the next GCOS 7 restart only, the IL01 question will be asked at the system console.
- By using the prerecorded reply, INIT(NAUTO), to the IL01 question, after a crash or shut down, for example. In this case, the Auto Mode cancellation only becomes effective for the subsequent GCOS 7 sessions, but not the session being started.

Modification to the preregistered answers to IL01 must be done by an update to the RELOAD_OPTIONS library member.

NOTE:

By using both of the above possibilities you can cancel Auto Mode for the next and subsequent sessions.

3.3.8.6 Dump Facilities in Automatic Mode

AUTO and REPEAT modes are necessary for an automatic restart in case of crashes. For dump storage after crashes in automatic mode the following commands should be registered in the RELOAD_OPTIONS library member:

DUMP(DISK) This permits the dump option commands to be retrieved from DUMP_CD parameters, instead of from the console.

DUMP_CD (O,volnm,DEF; EXEC,DPS_STD;END;) This stores the commands which save the GCOS 7 dump in the SYS.SYSDUMP file. A standard dump is produced.

In the case of a crash dump in automatic mode the question CRASH COMMENTS? is not issued and the reply "AUTO" is provided by the system.

See Section 3.4.3.5 *DUMP* for a detailed description of Dump facilities.



3.3.8.7 Abnormal Events During First Phase of GCOS 7 Loading (ISL)

In AUTO mode, any abnormal event occurring during the first phase of GCOS 7 loading (ISL) and for which no response is automatically provided by the system or for which an operator action is needed, leads to display of the related ILxx error message prefixed by the "##FW" character string:

```
##FW ILxx .....
```

The system then enters an endless loop. The error must be corrected and a new Initial System Load executed, or the hardware and firmware must be manually reinitialized as described in Chapter 2.

If an I/O error occurred on system files (SYS.BKSTi, SYS.LIBi, SYS.PVMFi, SYS.TVMFi) during last GCOS session, the message IL21 will be displayed and the AUTO mode will be stopped (IL72). In this case, the IL01 question is then displayed to the main operator in order to modify the corresponding list of system files.



3.4 Loading Operations: Attended Mode

The ISL banner consists of two or three lines. The first line shows the technical status of ISL components:

```
>>> ISL-vvv.v xx -n (aa bb ...) / yy -m (cc dd ...)
```

where

vvv.v	= version.revision
xx -n (aa bb ...)	refers to SERs (Software Error Report) for INIT of ISL, where: xx = highest SER rank n = number of missing SERs aa bb ... = ranks of missing SERs
yy -m (cc dd ...)	refers to SERs for DPS of ISL, where: yy as for xx m as for n cc dd ... as for aa bb

The second line shows the current local date, time and the time deviation value, when available:

```
>>>HH:MM:SS month DD,YYYY TIMEDEV=+01.00
```

NEW TIMEDEV is used instead of TIMEDEV if the time deviation value has changed.

The third line shows the channel number of the ISL device (LOAD PCLC) preceded by the channel number of the hardware device (INIT PCLC), when Initial System Load occurs just after the INIT phase:

```
>>>LOAD PCLC = Chn1  
          OR  
>>>INIT PCLC = Chn1 LOAD PCLC = Chn2
```




3.4.2 IL01 Options

NOTE:

Examples of answers to IL01 OPTIONS question are given in Section 3.4.7 *Example of Answer to the IL01 OPTIONS ? Question.*

The answer given to the IL01 question directs the remaining loading operations. This answer can consist of several elements chosen by the operator, depending upon the following:

- Whether or not a Dump session should be performed.
A dump session is needed if the previous session terminated abnormally and if a dump is available. A dump session allows the dump to be printed on the line printer, displayed at the operator console, or saved on a disk for later processing. At the end of the Dump Session, the IL01 question is displayed again.
- Whether or not a Restore Session is needed.
- Whether or not a Cold Restart or a Clean Restart should be performed.
The default option - a Warm Restart - is the usual case; sometimes a Cold Restart and, in rare circumstances, a Clean Restart is required.
- Whether or not MAM (Message Access Method) is to be included as parameter in the answer to the IL01 question to define handling of telecommunication message queues.

The answer to Question IL01 can also establish the status of the session by indicating or modifying the following:

- the list of resident disks
- the list of backing store disks
- the list of shared disks (coupled systems)
- the behavior of the system for the next storage load
- the behavior of the system for the next system interruption.

After normal shutdown, the operator simply presses the EXECUTE (or TRANSMIT) key to perform a Warm Restart. The date and time are current; the list of resident, backing store or shared disks is unchanged, unless message IL16 has been displayed before the IL01 question.



3.4.3 Answering the IL01 Question

After indicating the system previous status, the system asks the question:

```
* IL01 OPTIONS ?
```

and the operator replies with one or more commands from the list shown in the following two pages. This answer directs the remaining loading operations and specifies the type, status and characteristics of the session. These commands are described in detail in the following Sections.

NOTES:

1. The maximum length of element lists (elements between left and right parentheses) is 255 characters.
2. The separator of elements inside element lists is the comma, except for the DUMP_CD command where a semi-colon is used instead.
3. The separator of the ISL options defined by the various following key words may be a comma, semi-colon or blank.

```
[{BKST(/[,volnm1, ... , volnm15])}]
[ {DATE(mm.dd.yy/hh.mm.[ss])} ]
[ { [ { * } ] } ]
[ { [ { BANNER } ] } ]
[ { DISPLAY [ ( { STATUS } ) ] } ]
[ { [ { TIME } ] } ]
[ { [ { OPTION } ] } ]

[ { DISK ( [ libnm.. ] member-name [:volnm] ... ) } ]

[ { [ { CONS } ] } ]
[ { DUMP [ ( { } ) ] } ]
[ { [ { DISK } ] } ]

[ { DUMP_CD ( IL03 response ; [ IL03 response ; ] ... ) } ]

[ { [ { ON } ] } ]
[ { HA ( { } ) } ]
[ { [ { OFF } ] } ]

[ { [ { CONS } ] } ]
[ { HELP ( { } ) } ]
[ { ? [ { PRINTER } ] } ]
```



```
[{      [ {NAUTO} {NSILENT} ]           }]  
[ {INIT[ ( {      }, {      } ) ]       }]  
[ {      [ {AUTO } {SILENT } ]         }]  
  
[ {LIB(volnm0 [ ,volnm1, ...volnm15])   }]  
  
[ {      [ {YES      } ]               }]  
[ {MAM[ ( {REFORMAT} ) ]               }]  
[ {      [ {NO      } ] ]               }]  
  
[ {NSYSSTUP                             }]  
[ {      [ {11      } [ {6} ] ]         }]  
[ {PAGESIZE[ ( {      } [*{ } ] ) ]     }]  
[ {      [ {n[.m] } [ {8} ] ] ]        }]  
  
[ {PVMF(volnm0 [,volnm1, ...volnm15])   }]  
  
[ {      {LIB [, PVMF] }                 }]  
[ {REFORMAT ( {      } ) ]              }]  
[ {      {PVMF [, LIB] }                 }]  
  
[ {      {OFF}                           }]  
[ {REPEAT ( {      } ) ]                 }]  
[ {      {mm }                            }]  
  
[ {RESIDENT (volnm1 [,volnm2, ...volnm10]) }]  
  
[ {      {WARM }                          }]  
[ {RESTART ( {COLD } ) ]                 }]  
[ {      {CLEAN}                          }]  
  
[ {RESTORE [ (parameter-list) ]         }]  
  
[ {SHARED (volnm1, volnm2, ...volnm36)   }]  
  
[ {SHARED2 (volnm37, ... volnm72)        }]  
  
[ {SHARED3 (volnm73, ... volnm108)       }]  
  
[ {SHARED4 (volnm109, ... volnm144)      }]  
  
[ {SHARED5 (volnm145, ... volnm180)      }]  
  
[ {SHARED6 (volnm181, ... volnm200)      }]  
  
[ {SPOOL (max, cache-option, cache_size, domain1, }]  
[ {      [ ...domain20 ] ) }]
```



```

[ {           { WARM }           } ]
[ { SYSQM ( { CLEAN } )         } ]
[ {           { BACKUP }         } ]

[ {           { + }             } ]
[ { TIMEDEV ( { } hh [.mm] )    } ]
[ {           { - }             } ]

[ {           { ON }            } ]
[ { TRC ( { } )                 } ]
[ {           { OFF }           } ]

[ { TVMF ( volnm0 [, volnm1, ... volnm15] ) } ]

```

3.4.3.1 BKST

```
BKST(/ [, volnm1, volnm2, . . . . volnm15] )
```

This option specifies the disk volumes which contain the temporary backing store files SYS.BKST, SYS.BKST1 up to SYS.BKST15. The first volume specified is the system disk which contains SYS.BKST, the second volume contains SYS.BKST1, the third contains SYS.BKST2 and so on up to SYS.BKST15. The slash "/" must be used instead of volume name to specify the system disk. The SYS.BKST should be on the system disk. If the first media name is not the system disk ("/"), the following message is displayed:

```
VM04 FATAL: ONLY / IS ALLOWED AS A MEDIA FOR SYSTEM BACKING STORE
```

The system then enters an endless loop. A new storage load must be performed and the operator must enter / to specify the volume name of SYS.BKST file.

The file SYS.BKST is a mandatory file. A session cannot take place without it. The absence or the error in opening this file causes the following message to be displayed.

```
VM06 FATAL: IT HAS BEEN IMPOSSIBLE TO OPEN SYSTEM BACKING STORE
```

The system then enters an endless loop. A new storage load must be performed.

Any volume name between volnm1 to volnm15 can be omitted indicating that the corresponding backing store file does not exist or is not to be used.

NOTE:

When a volume is removed, the commas must remain.



The list of volumes is used during the initial checking of device states and is retained from session to session even if, in the mean time, the system has been powered off. The command needs to be specified only during the first storage load following the creation of a new system disk or when it is required to change the list of backing store volumes.

The backing store volumes must be online and in READY state when the load is performed, otherwise the following messages are displayed:

```
IL61 MEDIA NAME NOT AVAILABLE FOR FILENAME
IL62 GIVE MEDIA NAME OR REPLY CR
```

The operator must enter the name of the volume where the given file is allocated.

The absence or an error in opening non mandatory files, i.e., SYS.BKST1, SYS.BKST2,, SYS.BKST15 produces the following warning message.

```
VM08 WARNING : THE sys.bksti HAS NOT BEEN OPENED
```

and the corresponding backing store file is not used for the session.

If the BKST contains only one file, the following message is displayed:

```
VM09 WARNING : NO OTHER PAGING BACKING STORE THAN THE SYSTEM
BACKING STORE IS ACTIVE
```

Disk mirroring is not supported on BKST volumes. If a MIRROR disk is declared in the BKST option, it is rejected with the message:

```
IL80 volnm: DEVICE MIRROR FORBIDDEN UNTIL END OF
INITIALIZATION
```

The concerned BKST file is ignored, message IL61 is sent to the operator console, and question IL62 must be answered. Refer to the *Mirror Disks User's Guide* for further details.

If all the BKST files are inactive, the following message is displayed:

```
VM10 FATAL: AT LEAST ONE BKST FILE IS COMPULSORY
```

The system then enters an endless loop. A new storage load must be performed. An error while loading or unloading the system structures produces one of the following messages:

```
VM01 FATAL: ERROR WHILE LOADING A SEGMENT FROM THE SYSTEM BACKING
STORE
```

```
VM02 FATAL: ERROR WHILE WRITING BACK A SEGMENT IN THE SYSTEM
BACKING STORE
```

The system then enters an endless loop. A new storage load must be performed.



However, it may be necessary to reformat the system disk depending on the Input/Output completion code. (Refer to Section 3.4.3.22 *RESTORE*).

The backing store volumes, other than the volume supporting the SYS.BKST file, are not normally resident. Those which are required to be resident must be declared in the RESIDENT option.

All disks declared in the BKST option have the BKST attribute (as returned by the DHW operator command), except for the system disk (containing the SYS.BKST file), which has the SYSTEM attribute.

3.4.3.2 DATE

DATE (mm.dd.yy/hh.mm[.ss])



IMPORTANT:

This command should be used only in very exceptional circumstances, for example if the Service Processor TIME-OF-DAY device (DPS 7000) is not working.

When **absolutely essential** this option can be used to set the reference date and time of the GCOS 7 session, where mm.dd.yy are numerical representations of the month, day and year and hh.mm.ss the time of day in hours, minutes and seconds. The values may be entered as one or two digits as appropriate. The reference date and time entered with the DATE option are used only for the current session and are not retained from session to session.

At the next reload of GCOS 7, the reference time is taken from the Service Processor. The local time is then calculated by the algebraic addition of the reference time to the time deviation value set with the TIMEDEV option.

If an invalid date is given the following message appears:

```
IL44 DAY/TIME M.D.Y/H.M [.S] ?
```

and the operator must reply with a valid date and time, for example:

```
I: 06.30.88/14.23.30
```



Then the system compares the local date and time resulting from the new reference time entered by the operator with the previous status date and time. If the current local date and time is lower, the following messages are displayed:

```
-----  
>>> PREVIOUS LOCAL DATE : hh:mm:ss MMM dd, yyyy   TIMEDEV = shh:mm  
  
>>> CURRENT LOCAL DATE : hh:mm:ss MMM dd, yyyy   TIMEDEV = shh:mm  
      IL70 SYSTEM WILL BE FROZEN  
      IL70 DO YOU AGREE (Y OR N) ?  
-----
```

s is either - or + . When the time deviation value has been modified the character string "NEW TIMEDEV" is used instead "TIMEDEV" in the "CURRENT LOCAL DATE" message.

The answer Y (yes) indicates that the operator agrees not to have the local time set back (an After Journal exists on the site): **the system will be frozen** until the current local time passes the previous status local time. The answer Y is forced in AUTO mode.

When the answer is not equal to Y (yes) the IL01 OPTIONS question is asked again in order to update the date or time deviation values or to continue the processing by pressing "Enter".

3.4.3.3 DISK

```
DISK([libnm..] member-name [ :volnm])
```

This option can only be used alone. It specifies that all the answers to the IL01 OPTIONS? question are in the named source library member. Default values are SYS.HSLLIB as the source library, and the system disk as the volume. After this answer, in the current session, answers to the IL01 question are read from the named source library member.

When the member addressed by the DISK option contains an erroneous option, a syntax error message is displayed and the IL01 OPTIONS question is sent again to the operator.

When a library member is given, it is retained from session to session. Consequently, when the AUTO mode is used sometime later, this library must contain the RELOAD_OPTIONS or INIT_OPTIONS and H_SYS_REPLY members.



3.4.3.4 DISPLAY

```

[ { * } ]
[ { BANNER } ]
DISPLAY ( ( { STATUS } ) ) ]
[ { TIME } ]
[ { option } ]

```

The operator should use this option to display the following:

*	The star (default value) means the following IL01 options are displayed: BKST, DISK, DUMP, DUMP_CD, HA, INIT, LIB, MAM, PAGESIZE, PVMF, REPEAT, RESIDENT, RESTART, RESTORE, SHARED, SHARED2, SHARED3, SHARED4, SHARED5, SHARED6, SPOOL, TIMEDEV, TRC, TVMF.
BANNER	The ISL banner is displayed.
STATUS	The system previous status is displayed.
TIME	The current local time is displayed.
option	Displays the value of any other IL01 option.

The * IL01 OPTIONS? question is asked again after the display.

The list of resident or shared disks displayed is from the previous ISL, it does not take into account any modifications that may have been made during previous GCOS 7 sessions.

This option cannot be used when stored in a library member.

**IMPORTANT:**

Use the DISPLAY (TIME) option to display the current local date, time and time deviation values. However, (DATE) and DISPLAY (TIMEDEV) give the last values entered under the DATE and TIMEDEV options of the IL01 question. These could be very old.



3.4.3.5 DUMP

```
      [ {CONS} ]  
DUMP [ ( { } ) ]  
      [ {DISK} ]
```

This option allows the operator either to print/display the dump left at the end of the previous session or to store it on disk (in the SYS.SYSDUMP file) to be subsequently printed or interactively analyzed by the DPAN utility.



IMPORTANT:

If a dump is not processed by a dump session during GCOS 7 loading, it will be corrupted by the current session, overwriting main memory and the backing store where most of the dump information is found.

The dump session dialog may be entered:

- from the console. This is done by answering question IL01 with the command: DUMP or DUMP(CONS).
- from disk. This is done by answering question IL01 with the command: DUMP(DISK)

If the DUMP command is issued, but no dump is available (e.g. there has been a normal termination of the last GCOS 7 session), the following message is displayed:

```
IL28 NO PREVIOUS DUMP
```

In this case, the dump session is terminated and the IL01 question is displayed again.

1. DUMP or DUMP(CONS)

If a dump is available from the previous GCOS 7 session, then the following question is asked:

```
IL03 DUMP?
```

The operator answers this question with one function at a time and the question is repeated until the dump session ends. Termination is done by entering END and pressing the EXECUTE or TRANSMIT key.

**NOTE:**

By using the HELP directive with IL03, you can obtain the syntax of all the DUMP commands.

To send a summary of the dump to the line printer, enter M in reply to the IL03 question and comments in reply to the IL14 message, but see Note below.

```
* IL03 DUMP? M
* IL14 CRASH COMMENTS?
```

```
I: (Enter your comments.)
```

```
...
```

```
* IL03 DUMP?
```

where M indicates that a mini dump (Mini Memory) is to be sent to the line printer

NOTE:

This facility should be used only as a last resort, if GCOS 7 cannot otherwise be restarted.

*IL14 CRASH COMMENTS gives the operator an opportunity to give any useful comments on the first page of the dump printout. (If more than one line of comments are to be given, each line must be terminated by a hyphen.).

The characteristics of the paper dimensions are taken from the ISL option, PAGESIZE.

To store a standard dump on disk, the following must be entered:

```
*IL03 DUMP? O, [efn:] volnm,DEF
```

```
*IL03 DUMP? EXEC,DPS_STD
```

where efn is the external file name (default is SYS.SYSDUMP), and volnm is the volume name of the disk which holds the SYS.SYSDUMP file. If it is the System disk, volnm can be replaced by "/". Note that the O (Outfile) in the operator reply is alphabetic.

The SYS.SYSDUMP file can be put on any disk. The dump overwrites any previously stored dump on the SYS.SYSDUMP file, as only one dump at a time may be written to this file.

Thus if this file already holds a dump from a previous session, which must be kept or which has not been printed, and a new dump occurs, the new dump must be printed (using the M function previously described to send it to the line printer). One alternative is to use another disk to hold the SYS.SYSDUMP file, if one is available. Another alternative is to use a different file on the same disk.



It is also possible to use a self defined dump procedure using "EXEC_LIB,library-name:volnm" and "EXEC,member-name" as answers to the ILO3 DUMP? question.

To change the default library name, the following can be entered:

```
*ILO3 DUMP? EXEC_LIB, libnm : volnm
```

EXEC_LIB defines the source library (libnm) to be opened for access to the subfiles named in subsequent EXEC commands. The default libnm is SYS.HSLLIB.

The EXEC command starts the execution of the dump commands as read from the named subfile. DPS_STD is the standard dump sequence source file. The IL01 question is reissued after dump commands, except in Auto Mode.

2. DUMP (DISK)

In this case, the answers to the IL03 question are entered from disk.

The answers must have been written previously on disk by answering question IL01 with DUMP_CD option (see Section 3.4.3.6 *DUMP_CD*, below).

3. The DPAN Utility

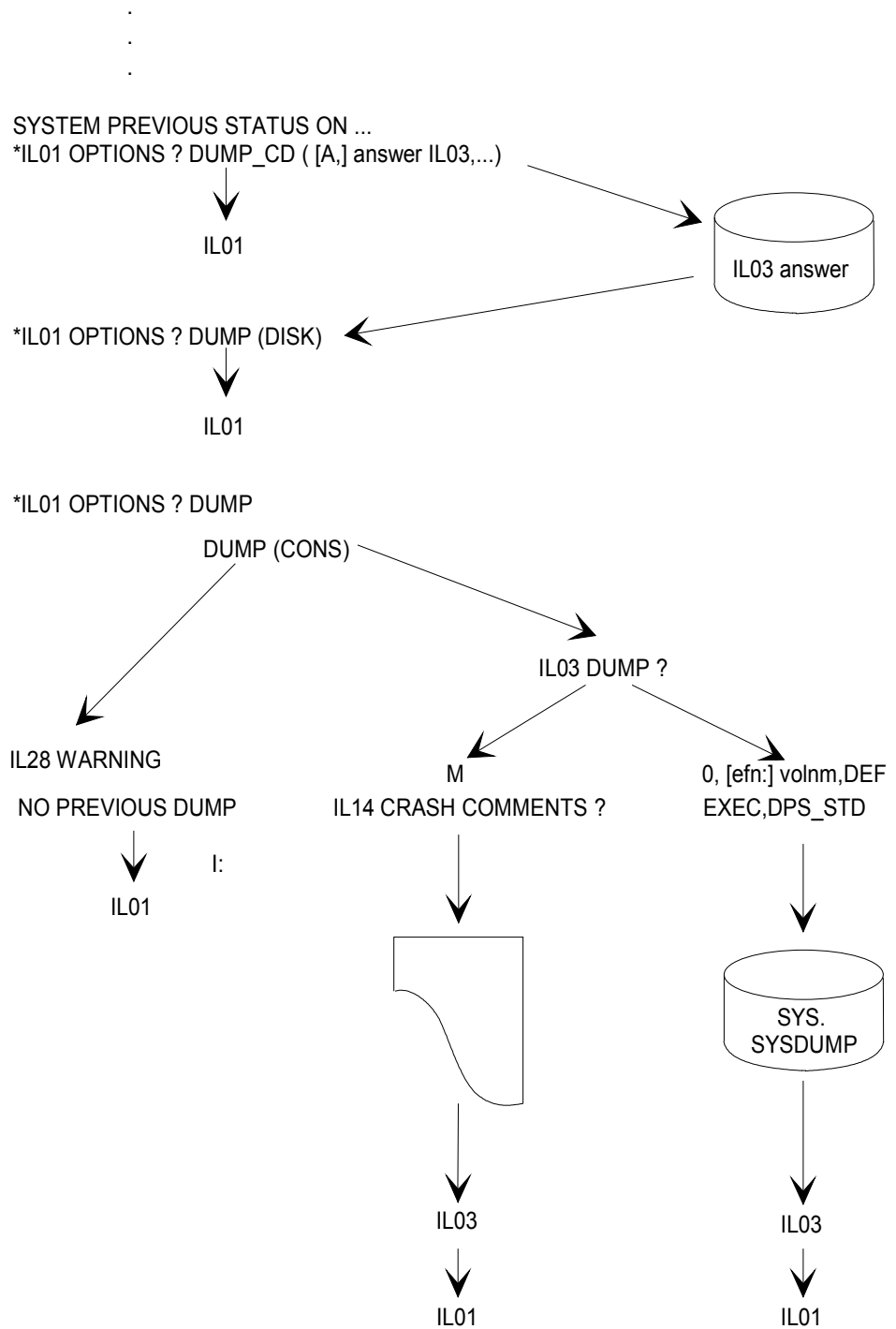
DPAN is a dump analyzer running under GCOS 7 which provides the operator with:

- a dump saving facility on tape (DP_SAVE)
- a dump printing facility (DP_PRINT).

See Appendix B *Calling Systems Utilities* for a detailed description.



4. SYNOPSIS OF FUNCTIONS RELATED TO DUMP





3.4.3.6 DUMP_CD

```
      [ {A} ]  
DUMP_CD ( [      ] ; IL03 answer ; [ IL03 answer ; ] ... )  
      [ {APPEND} ]
```

This option is used to record on disk the answers to the IL03 DUMP question (DUMP session). These recorded answers will be used when answering DUMP(DISK) to the IL01 question (refer to Section 3.4.3.5 *DUMP*).

When A or APPEND is used, the answers are appended to those already existing.

If the last recorded answer is not "END", the IL03 question is displayed again at the console when DUMP(DISK) is used.

NOTE:

This option does not create a library member. Consequently, it is not possible to update the DUMP_CD information with the maintain library processor.

3.4.3.7 HA (High Availability)

```
      { ON }  
HA ( {      } )  
      { OFF }
```

This option is for management of the High Availability product. When ON is entered, the Complex Management Service (CMSC) is started at GCOS 7 initialization. The default is ON.



3.4.3.8 HELP (or ?)

HELP (or ?)

This option (when typed first as a reply to the IL01 or the IL03 question) displays a HELP text with all possible replies to the *IL01 OPTIONS? and *IL03 DUMP? questions on a main console.

When the HELP text to the *ILO3 DUMP? question is displayed, the directive

MORE (Y/N)

appears. If "Y" is entered, additional HELP text will be displayed. If "N" is entered, the display terminates.

The HELP text may be directed to the line printer by using the form:

HELP (PRINTER)

The *IL01 OPTIONS? question is then displayed again.

This option cannot be used on a library member.

3.4.3.9 INIT

```

[ {NAUTO} {NSILENT} ]
INIT [ ( { } { } ) ]
[ {AUTO} {SILENT} ]

```

This option specifies the processing of the next, and subsequent GCOS 7 initializations. When the AUTO parameter is entered, the next GCOS 7 initialization automatically uses the ISL options stored on disk.

These options are stored in two members of the SYS.HSLLIB library, or two members of the library specified by the DISK option. Depending on whether or not there has been a firmware reinitialization, the options are read from different library members:

- When the GCOS 7 reload is preceded by a firmware reinitialization, the options are read from the INIT_OPTIONS member.
- When there is no firmware reinitialization preceding the GCOS 7 reload, the options are read from the RELOAD_OPTIONS member.



For the DPS 7000, the mode of the first initialization (e.g. after power up) is determined by the value of the Service Processor initialization parameter: AUTOMATIC or MANUAL. Subsequent reloads are automatic if the AUTO parameter is selected for the INIT option.

In AUTO mode the IL14 CRASH COMMENTS? question is automatically answered by the string "AUTO". If the SILENT parameter is also selected, no messages are issued to a main console during the next GCOS 7 initialization until device-testing time (see Section 3.4.4 *Date and Time Management*). This parameter is valid only if the AUTO parameter is also specified.

3.4.3.10 LIB

```
LIB(volnm0 [,volnm1, . . . .volnm15] )
```

This command specifies the disk volumes which are to contain the backing store files for permanent information, SYS.LIB, SYS.LIB1 up to SYS.LIB15.

The first volume specified is the disk which contains SYS.LIB, the second volume contains SYS.LIB1, the third contains SYS.LIB2 and so on up to SYS.LIB15. The slash "/" may be used instead of volume name to specify the system disk. Any volume name between volnm1 to volnm15 can be omitted indicating that the corresponding permanent backing store file does not exist.

The list of volumes is used during the initial checking of device states and is retained from session to session even if, in the mean time, the system has been powered off. The command needs be specified only during the first storage load following the creation of a new system disk or when it is required to change the list of backing store volumes.

The volumes supporting a SYS.LIBi file must be online and in the READY state when the load is performed, otherwise the following messages are displayed:

```
IL61      MEDIA NAME NOT AVAILABLE FOR FILENAME  
IL62      GIVE MEDIA NAME OR REPLY CR
```

The operator must enter the volume name of SYS.LIBi file.

The SYS.LIB is a mandatory file. A session cannot take place without it. An error in opening this file causes the following message to be displayed.

```
VM07 FATAL: THE SYS LIB IS MANDATORY AND IT HAS NOT BEEN POSSIBLE  
          TO OPEN IT
```

The system then enters an endless loop. A new storage load must be performed.



The absence or an error in opening of non mandatory files, i.e. .LIB1, SYS.LIB2,,SYS.LIB15 causes the following warning message:

```
VM08    WARNING: THE  sys.libi  HAS NOT BEEN OPENED
```

and the corresponding backing store file is not used for the session.

If all the LIB files are inactive, then the following message is displayed:

```
VM10 FATAL: AT LEAST ONE LIB FILE IS COMPULSORY
```

The system then enters an endless loop. A new storage load must be performed and the operator must specify the disk volumes which contain the backing store files for permanent information. At least the SYS.LIB file is mandatory.

Because of the information in SYS.LIBi files - sharable modules, pre-initialized load modules, checkpoints - during RESTART COLD or WARM, the configuration of these files must not be changed. If the media names are not the same as that of the previous session, the following message is displayed:

```
VM05 FATAL: NEW MEDIAS HAVE BEEN SPECIFIED FOR THE PERMANENT BKSTS
          DURING A WARM OR COLD RESTART
```

The system then enters an endless loop. A new storage load must be performed.

Also, during a cold or warm restart, if there is a mismatch between the characteristics of SYS.LIB files for the current and previous sessions, the system crashes at GCOS 7 initialization. For example, if the file size has been modified.

Concerning these last two situations, the operator must do one of the following:

- To change a SYS.LIBi file, perform a new storage load specifying a clean restart. In this case, pre-initialized load modules and sharable modules are lost.
- To use the SYS.LIBi files from the previous session, re-enter the LIB parameters specifying the old volume names.

All disks declared in the LIB option have the BKST attribute (as returned by the DHW operator command), except for the system disk (if present in the list), which has the SYSTEM attribute.

The disks declared in the LIB option are not normally resident (except for the system disk, if relevant). Those which are required to be resident must be declared in the RESIDENT option.

Disk mirroring is not supported on BKST disks. If a MIRROR disk is declared in the LIB option, it is rejected with the message IL80 (refer to "BKST" above).



3.4.3.11 MAM

```

[ {YES} ]
MAM [ ( {REFORMAT} ) ]
[ {NO} ]

```

This option processes telecommunication message queues. The parameters are described below.

- | | |
|----------|--|
| YES | This parameter causes all queues to be left as they were. A roll back of queues assigned to each job is requested. (See " <i>File Salvager</i> " under WARM RESTART, below.) |
| | This is only possible if the RESTART(WARM) option is also specified. |
| REFORMAT | All queues are cleared. All messages are erased. |
| NO | No telecommunication job can be executed until the network is generated again. |

3.4.3.12 NSYSSTUP

With this option, GCOS 7 can be started without executing System Startup (SYSTEM member in the SITE.STARTUP library).

3.4.3.13 PAGESIZE

```

[ {11} ] [ {6} ]
PAGESIZE [ ( { } ) ]
[ {n [.m]} ] [ {8} ]

```

This option sets the characteristics of the paper dimensions where "n.m" represents the height of the paper in inches and the other parameter is the print density in lines per inch. The defaults are 11 and 6 respectively.



3.4.3.14 PVMF

```
PVMF(volnm0 [,volnm1, . . . . .volnm15] )
```

This option specifies the disk volumes which contain the permanent VMF backing store files SYS.PVMF, SYS.PVMF1 up to SYS.PVMF15. The first volume specified is the disk which contains SYS.PVMF, the second volume contains SYS.PVMF1, the third contains SYS.PVMF2 and so on up to SYS.PVMF15.

The slash '/' may be used instead of volume name to specify the system disk. Any volume name between volnm1 to volnm15 can be omitted indicating that the corresponding permanent backing store file does not exist.

NOTE:

When a volume is removed, the commas must remain.

The list of volumes is used during the initial checking of device states and is retained from session to session even if, in the mean time, the system has been powered off. The command needs to be specified only during the first storage load following the creation of a new system disk or when it is required to change the list of backing store volumes.

The Backing Store volumes must be online and in the READY state when the load is performed, otherwise the following messages are displayed:

```
IL61 MEDIA NAME NOT AVAILABLE FOR FILENAME  
IL62 GIVE MEDIA NAME OR REPLY CR
```

The operator must enter the volume name of SYS.PVMFi file.

The file SYS.PVMF is a mandatory file, a session cannot take place without it. The absence or an error in opening this file causes the following message to be displayed:

```
VM07 FATAL: THE SYS.PVMF IS MANDATORY AND IT HAS NOT BEEN  
POSSIBLE TO OPEN IT
```

The system then enters an endless loop and a new storage load must be performed.

The absence or an error in opening of non mandatory files, i.e. SYS.PVMF1, SYS.PVMF2,, SYS.PVMF15 causes the following warning message:

```
VM08 WARNING : THE SYS.PVMFi HAS NOT BEEN OPENED
```

and the corresponding backing store file is not used for the session.

If all the PVMF files are inactive, then the following message is displayed.

```
VM10 FATAL: AT LEAST ONE PVMF FILE IS COMPULSORY
```



The system then enters an endless loop. A new storage load must be performed and the operator must specify the disk volumes which contain the backing store files for permanent VMF files.

During a cold or warm restart, if the media names are not the same as that of the previous session, the following message is displayed.

```
VM05 FATAL: NEW MEDIAS HAVE BEEN SPECIFIED FOR THE PERMANENT BKSTS
DURING A WARM OR COLD RESTART
```

The system then enters an endless loop and a new storage load must be performed.

Also, during a cold or warm restart, if there is a mismatch between the characteristics of the SYS.PVMF files for the current and previous sessions, then there is a system crash at GCOS 7 initialization (for example, modification of file size).

Concerning these last two situations, the operator must do one of the following:

- To change a SYS.PVMFi file, perform a new storage load specifying a clean restart. In this case, all permanent VMF files are lost.
- To use the SYS.PVMFi files from the previous session, re-enter the PVMF parameters specifying the old volume names.

All disks declared in the PVMF option have the BKST attribute (as returned by the DHW operator command), except for the system disk (if present in the list), which has the SYSTEM attribute.

The disks declared in the PVMF option are not normally resident (except for the system disk, if relevant). Those which are required to be resident must be declared in the RESIDENT option.

Disk mirroring is not supported on BKST disks. If a MIRROR disk is declared in the PVMF option, it is rejected with the message IL80 (refer to "BKST" above).



3.4.3.15 REFORMAT

```
REFORMAT ( { LIB [ , PVMF ] }  
           { PVMF [ , LIB ] } )
```

This option is used to reformat the permanent backing store. The valid parameters are LIB and PVMF. This option forces a clean restart of GCOS 7 (permanent information is lost) even if RESTART(CLEAN) is not explicitly specified in the answer to IL01 OPTIONS question.

In case of an invalid parameter the following message is displayed:

```
VM03  ILLEGAL SYNTAX OF BACKING STORE COMMANDS
```

The system then enters an endless loop. A new storage load must be performed.

3.4.3.16 REPEAT

```
REPEAT ( { OFF }  
         { mm } )
```

This option specifies the processing of the next GCOS 7 interruption and the subsequent ones triggered either by the TERMINATE_SYSTEM command or by a crash.

When the parameter OFF is entered, the system is stopped after GCOS 7 interruption.

When mm (minimum time in minutes between two consecutive GCOS 7 interruptions by crash) is entered, GCOS 7 is automatically reloaded after its interruption. If this facility is used with the AUTO parameter of the INIT command, AUTO mode is stopped if two consecutive crashes occur inside the time specified by mm.



3.4.3.17 RESIDENT

```
RESIDENT({volnm1 | /} [,volnm2,...])
```

This option specifies those disk volumes which are to be resident.

A disk declared in the RESIDENT option has the RSDT attribute (as returned by the DHW operator command) and is allocated to X1 (system). Therefore it cannot be held nor put on STANDBY. RSDT volumes are used, by default, to allocate temporary files.

The system disk need not be explicitly declared as resident. Although the RSDT attribute is not returned when the DHW command is issued on the system disk (only SYSTEM is displayed), the system disk is flagged as resident in the system tables; therefore it is taken into account for the allocation of temporary files.

Although it is not necessary to specify the system disk in this list, it can be specified, if desired, by replacing volnm1 by / (a slash) in the list, for example:

```
RESIDENT(/,HIT90,FSD15)
```

The list of resident volumes is used during the initial checking of device states and is retained from session to session even if in the meantime, the system has been powered off. Therefore it need be specified only at the first storage load following the creation of a new system disk or when the list needs to be changed.

The list of resident volumes; may be changed safely only during a cold or clean restart or during a warm restart where the previous session was terminated by a TERMINATE_SYSTEM command. If a warm restart is being performed after the abnormal termination of the previous session, avoid making changes to the list especially if the volumes to be changed contain files which were busy when the termination occurred.

Given that the list of resident volumes can be modified only at a restart and that most of the OLTD cannot take place on a device that supports a resident disk (since they run on out devices), the number of resident volumes should be kept to a strict minimum. In general, resident volumes should only contain permanent system files and temporary files; consequently, the frequently accessed user files should be cataloged and localized on non-resident volumes.

Error Detection

- The maximum number of resident volumes is 10. If more than 10 resident volumes are declared, the following message is displayed:

```
DV37 GCOS: NUMBER OF RESIDENT EXCEEDS 10, NEW LIST IS REQUIRED
```

and the DV43 question ("REPLACE BY") must be answered (see Section 3.4.5.2 *DV43 Question* below).



- Disk mirroring is not supported on resident volumes. If a MIRROR disk is declared in the RESIDENT option, it is rejected as follows:

```
DV38 DECLARED <RESIDENT> LIST:
      <volnm1> <volnm2>...
DV05 RESIDENT MIRROR FORBIDDEN:
      <volnmi> <volnmj>...
```

and the DV43 question must be answered (see Section 3.4.5.2 *DV43 Question* below).

Refer to the *Mirror Disks User's Guide* for further details.

- If a declared resident disk has not been found by the system, e.g. because the disk drive is STANDBY or because the volume name was erroneous in the RESIDENT option, the operator is notified as follows:

```
DV38 DECLARED <RESIDENT> LIST:
      <volnm1> <volnm2>...
DV40 RESIDENT NOT FOUND:
      <volnmi> <volnmj>...
```

and the DV43 question must be answered (see Section 3.4.5.2 *DV43 Question* below).

Note that the new list of resident volumes (result of the answer to DV43 question) keeps valid for all subsequent GCOS 7 sessions, until:

- either a new list is specified using the RESIDENT option at ISL time,
- or a RESTART (CLEAN) is performed. In this case, the VMF file in which the list was stored is lost, and the valid list is the list of volumes declared in the last RESIDENT option of ISL.



3.4.3.18 RESTART

```
RESTART ( { WARM }  
          { COLD } )  
          { CLEAN }
```

This option specifies how jobs, outputs, and system characteristics from the previous session are to be treated. The three types of restart are described separately.

NOTE:

The permanent Backing Store or the media cannot be changed with a WARM or a COLD Restart. The permanent Backing Store space, and the media specified and the number of media must be the same as in the preceding ISL.

3.4.3.19 RESTART(WARM)

This is the default option and does not need to be specified. It restarts the session as near as possible to the state in which it terminated. All system characteristics set by operator action, i.e. the multiprogramming level, job-class characteristics, are left unaltered. The events which occur vary depending on whether the previous session was terminated by the TERMINATE_SYSTEM command or whether it terminated abnormally. (See Section 3.5, *Terminating the Session*).

Warm Restart After a Normal Shutdown

1. All jobs and outputs are restored to the state they were in at the end of the previous session. Apart from the fact that the Output Writer, FEPS, GTWRITER, IOF, and RBF service jobs need to be started again, everything is exactly as it was before the shutdown.
2. One of the following messages is displayed for each job which was executing or in suspended state at shutdown time:

```
RS06 ron REACTIVATED
```

```
RR05 ron.ssn REMAINS SUSPENDED
```

The GCL commands DISPLAY_OUTPUT (DO) and DISPLAY_LOAD (DLD) may be used to check job outputs.



Warm Restart After an Abnormal Termination

1. The File Salvager checks the state of all files which were being used. A user file is said to be stable if it was either not opened or opened only in input mode. If some are unstable, that is, opened and in output mode, salvager action is usually needed. The UFAS access method provides a built in salvager and there is no action by the operator. However, for VBO files using the older BFAS access method, this question is asked:

*SV05 GCOS: FILE SALVAGING?

- If the answer NO is entered, salvaging is not attempted and no further messages are displayed.
- If the answer YES is entered, file salvaging is attempted and an SV09 message is displayed for each file indicating its state.

```

                                { VERIFIED      }
                                { RECOVERED     }
                                { UNSTABLE      }
SV09 efn FILE { UNALLOCATED  }
                                { VTOC DAMAGED  }
                                { NON STANDARD  }

```

VERIFIED = The file is stable.

RECOVERED = The file has been corrected and is stable.

UNSTABLE = The file cannot be corrected.

UNALLOCATED = A space allocation was being performed at the time of the abnormal system termination and has not been completed. The file must be reallocated.

VTOC DAMAGED = The VTOC (Volume Table Of Contents) of the disk is damaged and salvaging is not possible. Use the VOLCHECK utility to check the integrity of the disk.

NON STANDARD = the file (or the disk) is not a standard native type; no salvaging is possible. If the disk should be standard, then damage to the disk may have occurred; run the VOLCHECK utility.

- If a disk containing one of the files is not ready, then the following question is asked:

*SVO7 GCOS: MOUNT volnm?

I:

To salvage the file, mount the disk and reply YES. To bypass file salvaging, reply NO.



2. All jobs which were not in the executing state are unmodified.

RR05 ron.ssn REMAINS SUSPENDED

- RS05 or RR06 messages as appropriate are displayed for jobs which were executing or suspended between steps.
- For jobs which were suspended within a step when a crash occurred, the current step is aborted and the RR01 question is asked:

* RR01 ron.ssn ABORTED lm-name SEVn[=mmm] REPEAT [FROM CHKPT xx]?
 [Status] RC = ccccccc --> siu, return code

* RR01 ron.ssn ABORTED lm-name SEVn[=mmm] REPEAT WHOLE JOB?
 [Status] RC = ccccccc --> siu, return code

A SEV code is the severity of the step termination. This is usually SEV6 indicating a system crash. For the other possible severity codes (SEV), refer to the *IOF Terminal User's Reference Manual*.

The first RR01 question -

* RR01 ron.step ABORTED lm-name SEVn[=mmm] REPEAT [FROM CHKPT xx]?

is issued if the executing step has been declared as being repeatable by the user. If no checkpoint (CHKPT) is specified, the beginning of the job step is used as the checkpoint value. The second RR01 question -

* RR01 ron.step ABORTED lm-name SEVn[=mmm] REPEAT WHOLE JOB?

is issued if the step is not repeatable but the job was declared as repeatable by the user.

NOTE:

The ROLLBACK? question is no longer asked. Now, if a Before Journal is used, Rollback is always automatically performed. Even if the output has been partly or totally printed, it is rebuilt, re-enqueued, and printed again from the beginning.

The possible answers are as follows:

- | | |
|-----|--|
| YES | The step executes from checkpoint xx; if no checkpoint was declared, the step restarts from the beginning. If the job was declared repeatable, the job restarts from the beginning. Any output from a part of the job which repeats is erased except the Job Report. In the case of the last RR01 message, the file and communications queues are rolled back. |
| NO | The current step and job are aborted. If the step was under GAC (Generalized Access Control), the appropriate files are rolled back. |



ALL	Indicates that the answer YES is to be assumed for this and all subsequent RR01 questions which will not then be asked. The salvage session proceeds.
HOLD	This is only a valid reply to the REPEAT WHOLE JOB? question. The job is put into the HOLD state and must subsequently be released before it can be executed.

NOTES:

1. If the job was not executing a step, i.e., job or step initiation or termination was being performed, execution proceeds without the RR01 question being asked.
2. If the Before Journal was being used, ROLLBACK action is taken automatically.
3. Outputs remain in the states they were in when the previous session terminated. For each output being printed when termination occurred, the following question is asked:

*OU33 dvm OUTPUT ron:ouseqnb RESTART FROM?

The possible replies are the same as for question OU02 described in Section 5 *Running User Jobs Under GCOS 7*.

4. A report is prepared, in output class A, which lists all the actions taken concerning files and jobs.
5. The Output Writer, IOF, GTWRITER, and RBF service jobs are terminated and must be started again.



3.4.3.20 RESTART(COLD)

This option causes information left over from the previous session to be erased, and provides the operator with an empty system. It is used when it is not necessary to complete the jobs from the previous session or if a RESTART(WARM) has failed.

The following sequence of events occurs:

1. All existing jobs are lost.
2. The operator either retains the contents of the SYS.OUT file or erases it, by replying to the following question:

```
* OU16 SYS.OUT RECOVER?  
I:
```

If he replies YES, the SYS.OUT contents are retained and output requests are re-submitted with the same parameters as at output creation, but job reports and private output notifications are lost. If recovery is not possible, the following message is displayed:

```
OU35 UNSUCCESSFUL SYS.OUT RECOVERY
```

and the SYS.OUT contents are lost.

If he replies NO the contents of SYS.OUT are erased and any file extents beyond its original allocation are deleted.

NOTES:

1. During a RESTART(COLD), outputs that were present when the system was stopped are rebuilt with the information known at the time of creation. Any modification done after creation (by HO, RO, or MO commands) is lost.
2. The OU16 question is not asked if the SYS.OUT file is empty.
3. If the answer to the OU16 question is YES, output classes are returned to their original class.
4. The system multiprogramming level, job classes, job class characteristics, account information and Message Of Today (MOT) remain unaltered from the previous session.
5. Pre-initialized Load Modules (PLMs) remain the same as for the previous session, provided that the SYS.LIB file has not been changed.
6. Temporary files created on non-resident volumes during the previous session are not deleted. The VOLCHECK utility (VBO) or the DELETE_TEMP command of the MAINTAIN_VOLUME utility (FBO) must be run to do this.
7. No report is prepared.



3.4.3.21 RESTART(CLEAN)

The clean restart is most often useful to install GCOS 7, or to install a patch. It may also be used at other times when a cold or warm restart cannot be performed.

**IMPORTANT:**

This option has an extremely drastic effect, providing the operator with a completely empty system and erasing all information from the previous session. It is to be used with caution.

The following events occur:

1. All existing jobs are lost. After a RESTART(CLEAN), outputs can be recovered, according to the answer given to the SYS.OUT RECOVER? question.
2. All Pre-initialized Load Modules are cancelled.
3. All accounting information is erased.
4. All job classes except A..Z and RB are lost and all job class characteristics are reset to their default values.
5. The multiprogramming level reverts to the value set at system configuration.
6. The sharable module libraries are emptied, and must be reloaded.
7. The network definition is deleted.
8. No report is prepared.
9. The internal job numbering system is reset (Run Occurrence Number is reset to 0001).
10. Temporary files created on non-resident volumes during the previous session are not deleted. The VOLCHECK utility (VBO) or the DELETE_TEMP command of the MAINTAIN_VOLUME utility (FBO) must be run to do this.
11. Unless RESIDENT and SHARED options are used in the current restart, the lists of resident and shared disks are reset to those last specified by the IL01 options. Any modifications made since then are lost (because the lists are stored in a VMF file).



12. The operator either retains the contents of the SYS.OUT file or erases it, by replying to the following question:

```
* OU16 SYS.OUT RECOVER?  
I:
```

If he replies YES, the SYS.OUT contents are retained and output requests are re-submitted with the same parameters, but job reports and private output notifications are lost. If recovery is not possible, the following message is displayed:

```
OU35 UNSUCCESSFUL SYS.OUT RECOVERY
```

and the SYS.OUT contents are lost.

If he replies NO, the contents of SYS.OUT are erased and any file extents beyond its original allocation are deleted.

NOTES:

1. The OU16 question is not asked if the SYS.OUT file is empty.
2. If the answer to the OU16 question is YES, output classes are returned to their original class.

3.4.3.22 RESTORE

```
RESTORE[(system-file-description [,NCONFIG] [,FORMAT])]
```

where system-file-description has the following format:

```
efn: {volnm|/}
```

This option specifies that system segments, including system patches, can be restored from the SYS.SYSTEM file on the specified volume. The system file description is kept from session to session; it is established at initial system load. The parameters in the RESTORE option have the following meaning:

```
SYSTEM_FILE_DESCRIPTION := { efn: {volnm} }  
                          {      { / } }
```

NCONFIG indicates that the original copy of system segments is restored.

FORMAT indicates that SYS.BKST is to be formatted. This option must be specified when the file has been corrupted due to a physical Input/Output error.

Note that the volnm can be replaced by a slash (/).



A restore session is usually performed by answering the IL01 question during ISL and entering

```
RESTORE (SYS.SYSTEM:volnm)
```

The first time a restore is required, enter RESTORE; for subsequent restore sessions, answer RESTORE to the IL01 question and press the EXECUTE key.

If the SYS.SYSTEM file cannot be found on the disk specified, the following messages are displayed:

```
IL38 SYS.SYSTEM NOT FOUND ONTO media name
```

```
* IL63 GIVE <EFN:> MEDIA NAME OR REPLY CR
I:
```

and the operator enters the correct <efn:>media name after the "I".

In this case the correct<efn:>media is not kept and must be reintroduced the next time a RESTORE session is performed.

A RESTORE results in all job classes except A..Z being lost.

3.4.3.23 SHARED

```
SHARED(volnm1, volnm2, ... volnm36)
SHARED2(volnm37,      ... volnm72)
SHARED3(volnm73,     ... volnm108)
SHARED4(volnm109,    ... volnm144)
SHARED5(volnm145,    ... volnm180)
SHARED6(volnm181,    ... volnm200)
```

This option specifies which disk volumes are to be shared with another system. SHARED() or SHAREDi() specifies an empty list and erases the previous one. NOTE: One blank character is needed between left and right parentheses.

The SHARED option is only valid for coupled systems. The management of coupled systems is explained in the *Coupled Systems User's Guide*.

The list of sharable volumes is used during the initial checking of device states and is retained from session to session, even if in the meantime the system has been powered off. Therefore the parameter need be specified only at the first storage load following the creation of a new system disk or if it is required to change the list of shared volumes.

The shared volumes are not normally resident. If they are required to be resident they must be declared as such by the RESIDENT option.



It is also possible to change the list of SHARED volumes during the GCOS 7 session by using the MDHW command.

Error Detection

- The maximum number of shared volumes is 200. If more than 200 shared volumes are declared, the following message is displayed:

```
DV66 THE LIMIT OF 200 SHARED DISKS HAS BEEN REACHED
```

and the extra volumes are ignored.
- If the system disk, or a BKST disk (declared in BKST, PVMF, TVMF or LIB option), or the service disk (containing SPOS files, DPS7000/An only) is declared as a shared disk, then it is ignored and one of the following messages is displayed:

```
DV31 GCOS: SHARED OPTION FOR volnm IS IGNORED: ALREADY DECLARED  
AS SYSTEM DISK OR BACKING STORE
```



```
DV31 GCOS: SHARED OPTION FOR volnm IS IGNORED: SERVICE DISK  
CANNOT BE SHARED
```
- The volumes specified in the list must be supported by suitable devices which have the dual sharing capability. If this is not the case, the following message is displayed:

```
DV30 GCOS: DUAL SYSTEMS: ILLEGAL DEVICE FOR SHARED volnm  
NEW ISL MANDATORY
```

and an endless loop is entered. The operator must perform a storage load again and correct the list of shared volumes at next IL01 question.
- If the system has not been correctly configured for dual systems (DUALSYS statement of CONFIG), or if the Marketing Identifier for dual systems is not validated, the following message is displayed:

```
DV27 GCOS: SITE NOT CONFIGURATED FOR DUAL SYSTEM USAGE.  
SHARED OPTION IS IGNORED
```

and the SHARED option is ignored.
- If a declared shared disk has not been found by the system, e.g. if its disk drive is in STANDBY, or if its volume name was erroneous when entering the SHARED option, or if the disk is seized by the other system, the operator is notified as follows:

```
DV39 DECLARED <SHARED> LIST:  
<volnm1> <volnm2>...  
DV41 SHARED NOT FOUND:  
<volnmi> <volnmj>...
```




and the DV43 question must be answered (see Section 3.4.5.2 *DV43 Question* below).

The new list of shared volumes (result of the answer to DV43 question) stays valid for all subsequent GCOS 7 sessions, until:

- either a new list is specified using the SHARED option at ISL time,
- or the MDHW operator command is used to dynamically update the list,
- or a restart (CLEAN) is performed. In this case, the VMF file in which the list was stored is lost, and the valid list is the list of volumes declared in the last SHARED (i) option of ISL.

It is important to keep in mind that the list of shared volumes must be identical for each of the coupled systems. This is of basic importance when using mirrored disks (which may be declared as shared volumes).

3.4.3.24 SPOOL

```
SPOOL(max, cache-option, cache-size, domain1 [...domain20])
```

This option is used to determine the following:

- the number of available SYS.SPOOLi files,
- GCL cache usage,
- the list of GCL domains to be preinitialized from SYS.HBINLIB.

This option enables system response time to be improved.

Allocating a number of SYS.SPOOLi files enables sharing of GCL Input/Output over several files and thus speeds up response time. (See the **Max** parameter.)

The use of GCL Cache enables a certain number of command definitions to be stored in memory. (This information is extracted from a BIN type library and temporarily stored in a SYS.SPOOLi file.) The result is a reduction in the number of I/Os on a SYS.SPOOLi file. (See the **Cache Option** and **Cache Size** parameters.)

The use of pre-initialized domains enables the same command definitions to be shared by all system users, thus preventing congestion of the SYS.SPOOLi files and the GCL Cache. (See **Domain** parameter.)

When the SPOOL option is specified, GCL batch jobs that are not entirely repeated from the beginning are aborted. If this occurs, an error message is sent to the main console.



Parameters

max

SYS.SPOOLi files can be numbered from 0 to 9 -
SYS.SPOOL1, SYS.SPOOL2, ... SYS.SPOOL9.
($0 \leq i \leq 9$)

The max parameter represents the maximum value for i when SYS.SPOOL files are allocated. For example if two SYS.SPOOL files are allocated, SYS.SPOOL0 and SYS.SPOOL4, the value of max is 4.

If the number of SYS.SPOOLi files is not known, then max should be given a value of 9, which is the default value.

Values for i do not have to be consecutive, as GCL displays a message indicating which SYS.SPOOLi files are not available.

If no SYS.SPOOLi file can be accessed, GCL automatically allocates a temporary SYS.SPOOL file to enable the system to start.

Question: How many SYS.SPOOLi files do you need?

- in IOF or batch mode, you need 1 SPOOL per block of 10 users, with a maximum of 20 users per SPOOL.
- for TDS, 1 SPOOL is sufficient.

Cache-Option

This parameter defines if and how GCL cache is to be used:

0 = GCL Cache is not used.

This value applies for TDS based operation.

1 = GCL cache is used for commands applied to pre-initialized domains.

This value applies if IOF is not often used and if users have defined only a few private commands.

2 = GCL cache is used for all commands.

You are advised to select this value if IOF is used intensively on your system.

The default value for Cache-Option is 1.



Cache-Size	<p>This parameter helps to specify the size of GCL cache in memory. The value assigned to it gives the number of 64Kbyte segments to be reserved for cache. Each segment contains approximately 16 commands. The maximum value of this parameter is 8. (0<=cache-size<=8).</p> <p>The value you select depends on the use made of GCL.</p> <p>If cache-option=0, no segments need to be reserved and cache-size should be given the value 0.</p> <p>If cache-option=1, you need to reserve 2 to 4 segments.</p> <p>If cache-option=2, you need to reserve 4 to 8 segments, according to the number of IOF users. Reserve 8 segments if the number of IOF users could reach 40 or more.</p> <p>The default value for the Cache-Size parameter is 2.</p>
domain 1 [...domain20]	<p>Names of GCL domains to be preinitialized; up to 20 GCL domains can be preinitialized. Each domain name after the first in the command must be preceded by one or more spaces.</p> <p>Default = H_BREAK, IOF, H_NOCTX, EDIT, MAIN</p> <p>I/O time is saved when frequently used GCL domains are preinitialized.</p>

NOTE:

Refer to the DISPLAY_GCL_CACHE command in Chapter 7 *Description of the Operator Commands*.



3.4.3.25 SYSQM

```

      {WARM  }
SYSQM({CLEAN})
      {BACKUP}

```

The SYS.QM file (or its backup SYS.QM_BACKUP) is a file through which the so-called permanent messages pass (Permanent messages include interactive answers given to commands, etc.). This ensures that the messages are memorized if they cannot be delivered; if the person who is due to receive the message is not logged on, for example.

Three options exist. They are linked to the value of the RESTART parameter (see table below):

- SYSQM (WARM) leaves the file as it is (in the case of a RESTART WARM).
- SYSQM (CLEAN) clears the contents of the SYS.QM files, and starts logging messages on those same files again.
- SYSQM (BACKUP) switches to using the SYS.QM_BACKUP file for message logging. Use this option if SYS.QM is corrupted or cannot be accessed.

	RESTART	
SYSQM	RESTART (WARM)	RESTART (COLD) or RESTART (CLEAN)
SYSQM (WARM)	SYSQM (WARM) is the default option for RESTART (WARM). The contents of the files are unchanged	This combination is authorized, but it has the same effect as SYSQM (CLEAN)
SYSQM (CLEAN)	SYSQM is cleared, despite the RESTART (WARM) option	SYSQM (CLEAN) is the default value for RESTART (COLD) or RESTART (CLEAN). Contents of SYS.QM and SYS.QM_BACKUP are cleared.
SYSQM (BACKUP)	Use this option if you cannot access the SYS.QM file.	Use this option if you cannot access the SYS.QM file.



3.4.3.26 TIMEDEV

```
TIMEDEV({ + | - } hh [.mm] )
```

In normal circumstances the service processor clock provides a reference date and time (for example the universal time: UT) and a time deviation value, which may represent the local time zone, to get the local time used by GCOS 7. This option specifies the difference in hours (and minutes) between desired GCOS 7 local time and the reference time supplied by the Service Processor clock. The range of acceptable values is from -23.59 to +24.0.

It is possible to modify the time deviation with the TIMEDEV option.

After having entered the new time deviation value, the system checks the resulting local date and time. If this value is lower than the GCOS 7 time at the previous system termination the following messages are displayed:

```
>>> PREVIOUS LOCAL DATE : hh:mm:ss MMM dd, yyyy   TIMEDEV = shh:mm
>>> CURRENT LOCAL DATE  : hh:mm:ss MMM dd, yyyy   TIMEDEV = shh:mm
      IL70 SYSTEM WILL BE FROZEN
      IL70 DO YOU AGREE (Y OR N) ?
```

The s is either - or + .

The answer Y (yes) indicates that the operator agrees not to have the local time set back (An After Journal exists on the site.): **the system will be frozen** until the current local time passes the previous local time. The answer Y is forced in AUTO mode. Any other answer implies that the current local date and time is acceptable for starting a new session. Then, one of the following messages is displayed:

- no change of the time deviation value:

```
>>> hh:mm:ss MMM dd, yyyy   TIMEDEV = shh:mm      (s is + or -)
```

- the time deviation value has changed:

```
>>> hh:mm:ss MMM dd, yyyy   NEW TIMEDEV = shh:mm  (s is + or -)
```

This command, when accepted, updates the time deviation value in the Service Processor.

NOTES:

1. It is preferable to use the MODIFY_TIME command under GCOS 7.
2. As the TIMEDEV value can be modified while GCOS 7 is running, the DISPLAY (TIMEDEV) and DISPLAY (DATE) options may not give current TIMEDEV and DATE, but the last values entered under the DATE and TIMEDEV options entered as answers to the IL01 question. These can be old. However, to get the current time, date and time deviation values the DISPLAY (TIME) option can be used, or the GCL command, DISPLAY_TIME.



3.4.3.27 TRC

```
TRC [ { ON } ]  
    [ ( { } ) ]  
    [ { OFF } ]
```

This option activates the system trace function with the ON value and deactivates it with OFF. The default value is ON.

3.4.3.28 TVMF

```
TVMF (volnm0 [,volnm1,.....volnm15] )
```

This command specifies the disk volumes which contain the temporary VMF backing store files SYS.TVMF, SYS.TVMF1 up to SYS.TVMF15.

The first volume specified is the disk which contains SYS.TVMF, the second volume contains SYS.TVMF1, the third contains SYS.TVMF2 and so on up to SYS.TVMF15. The '/' may be used instead of volume name to specify the system disk. Any volume name between volnm1 to volnm15 can be omitted indicating that the corresponding temporary backing store file does not exist.

The list of volumes is used during the initial checking of device states and is retained from session to session even if, in the mean time, the system has been powered off. The command needs to be specified only during the first storage load following the creation of a new system disk or when the list of backing store volumes must be changed.

The Backing store volumes must be on-line and in the READY state when the load is performed, otherwise the following messages are displayed:

```
IL61      MEDIA NAME NOT AVAILABLE FOR FILENAME  
IL62      GIVE MEDIA NAME OR REPLY CR
```

The operator must enter the volume name of the backing store SYS.TVMFi.

The file SYS.TVMF is a mandatory file. A session cannot take place without it. The absence or the error in opening this file causes the following message to be displayed.

```
VM07 FATAL: THE SYS.TVMF IS MANDATORY AND IT HAS NOT BEEN  
          POSSIBLE TO OPEN IT
```

The system then enters an endless loop. A new storage load must be performed and the operator must enter the SYS.TVMF parameter.



The absence or an error in opening of non mandatory files, i.e. SYS.TVMF1, SYS.TVMF2,, SYS.TVMF15 causes the following warning message.

```
VM08 WARNING : THE SYS.TVMFi HAS NOT BEEN OPENED
```

and the corresponding backing store file is not used for the session.

If all the TVMF files are inactive, then the following message is displayed.

```
VM10 FATAL: AT LEAST ONE TVMF FILE IS COMPULSORY
```

The system then enters an endless loop. A new storage load must be performed and the operator must specify the disk volumes which contain the SYS.TVMFi backing store files. At least the SYS.TVMF file must be present.

All disks declared in the TVMF option have the BKST attribute (as returned by the DHW operator command), except for the system disk (if present in the list), which has the SYSTEM attribute.

The disks declared in the TVMF option are not normally resident (except for the system disk, if relevant). Those which are required to be resident must be declared in the RESIDENT option.

Disk mirroring is not supported on BKST disks. If a MIRROR disk is declared in the TVMF option, it is rejected with the message IL80 (refer to "BKST" above).

3.4.4 Date and Time Management

During a GCOS 7 session, dates and times included in the various console messages are always expressed in "local date and time". This local date and time is the result of the algebraic addition of the "reference date time" to the "time deviation".

For example, the reference date and time can be set to the Universal time: UT (or GMT time) and the time deviation set to the local time zone. On such a site, changing from winter time to summer time is made by adding one hour to the time deviation value.

On all the DPS 7000 systems the reference time is kept in the service processor clock and is never managed through GCOS 7. On DPS 7000/A, DPS 7000/4xx/Cxx, DPS 7000/5x0/7xx/8x0/Dx0/Mx0/MTxx models the time deviation is available through this clock and is managed globally by the service processor and GCOS 7. On other hardware models, as no time deviation is available on the service processor, time deviation is managed by GCOS 7.

On DPS7000/XTA, when V7000 starts, the reference date and time (UT) are got from WINDOWS™ instead of Service Processor.

Time deviation is managed through V7000 System Control. (refer to V7000 Operator's Guide).



When a GCOS 7 session is started the reference date and time is read from the service processor. Depending on the hardware models, the time deviation is read either from the service processor or from the previous GCOS 7 session information. Then, the current local date and time is calculated and displayed on the console followed by the time deviation value. Example:

```
>>> hh:mm:ss MMM dd,yyyy    TIMEDEV = +01.00
```

If the local date and time is lower than the date and time at the end of the previous GCOS 7 session the following messages are displayed:

```
>>> PREVIOUS LOCAL DATE : hh:mm:ss MMM dd, yyyy    TIMEDEV = shh:mm
>>> CURRENT LOCAL DATE  : hh:mm:ss MMM dd, yyyy    TIMEDEV = shh:mm
      IL70 SYSTEM WILL BE FROZEN
      IL70 DO YOU AGREE (Y OR N) ?
```

The s is either - or + . The answer Y (yes) indicates that the operator agrees not to have the local time set back (An After Journal exists on the site.): **the system will be frozen** until the current local time passes the previous local time. The answer Y is forced in AUTO mode. Any other answer implies that the current local date and time is acceptable for starting a new session.

The operator may update the time deviation later on, at IL01 OPTIONS question through the TIMEDEV option or during the GCOS 7 session through the MDTIME operator command (Refer to Chapter 8 *Description of the Operator Commands*).

When the time deviation is updated from the GCOS 7 side (IL01 or MDTIME) the new value is sent and registered into the service processor if the hardware model allows it.

When the time deviation is updated by the service processor operator the new value is communicated to GCOS 7 only for display on the system console. It is not taken into account until the next reloading.

At the IL01 OPTIONS question it is also possible to change the GCOS 7 date and time through the DATE option. This new date will be valid only for the current session, it is neither kept from session to session nor transmitted to the service processor.

Each time the DATE or the TIMEDEV option or both are keyed in at IL01 OPTIONS, the system checks that the new current local time is not lower than previous local time, as is described above. **When the system freeze is rejected (answer not equal to Y at IL70) the IL01 OPTIONS question is asked again in order to update the date or time deviation values or to continue the processing by pressing "Enter".**



During a GCOS 7 session (from SYSTEM READY to END OF SESSION or CRASH), the operator may modify the local time through the MDTIME command (time deviation modification). It is not possible to modify the reference time at the GCOS 7 level.

When the SP time-of-day device is unavailable or failed the IL92 message is displayed on the system console (IL92 UNABLE TO GET CLOCK FROM SP), then a new local date and time is calculated depending on the system context. The algorithm is summarized in the following table:

Table 3-1. Time and Date Algorithm

	RELOAD and Previous Status	RELOAD and No Previous Status	HW INIT and Previous Status	HW INIT and No Previous Status
GCOS 7 Reference Date & Time (TDO)	TOD_CPU register	TOD_CPU register	Previous TOD (TOD_Prev)	2/1/81:00/00
GCOS 7 Time Deviation (TDEV)	Previous time deviation (TDEV_Prev)	00:00	Previous time deviation (TDEV_Prev)	00:00
GCOS 7 Local Date & Time	TOD_CPU + TDEV_Prev	TOD_CPU	TOD_Prev + TDEV_Prev	2/1/81:00/00
CONSEQUENCES	None	Local date & time = reference date & time	Local date & time is late : (a few minutes)	Wrong local and reference date & time

TOD_CPU Time-Of-Day register in Central Processing Unit (valid after reload function)

Prv.St. Previous status : it gives information stored at GCOS 7 session termination time.

TOD_Prev Reference Time (TOD) stored in the previous status

TDEV_Prev Time deviation value stored in the previous status

WARNING : on DPS 7000/XTA the date and time displayed at the right side of the Windows™ tool bar is the Windows™ local time. It is not reference date and time. This reference is internal to the system. It must be expressed in Universal Time (UT) and can only be set through a BIOS command.

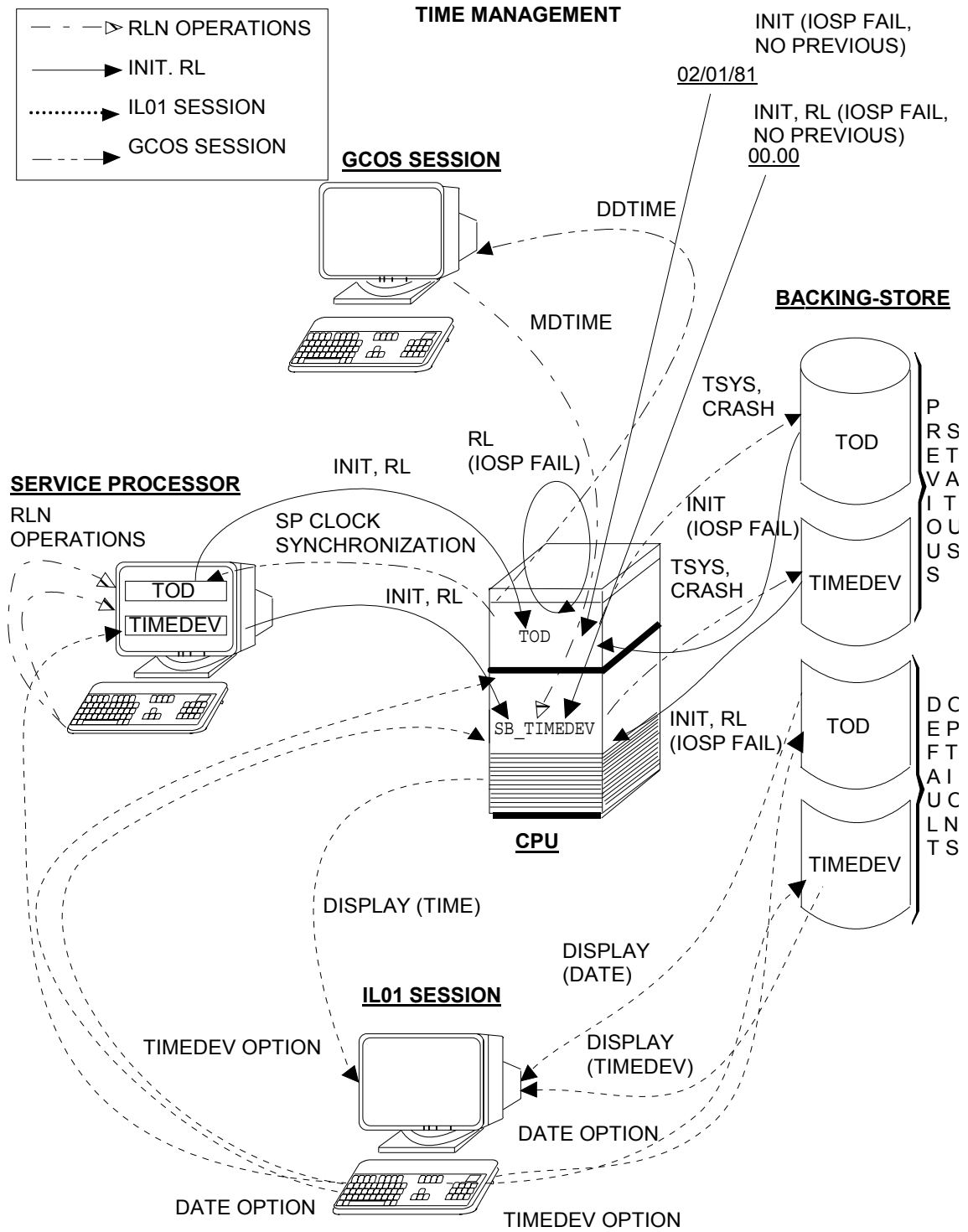


Figure 3-1. Time Management



3.4.5 Device Testing and Volume Recognition at GCOS 7 Restart

3.4.5.1 General

After the dump has been taken and/or the restore has been made (if necessary), ISL is terminated and GCOS 7 becomes active.

The initial AVR (Automatic Volume Recognition) phase takes place. The state of all devices of the configuration is tested. Messages are displayed for all READY storage devices (disks, tapes, cartridges, diskettes) and printers.

DV21 AVR messages are sent to the operator console, except for standard MIRROR disks (see below):

message for disks

```

_____
                                     { [NVTOC] }
                                     { SYSTEM NVTOC }
                                     { BKST [NVTOC] }
                                     { SRV NVTOC }
hh.mm dvmn PREMOUNTED { volnm { NSTD [FBO] } }
                                     { NSTD MIRROR }
                                     { NSTD }
_____

```

messages for tapes and cartridges

```

_____
hh.mm dvmn PREMOUNTED { volnm } [AUTO] attributes [PROTECT]
                       { NSTD }
                       { "WORK" }
_____

```

where attributes are built from tape or cartridge device attributes.



messages for printers

```
hh.mm dvnm PREMOUNTED volnm
```

If at least one sharable disk device is seized by the other system, the following message is displayed:

```
DV34 GCOS: dvnm AVR IN PROGRESS: TRAFFIC WILL RESUME
```

The lists of resident and shared disks are processed. Possible error messages have been described in section 3. When the DV43 question is asked, it must be answered as explained below in Section 3.4.5.2 *DV43 Question*.

When no error is detected, resident and shared disks are allocated to X1 (system). Likewise, BKST disks are allocated to X1, including the SYSTEM disk. In addition, on DPS7000/An systems, the service disk is allocated to X1.

MIRROR disks are then processed, if any exist on the site (system configured with MIRROR = YES). Pairs of mirrored copies are identified and checked; resynchronization, automatic refilling or copy invalidation may take place at that time. AVR messages are sent to the operator console:

messages for MIRROR disks

```
MR01 dvnm PREMOUNTED volnm MIRROR { PRIM [ RESYNCHRONIZED ] }
                                     { SEC [ RESYNCHRONIZED ] }
                                     { ALONE }

MR01 dvnm PREMOUNTED volnm MIRROR { WILL BE COPIED }
                                     { WILL BE REFILLED }
```

Refer to the *Mirror Disks User's Guide* for further details.



3.4.5.2 DV43 Question

When the DV43 question has been asked:

```
0/1 DV43  REPLACE BY:
```

the system is waiting for an answer and the operator must:

- issue a **BREAK**,
- at "S:", enter the system question number (01 in the above example),
- enter the correct list of either shared or resident volume names, separated by spaces or commas, and press **ENTER**; if the new list is empty, then **ENTER** is to be pressed directly after the system question number.

The DV43 question is asked after:

- DV37 GCOS: NUMBER OF RESIDENT EXCEEDS 10, NEW LIST IS REQUIRED.
- DV05 RESIDENT MIRROR FORBIDDEN:
 <volnm1> <volnm2>...
- DV40 RESIDENT NOT FOUND:
 <volnm1> <volnm2>...
- DV41 SHARED NOT FOUND:
 <volnm1> <volnm2>...

3.4.6 System Ready

After the Restart session is completed, the system becomes **READY**, and the message:

```
hh.mm GCOS - RELEASE V7 XXXX READY mmm dd, yyyy
*****
```

is displayed. This means that Storage Load is now completed and that GCOS 7 is fully operational. Most of the service jobs are started. Some jobs may be in the **IN SCHEDULING** state if a Warm Restart was requested.

They are started as described under "Job Scheduling and Execution" in Chapter 5 *Running User Jobs Under GCOS 7*.

Some jobs may be **EXECUTING**, if the previous System Session was terminated by a **TERMINATE_SYSTEM** command (**TSYS**). Jobs which were interrupted between steps are now resumed.

Some outputs may be **READY** and the Output Writer must be started using the **START_OUTPUT_WRITER** command (**SOW**).

New jobs can be introduced by using the Input Reader or the **ENTER_JOB_REQUEST** Command (**EJR**).



3.4.7 Example of Answer to the IL01 OPTIONS? Question

```
IL47 WARNING MEMORY HAS BEEN CONFIGURED

>>> ISL_142  15  -01 (13) / 2
>>> 11:18:17 JUN 25, 1992  TIMEDEV = + 02:00
>>> LOAD PCLC = 1502

SYSTEM PREVIOUS STATUS ON 6/25/92 AT 11:16:37
SYSTEM RESTART
*  IL01 OPTIONS?
I:
    OP65 NO RECORDED REPLY FOR SYSTEM QUESTIONS

    11.19 MS93  PREMOUNTED HIT193 BKST NVTOC
    11.19 MS92  PREMOUNTED HIT192 SYSTEM NVTOC
    11.19 MS97  PREMOUNTED HIT197
    11.19 MS94  PREMOUNTED HIT194 NVTOC
    11.19 MS28  PREMOUNTED FSD528 NVTOC
    11.19 MS17  PREMOUNTED FSD517
    DV34 GCOS: MS95 AVR IN PROGRESS: TRAFFIC WILL RESUME
    11.19 DK11  PREMOUNTED MD2 SZ256/DCE/MD
    CL60 NUMBER OF SYS.SPOOLS : 3
    TW01 GENERATION NOT PERFORMED
    11.21 X297.1  ABORTED LAEH OPERATOR Z SEV6=61000
    11.21 X298.1  ABORTED FEPS OPERATOR S SEV6=61000
    RS06 X299 REACTIVATED
    11.21 X301.1  ABORTED WRITER OPERATOR W SEV6=61000
    11.21 X307.1  ABORTED IOF OPERATOR Z SEV6=61000
    11.21 X306.1  ABORTED RAEH OPERATOR S SEV6=61000
    11.22 PR11  PREMOUNTED 0000
    11.22 GCOS7 RELEASE V6100 6146 READY JUN 25, 1992
    *****
```



3.5 Terminating the Session

3.5.1 Normal Termination

A session is normally terminated by using the `TERMINATE_SYSTEM` (TSYS) command, whereby all the system's activities are stopped at the end of the session.

The `TERMINATE_SYSTEM` command is refused if there are any jobs which are suspended (during a step execution) due to operator action (`HJ-HOLD_JOB`). As soon as the `TERMINATE_SYSTEM` command is accepted, no more jobs are scheduled and the commands `HJ/RJ-RELEASE_JOB`/`FJ-FORCE_JOB` are refused.

A normal termination means that:

- No step is executing; a `TERMINATE_SYSTEM` command causes all executing jobs to be suspended at the end of their current step or to be aborted at the next step checkpoint.
- No job is suspended in a step.
- The Output Writer is stopped. All `WRITER` service jobs are terminated at the end of their current outputs.
- The system files are closed.

NOTE:

Outputs of jobs remain in the `SYS.OUT` file to be output at the next session.

After you enter the `TERMINATE_SYSTEM` command and shutdown is completed (indicated by message `AV02`), you can either power down the installation or reload the system.

The `TERMINATE_SYSTEM` command used to shut down the system may now be submitted by either:

- an operator using a system console,
- a network operator.

For the DPS 7000 an optional parameter (`INIT`) has been added to the `TSYS` command to request that a hardware initialization occur before the next `ISL` (useful for changing the system technical status). You may also predefine the identifier of the device from which the system will be loaded at the next session.

Further, the `TSYS_GCOS_PWROFF` command not only terminates the system, but also powers-off the hardware. (Not available on DPS 7000/2X0/3X0 or DPS 7000/XTA series). Refer to Chapter 8 *Description of the Operator Commands* for a description of the `TSYS` (`TERMINATE_SYSTEM`) command.



On DPS 7000/XTA series, the TSYS power off command terminates GCOS but the power off action is not available. Power off can be done using Windows™ shut down command.

The SHUTDOWN procedure automatically resets the reconnection time to zero. You may define a new GCL command on-site to include specific commands before issuing the TERMINATE_SYSTEM command. For example, you may process the accounting file, or shut down telecommunications.

Shutdown involves two major phases:

1. Enter the TERMINATE_SYSTEM command. The message SHUTDOWN IN PROGRESS is sent to all operators. From then on, no new job can be scheduled and those jobs being executed are held at the end of their current step. Not even main operators are authorized to log on under IOF. IOF users who are still logged on at system consoles are disconnected. IOF users logged onto the network are also disconnected when you enter the TERMINATE_TELECOM_SERVER command (TTSVR).
2. Disconnect the other operators still logged onto the network by entering the TT or TTSVR STRONG command. Before entering BYE, check that all the shutdown conditions are satisfied because you will not be authorized to log on again. Finally, the OMH job is shut down.

3.5.2 Abnormal Termination

If an error is detected which cannot be bypassed, the system crashes. The following message is displayed:

```
-----  
* AV01 GCOS: CRASH. { [PARTIAL] DUMP AVAILABLE }  
                   { REDUCED DUMP           }  
                   { NO DUMP                }  
                   { TRUNCATED DUMP        }  
-----
```

This is the first line of a three-line message; the second line may be one of the following:

```
AV01 CHANNEL EXC. ON MCnn pcpp  
AV01 IOP EXCEPTION ON iopnm  
AV01 REASON : HARDWARE FAILURE  
AV01 REASON : SYSTEM EXCEPTION  
AV01 REASON : CALL TO PANIC  
AV01 REASON : SYSTEM RESTART  
AV01 REASON : EMERGENCY SHUTDOWN
```




while the third line is:

```
AV01 PA xxxxxxxx xxxxxxxx xxxxxxxx
```

NO DUMP indicates that the system was unable to store the dump on disk. It may indicate a problem related to either the system disk, the disk unit, or the Mass Storage Processor. Refer to the appropriate *Operator Guide* or *Operator Reference Manual*.

The second message line gives the reason for the crash, and the third provides additional information (crash parameters).

To shut down the system as cleanly as possible and take a dump, issue an SR command under SPOS. Data updates are stored and are used at the next warm restart to restore the files, jobs and outputs.

After a system crash, the installation can be powered off, or processing recommenced.

In all cases, refer to the appropriate *Operator Reference Manual*.



IMPORTANT:

NEVER, under normal circumstances, terminate the system by using the INIT/SYSTEM INITIALIZE/RESET or POWER OFF pushbuttons or the POWER OFF command.

When the system stops due to failure or discharge of the back-up battery, the following messages are sent:

```
AV01 EMERGENCY SHUTDOWN  
AV01 POWER FAILURE
```

When the system stops due to failure of the fan, the following messages are sent:

```
AV01 EMERGENCY SHUTDOWN  
AV01 FAN FAILURE
```

When the system stops due to overheating, the following messages are sent:

```
AV01 EMERGENCY SHUTDOWN  
AV01 EXCESSIVE TEMPERATURE
```

If sufficient charge remains in the battery, a control dump is taken first.



3.5.3 Information Left Over After Termination

After system termination, information remains in the system files about the following:

- Jobs known to the system,
- Input enclosures associated with these jobs,
- Output prepared by (or for) these jobs.

Some information remains in the SYS.LIB system file.

The system dump, if there is one, is stored on the backing store file SYS.BKST.

After an abnormal termination (crash) the system context is partly saved in BKST and partly kept in main memory. The a reload is automatically forced in order to process the system dump (refer to Section 3.4.3.5 *DUMP*). The error context is rebuilt using BKST and the main memory part that is not damaged by the reload mechanism.

In the event of a system shutdown, i.e. termination of a session by means of the TERMINATE_SYSTEM command, all system files are closed. In any other case they are left open, as are the user files, to be recovered by File Salvager if a warm restart is requested at the next storage load.

3.6 Error Messages Displayed During System Loading

IL type, AV type, DV type and VM type messages may appear at the system console during GCOS 7 loading. For a complete explanation of these messages, see the *Console Messages Directory*.





4. Operating under GCOS 7

4.1 Introduction

After the system is initialized, programmed operator applications exist to perform the following functions:

- adjust system parameters depending on such conditions as date, time, system load, and urgency of jobs;
- schedule jobs;
- correct errors and activate preventive maintenance;
- check system performance, detect lack of resources, plan capacity, and tune the system;
- provide job accounting;
- collect statistics periodically and provide specialized displays.

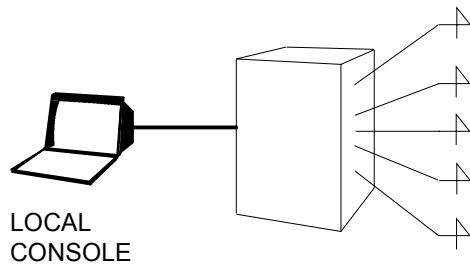
You can perform these functions either automatically or manually from the following workstations:

- local system consoles.
- remote terminals if a link is established;
- remote terminals connected to another DPS 7000 (known as system control center) by use of the primary network. In this case, the operator may control several systems simultaneously.

At any time, one or several of these workstations (up to 16) may assist in operating a DPS 7000 System. For example, a local console may receive media mount messages; a remote terminal may be dedicated to the TDS while another remote terminal may receive all urgent messages such as errors, aborts and alarms for exceptional events. If several satellite systems are connected to the same network, you may control them from:

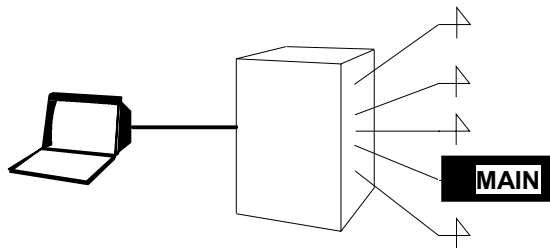
- either a central console;
- or several consoles.

For example, in the latter case, you may group all application messages on one console and route all operator requests and urgent messages to another.



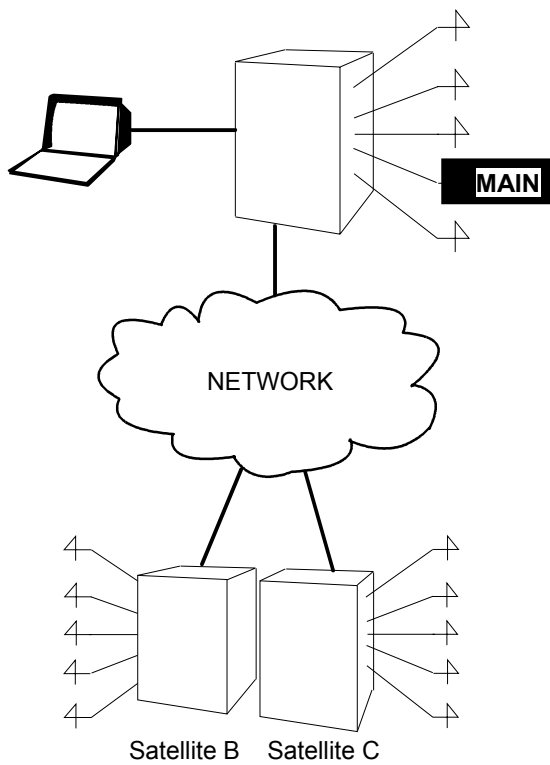
①

The simplest configuration. The system is controlled by the Local Console. The "flags" on the right symbolize terminals connected through a communications processor.



②

Added to the local consoles, one of the telecommunication terminals is logged on with the MAIN attribute, giving it access to MAIN OPERATOR commands. (Not to be confused with the back-up local consoles present on redundant systems).



③

Systems B and C are satellites of system A. In this way, a number of computers, located in different places, can be operated from a single centralized control room.

Figure 4-1. Control Consoles



4.2 Description of an Operator Workstation

The operator may use system consoles in line mode or IOF terminals in line or formatted mode.

The GCL command language now includes the privileged MAIN operator commands.

On a synchronous terminal, you may take full advantage of GCL menus, prompts and helps in formatted mode. Color terminals are supported.

On hardcopy terminals as on system consoles, you may submit commands using GCL menus, prompts and helps in line mode.

When you log onto the system, your initial profile is

```
GCLMODE = GCL
OPERATING MODE = CONS
GCLFORM = LINE
NOVICE = 1
MENU = 1
```

To facilitate operating in CONSOLE mode, it is recommended that you program an auto-transmit break function key on your terminal.

The main operator may find it easier to receive messages by setting:

```
PAGEMODE = 0.
```

As a MAIN operator you must not activate interactive processors. In interactive mode you might generate a mount or a recovery request. Because these requests could not appear on your terminal, deadlock would occur. Another reason why you must not execute in interactive mode is that the main operator jobs are executed with a high dispatching priority which means you delay the execution of user jobs.

If a MAIN operator wishes to work under IOF, he must be registered under two separate projects in the catalog: one with the MAIN attribute, the other without it. Under the second project name, he may use interactive processors, for example, to scan outputs.

To build a specialized workstation, open filters. Select the messages to be displayed on your terminal by submitting filter commands. You are advised to store filter commands in your startup sequence to avoid having to enter them at each session. For more information on filters, see Section 4.7 *Filtering System Messages*.

Each logged-on operator works under a given PROJECT. Depending on the set of commands the system administrator grants to your project, you can access all or a subset of the privileged commands. Each installation may define its own GCL environments and the commands you may use within these environments.



The normal procedure is that the operator logs on under the IOF application with a project having the MAIN attribute. Once the system level prompt S: or the main menu appears, you may introduce commands.

The Remote Multiplexed Operator Support (RMOS) enables you to monitor one or several sites using the same operator terminal without having to log on/off. For more information, see Chapter 9 *Remote Multiplexed Operator Support (RMOS)*.

GCOS 7 scans and adapts the I/O operations involving the main operator consoles.

Thus software enables main operator consoles to:

- communicate with terminals in formatted mode;
- communicate with TTY-like terminals in unformatted mode;
- substitute the default character (.) for certain unprintable characters, in particular escape sequences;
- group messages if the operating mode is interactive.



4.3 Connection/Disconnection

Begin by entering a break on your console or terminal. When the system prompts you, enter the proper password and identify yourself to the system.

The installation startup and the main operator startup are different when compared to previous releases:

- When the system is ready, a system startup is automatically executed. Its purpose is to set up installation-specific parameters and initialize important activities, for example, start telecommunications. This is useful when the main operator of a remote controlled site logs on through the network.
- Like any IOF user, the main operator has one or two startup levels:
"USER", "PROJECT" (or "SITE").

The user startup level allows a very specific processing context to be established - for example by submitting filter commands. The project level startup specifies project level options.

A site operator can become a main operator by logging on a console or terminal, provided that he has the default attribute MAIN or that he specifies a project that has the MAIN attribute in the site catalog.

NOTE:

To activate an interactive processor, do not log on under a project having the MAIN attribute, for example, in the case of the system administrator who wishes to activate an interactive processor. If the SYSADMIN project has the MAIN attribute, the MAINTAIN_CATALOG functions can be done only in batch mode.

It is possible to activate an interactive processor from a system console provided that the current user does not belong to a project with the MAIN attribute. For example, on a remote controlled site, IOF users may use local consoles if no main operators are connected.

A main operator can still log on in the same way as in previous releases. For sites where several operators need to be logged on simultaneously, the multi-console option may be purchased.

A main operator may log on and off without affecting system operations. Similarly, a console failure or an off-line transition does not affect the system which can work in unattended mode.

The WAITS FOR RECONNECTION state does not apply to MAIN consoles. The context of the previous telecommunication session is lost if you log on again.



4.4 Distributing Tasks

Read this section if the multi-console option was purchased for your site. Remember the multi-console feature enables several main operators to log on simultaneously.

What characterizes an operator session?

- a command environment,
- operator attributes,
- filter sets.

4.4.1 Command Environment

The environment is a set of commands which are accessible to a common set of users. Each operator works under a given project which has a default environment. The system administrator defines these environments and grants different operators access to privileged commands.

You could have the following environments:

- One in which you use commands to set classes, multiprogramming levels and the allocation of non-sharable resources among several operators. A single project could be associated with this environment in which the command "verbs" TERMINATE, FORCE and HOLD would be extensively used.
- One dedicated to handling device operations. In this case, you could not handle jobs during execution unless you use the CANCEL_REQUEST command.
- An environment dedicated to the SYS.OUT and other system files.
- An environment providing access to all privileged commands accessible only by the system administrator.

To add a new command to your environment, for example, if you want periodically to launch file saves to your current working environment, use the GCL MAINTAIN_COMMAND utility.



4.4.2 Operator Attributes

These determine how commands are executed and how messages are issued. You can specify operator attributes at project level in this catalog. Operators may have up to two attributes: MAIN and STATION whereas other end users have the IOF attribute only. Depending on the attribute value sent to the system, the system executes commands differently. For instance, a STATION operator can access only the objects belonging to his station whereas a MAIN operator can access all objects in the system. The MAIN operator attribute is important for the reception of system messages. See Section 4.7 *Filtering System Messages*.

4.4.3 Filter Sets

By using the CREATE_FILTERSET command, a main operator may specify the class of messages that he wishes to receive. On a site where several operators are active simultaneously, a filter set defined by a main operator can be assigned a higher priority relative to filter sets defined by other main operators. If a main operator with the highest priority for a certain class of message logs off, further incoming messages in this class are automatically sent to operators with the next highest priority for this class of message. See Section 4.7 *Filtering System Messages* for more information on filters.



4.5 System Functions

4.5.1 Backing Store Files

Since the size of the main memory is not infinite, the system uses backing store files to contain information and temporarily to store executing programs not being processed by the system. The contents of the backing store files are mainly:

- The static segments of the system.
- Permanent information (permanent means kept from one RESTART, COLD or WARM, to another):
 - Pre-initialized load modules.
 - Sharable modules.
 - Checkpoints and repeatable user programs (maintained at WARM RESTART).
 - Permanent Virtual Memory Files (VMF) used by the system.
- Temporary information
 - Non-repeatable user programs.
 - Dynamic and temporary segments of the system.
 - Temporary Virtual Memory Files used by the system.

When several backing store files are used their names and their contents are as follows:

SYS.BKST	static segments of the system and temporary information (except TVMF)
SYS.PVMF	permanent Virtual Memory Files of the system
SYS.LIB	permanent information as explained above (except permanent VMF).
SYS.TVMF	temporary Virtual Memory Files of the system
SYS.BKST 1...15	temporary information, as described above (except TVMF)
SYS.PVMF 1...15 }	
SYS.LIB 1...15 }	additional files, as required.
SYS.TVMF 1...15 }	

Every backing store disk must be mounted when ISL is performed. Since SYS.PVMF and SYS.LIB contain only permanent information, a change of the disk cannot be performed at a warm restart.



NOTES:

1. The minimum size of SYS.PVMF is 30 cylinders of FSD 500, irrespective of the configuration.
2. Any change to the SYS.LIB file, which can only be done at ISL time, has the following consequences:
The permanent information has to be re-established.
The system sharable modules are reloaded (automatically performed by the system).
The user sharable-modules (if any) must be reloaded.
The PLM must be reissued.
3. SYS.BKST always increases during the life of a system. The life of the segments belonging to a system, and which are not deleted, is the life of a GCOS session. Pages are swapped on BKST depending on the system load. Thus if more of the system is loaded, the BKST increases further. The BKST corresponding to segments of J which is finished is restored.

4.5.2 Main Operator Console Log

All system operations may be logged in a system file called SYS.LOGC. If there is more than one console using the MAIN operator privilege, the SYS.LOGC file is the only place where all dialogs between the various operators and the system are registered. Its records are therefore useful for solving problems concerning system operations. It is activated after the ISL is completed and contains messages sent to main operators as well as operator dialogs including commands and associated replies.

NOTE:

The messages sent explicitly to a main operator are not recorded in the SYS.LOGC if this operator is not logged-on.

The START_CONSOLE_LOG (SCLOG) and TERMINATE_CONSOLE_LOG (TCLOG) commands manage this file. SCLOG must not be issued while TCLOG is being executed and vice versa.

The SCLOG CLEAR option makes the SYS.LOGC file available to the system by erasing its contents and then starts logging the new session. TCLOG terminates the logging and makes the SYS.LOGC file unavailable for GCOS 7; SCLOG restarts the logging.

We recommend that the SCLOG, SCLOG CLEAR and TCLOG commands be used only for recovery after incidents.



If during the logging of the Main System Session the hard copy fails, this does not result in the loss of information.

The use of the SYS.LOGC file enables a site without hard copy to record the system session. The DISPLAY_REQUESTS (DR) command allows the system to list all the messages for which operator intervention is required. This is very useful for a site that has only a screen or for a site that has a very large number of messages.

It is very important to have enough space in the SYS.LOGC file because it may be overwritten and information lost. When the file is half-full the following messages are displayed:

```
EL01 SYS.LOGC FILE 50% FULL
* EL03 RUN PRLOGC
```

We recommend that at this point the PRLOGC utility be run to print the SYS.LOGC file and free the space within it.

The full syntax and description of the PRLOGC utility are given in Appendix B: Calling System Utilities.

PRLOGC may only be started if the SYS.LOGC file is made available by GCOS 7. If the TCLOG command has been issued, the utility PRLOGC is not executed.

If the SYS.LOGC file is full and PRLOGC has not been executed on it, then the following messages are displayed.

```
EL01 SYS.LOGC FILE FULL
      DATA LOST UP TO date
```

This means that all information that was in the file up to "date" has been lost. If you have problems with the current SYS.LOGC file, you may switch to another as follows:

- stop the current SYS.LOGC activity by TCLOG,
- rename the current SYS.LOGC file,
- allocate another SYS.LOGC file,
- restart the SYS.LOGC activity with the new file by SCLOG CLEAR.



Certain abnormal situations may occur which result in the following messages being displayed:

EL01 SYS.LOGC FILE REINITIALIZED

This means that the SYS.LOGC file has been reformatted after a device failure or had been initialized on another system disk (disk duplication).

EL01 SYS.LOGC FILE UNAVAILABLE

This means that the SYS.LOGC file cannot be accessed due to a hardware failure, for example a bad track. This message can also appear if SYS.LOGC has not been pre-allocated (reserved) at the time of installation.

Any main operator may scan the contents of SYS.LOGC on his screen by using the DISPLAY_CONSOLE_LOG (DCLOG) command. You can retrieve data by giving either its value or its date/time. Moreover, SYS.LOGC may be scanned either forwards or backwards on a page mode basis.



4.6 Console Message Handling

You may reduce console traffic by routing only essential message to an operator's console.

Important messages are of two types:

- Request messages (class R) which require an operator reply or action.
- Class M messages which require manual action.

To receive only important messages, use filter type criteria R in the CRFLT command.

Messages are issued to those operators with the appropriate filters. If the number of operators on-line changes and the filters created vary, then these changes are taken into account at the next repetition of a repeatable message.

If the message table becomes full, new class R messages are queued until space is freed.

To find out the number of pending requests, enter the DISPLAY_REQUESTS command. If the request is a question, the requesting job remains in the WAITS FOR OPERATOR REPLY state until an operator replies. If another operator tries to reply to the same question, the message INVALID REPLY IDENTIFIER OR ALREADY REPLIED appears on the operator terminal. If the requesting job is aborted or terminated, the question is also cancelled.

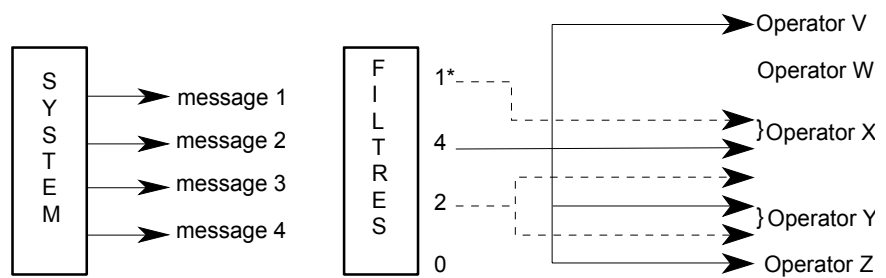
In unattended mode, it is recommended you ensure that as few jobs as possible ask MAIN operators questions and that programs provide replies.



4.7 Filtering System Messages

4.7.1 What is a Filter?

A filter lets certain messages be displayed on your terminal and rejects, or stops others. So you use a filter to select the messages you wish to be displayed on your terminal. See Figure 4-3 *Setting up Filter Sets and Filters*. Consider that the system issues several messages named message 1, message 2, message 3, and message 4. Filters adapt these messages to a particular operator.



* represents the number of receivers.

Message 1 is sent to operator X only.
 Message 2 is sent to four operators: V, X, Y and Z.
 Message 3 is sent to two operators: X and Y.
 Message 4 is sent to no operators.

Figure 4-2. Using Filters

Operator V	uses filters which let message 2 pass through but which reject messages 1, 3 and 4.
Operator W	uses filters which reject all four messages issued by the system.
Operator X	uses filters which let messages 1, 2 and 3 pass through but which reject message 4.
Operator Y	uses filters which let messages 2 and 3 pass through but which reject messages 1 and 4.
Operator Z	uses filters which let message 2 pass through but which reject messages 1, 3 and 4.



Messages can be either specific or generic.

A specific message: the issuer explicitly identifies the receiver who always receives the message.

A generic message: the system issues a message to the generic class of operators who have set the filter to accept the message.

Filters only apply to generic messages. Filter criteria are the following:

KEY - name-8

TYPE

* = all messages,

R = an operator reply, either a manual or automatic action is needed,

M = the operator must take action.

JOBID - the identification of the issuing JOB,

RON - the run occurrence number of the issuing JOB.

By default, if you do not define filters, you will not receive generic messages.

The commands for defining filters are fully described in Chapter 7 *Description of the Operator Commands* and summarized below:

- CREATE_FILTERSET command creates a filter set.
- CREATE_FILTER command creates a filter, that is, defines the message characteristics you want to receive.
- DELETE_FILTER command deletes filter set(s) or filter(s) previously created.
- MODIFY_FILTER command activates or deactivates-activates a filter or a filter set.
- LIST_FILTER command lists filter(s) or filter set(s).

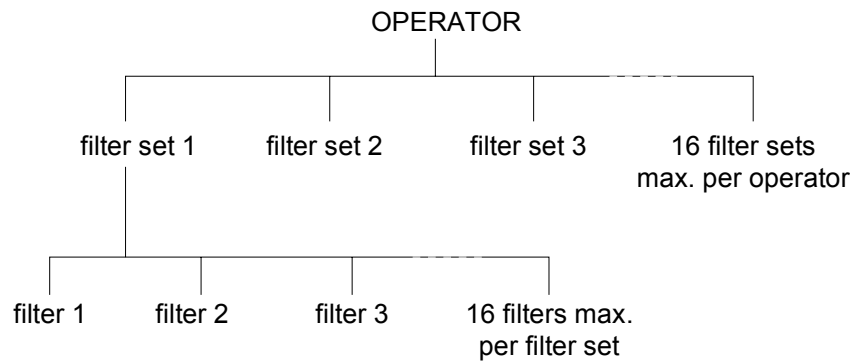


Figure 4-3. Setting up Filter Sets and Filters

A message must satisfy all defined criteria to match the filter.

EXAMPLE 1:

```
CREATE_FILTER A1 A101 I KEY = (AB01,AB99) JOBID = TEST;
```

Defines filter A101 within filter set A1. For a message to be accepted by this filter, its key must be in the range of AB01 and AB99 and must be issued by the job TEST.

CRITERIA

KEY	TYPE	JOBID	RON	Does the message match the filter?
AB99	M	TEST	X45	yes
AB50	R	TEST	X10	yes
DV08	M	TEST	X55	no
AB35	R	APPL1	X99	no

**EXAMPLE 2:**

```
CRFLT A1 A101 I TYPE = M JOBID = APPL1 RON = X32;
```

For a message to be accepted by this filter, its type must be M (message requiring action), and it must be issued by the job APPL1 with run occurrence number X32.

CRITERIA

KEY	TYPE	JOBID	RON	Does the message match the filter?
AB31	M	APPL1	X32	yes
DV59	M	APPL1	X32	yes
OP05	M	APPL1	X32	yes
OP05	M	APPL1	X32	yes
OP05	M	APPL2	X32	no
OP05	M	APPL1	X33	no

4.7.2 Filter Set

Filters are arranged within filter sets. A filter set is a set of filters (maximum 16 filters in a set) defined with an associated priority. Each time a generic message is issued by the system it is submitted to all defined active sets and filters.

A generic message is sent to an operator if:

- this message matches at least one active filter set belonging to this operator. A message matches a filter set depending on the type of filter (inclusive, exclusive, or mandatory).
- the priority of the first matching filter set allows it.



4.7.3 Inclusive, Exclusive, and Mandatory Filters

To match a filter set, a message must match:

- at least one of its inclusive filters (if any),
- none of its exclusive filters,
- all its mandatory filters.

EXAMPLE 1: Select all generic messages

```
CRFLTST A1 0;
CRFLT A1 A101 I TYPE = *;
```

CRITERIA

KEY	TYPE	JOBID	RON	Does the message match the filter?
AB01	M	TEST	X52	yes
AB01	M	TEST	X108	yes
DV31	M	ANY	X305	yes

EXAMPLE 2:

Select all generic messages whose message keys begin with OP.

```
CRFLTST A1 0;
CRFLT A1 A101 I KEY = (OP01, OP99);
```

or

```
CRFLTST A1 0;
CRFLT A1 A101 M KEY = (OP01,OP99);
```

CRITERIA

KEY	TYPE	JOBID	RON	Does the message match the filter?
OP01	R	APPL1	X52	yes
OP99	M	TEST	X108	yes
DV31	M	TEST	X108	no

**EXAMPLE 3:**

Select all generic messages issued by the job TEST with ron X245 where the message key beings with AB.

```
CRFLTST A1 0;
CRFLT A1 A101 I KEY = (AB01,AB99);
CRFLT A1 A102 M JOBID = TEST RON = X245;
```

CRITERIA

KEY	TYPE	JOBID	RON	Does the message match the filter?
AB35	M	TEST	X245	yes
AB35	R	TEST	X245	yes
OP58	R	TEST1	X245	no
OP58	R	TEST	X246	no
OP58	M	TEST	X245	no

EXAMPLE 4:

Select all generic messages issued by the job TEST with ron X245 excluding messages whose key is in the range AB60, AB69.

```
CRFLTST A1 0;
CRFLT A1 A101 I TYPE=*;
CRFLT A1 A102 M JOBID = TEST RON = X245;
CRFLT A1 A103 E KEY = (AB60,AB69);
```

CRITERIA

KEY	TYPE	JOBID	RON	Does the message match the filter?
AB35	M	TEST	X245	yes
AB35	R	TEST	X245	yes
AB69	M	TEST	X245	no
DV35	R	TEST	X245	yes



EXAMPLE 5:

Select all generic messages except those whose key is **DV50**.

```
CRFLTST A1 0;
CRFLT A1 A101 I TYPE = *;
CRFLT A1 A102 E KEY = DV50;
```

CRITERIA

KEY	TYPE	JOBID	RON	Does the message match the filter?
DV50	M	APPL1	X38	no
DV51	M	APPL1	X38	yes
DV52	M	APPL1	X38	yes

4.7.4 Priority of a Filter Set

The priority of a filter set is a digit in the range 0-9. The highest priority is 0 and the lowest is 9.

When several sets, belonging to different operators, match a message, this message is sent to the operators with the highest priority filter sets.

EXAMPLE:

The system issues a generic message which is submitted to active filters and sets.

For the message to be sent to an operator, the following conditions must be fulfilled:

- The message must match at least one filter set;
- Matching filter sets belonging to other operators must have the same priority levels as the first matching set.



OPERATOR	SET	(1) does the message match the filter set?	(2) priority of the set
operator 1	A1	yes	4
	A2	yes	5
	A3	no	3
operator 2	B1	yes	4
	B2	no	0
operator 3	C1	no	4
	C2	yes	8
	C3	no	1
	C4	no	0
operator 4	D1	no	4
	D2	yes	4
	D3	no	0
operator 5	E1	no	0
	E2	no	0
	E3	no	4
	E4	yes	6

The sets of higher priority and which match the message are A1, B1 and D2.
Operators 1, 2 and 4 will receive the message.
Operators 3 and 5 will not receive the message.



4.8 Managing the Network

Communications network management for the front-end processors is handled by:

- TNS for the CNP7 integrated front-end processors in the DPS 7000/An and DPS 7000/2x0 systems,
- OCS (Open Communication Subsystem) for the CP7 Communications processors integrated in the DPS 7000/4xx /5x0 /7xx /8x0 /Dx0 /Mx0 /MTxx /TAxxx /XTA systems.
- FEPS modules for the Datanets configured as front-end processors on the other DPS 7000 systems.

Since the front-end processor provides the DPS 7000 with its communications capability, either TNS or FEPS must be operational before a communications session is available.

Network operations are controlled and monitored by the network control operator, a role which the main operator may assume by default. Commands are entered in GCL mode.

There are three principal categories of network control commands:

1. Normal Operations:

establishing and terminating the communications session:

STSVR and TTSVR on TNS and/or FEPS and/or OCS

starting and terminating the "load" function for the front-end processors:

STSVR and TTSVR on FECM (ACTION=LOAD)

establishing the MCS environment and controlling the MCS application:

STSVR and TTSVR on QMON

SNDQ (SEND_QUEUE)

MDQED (MODIFY_QUEUE_EDIT)

monitoring the resources and functions of the network servers:

DNET (DISPLAY_NETWORK)

DTSVR (DISPLAY_TELECOM_SERVER)

2. Administration

starting and terminating the "system generation" function for the front-end processors:

STSVR and TTSVR on FECM (ACTION=SYSGEN)

controlling and modifying the resources of the network:

STSVR and TTSVR on DSAC

MDNET (MODIFY_NETWORK)

EDD \$NOI (ENTER_DSA_DIALOG) on the NOI mailbox



3. Maintenance

starting and terminating the "dump" function for the front-end processors:

STSVR and TTSVR on FECM (ACTION=DUMP)

starting and terminating traces and debugging facilities:

STTR and TTTR on TNS, FEPS, OCS and UFT

EDD \$DEBUG (ENTER_DSA_DIALOG) on the DEBUG mailbox

DBSVR (DEBUG_SERVER)

executing and closing in line tests:

EXTL and CLTL

printing the trace file:

PRTRF

All these commands are fully described in the *Network User's Guide*.



5. Running User Jobs under GCOS 7

5.1 Introduction

This section consists of the following main subsections:

- **User Jobs: General Concepts**
introduces the main concepts and terminology of GCOS 7.
- **Multiprogramming**
describes the various commands which enable the operator to control the system resources.
- **Job Introduction**
describes how jobs can be introduced to the system, particularly through the Input Reader.
- **Job Scheduling and Job Execution**
defines the respective roles of the operator and the system in controlling the passage of jobs through the system.
- **Job Termination**
describes how a job may terminate.
- **Output Writer**
describes the use of the Output Writer.
- **Automatic Resource Management**
introduces the concepts of Automatic Resource Management (ARM) and summarizes the commands.



5.2 User Jobs - General Concepts

Operators will find that the terms become gradually more familiar, but the broad picture which follows may be helpful:

1. Programmers who send work to be executed on the computer are **users**.
2. A user submits his work in discrete units, each of which is called a **job**. Accounting related to these jobs is performed by the system.
3. A job takes the form of a set of records on a magnetic support, containing a series of **Job Control Language (JCL)** statements. JCL is the language with which the user tells the system the details of how he wants his batch job to be run and what work is to be performed.

The interactive (IOF) user uses the **GCOS 7 Command Language (GCL)** as an equivalent to JCL.

This is fully described in the IOF manual set.

4. The user gives his job a name: the **job identification (jnm)**.
5. Each job consists of one or several **steps**: the system allocates a step number to each step.
6. A step is the execution of a **load module**.
7. A load module is a **compiled program** in a form such that it can be loaded into memory and executed.
8. The whole life of a job in the system is a **run**.
9. The system generates a **run occurrence number (ron)** for each job in the system. These are allocated sequentially with no possible duplication. This ron is the means by which the operator normally identifies a job once it has been introduced. (The ron consists of one to four digits preceded by an "X". The system itself uses X1, and X0 is not allowed).
10. Some of the system functions are done by specific jobs called **service jobs** (e.g., JCL Translator, WRITER). These jobs run in a manner similar to user jobs, but in general they do not display any messages on the system console (see "Service Jobs" in this section).

5.2.1 The Job State Concept

A job passes through various stages between the time of its introduction and the time of its deletion from the system. The system assigns a Job State in accordance with the stage it has reached.

The significance of these Job States is outlined in Figure 5-1, Figure 5-2, and Table 5-1, in the following pages.

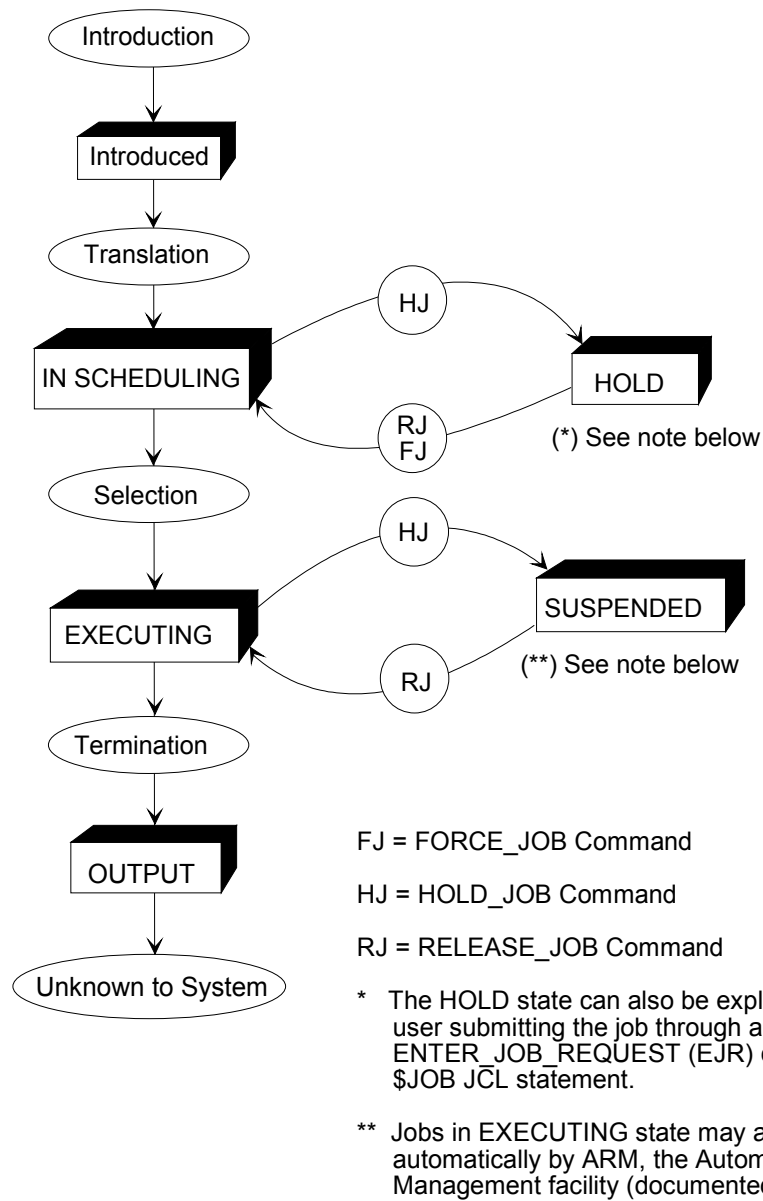


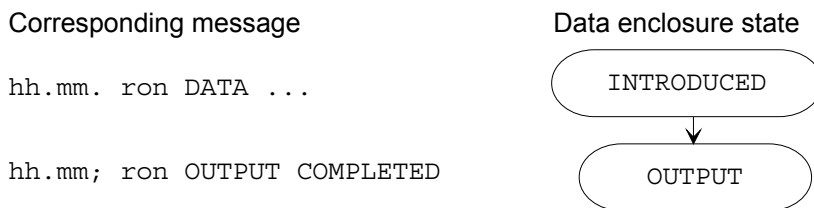
Figure 5-1. The Job State Concept



Table 5-1. Job States

Job	States in the Life of a Job
1. INTRODUCED	A job, introduced by the Input Reader, must first be translated. Until translation is completed the job is said to be INTRODUCED.
2. HOLD or IN SCHEDULING	Once the job is submitted to the system it can be either: a. Held until released by the operator or another job. b. Ready to be selected for execution.
3. EXECUTING	The system selects, from the ready jobs, the one which is to be executed. The selected job is then started.
3b SUSPENDED	While in the EXECUTING state, a job can be temporarily SUSPENDED by the operator or by the system: no further instruction or execution takes place until it is released for execution by the operator or by the system.
4. OUTPUT	Eventually, job execution is terminated, but the job remains in the system waiting for the printing of its outputs to be completed. Once the output is complete the job is deleted from the system.
5. IDLE	Three types of service job (JCL Translator, WRITER and GTP) go into a specific state called IDLE whenever they have no more work to do. They automatically start EXECUTING whenever needed.

In addition to the job concept, the system offers a data entry facility intended to be used in a cardless environment. Data is introduced to the system and moved into user specified libraries without any job execution. No job report is produced unless one is requested or an error occurs. Such data descriptions receive a ron at their introduction:



The OUTPUT state for these rons is only reached when and if necessary.

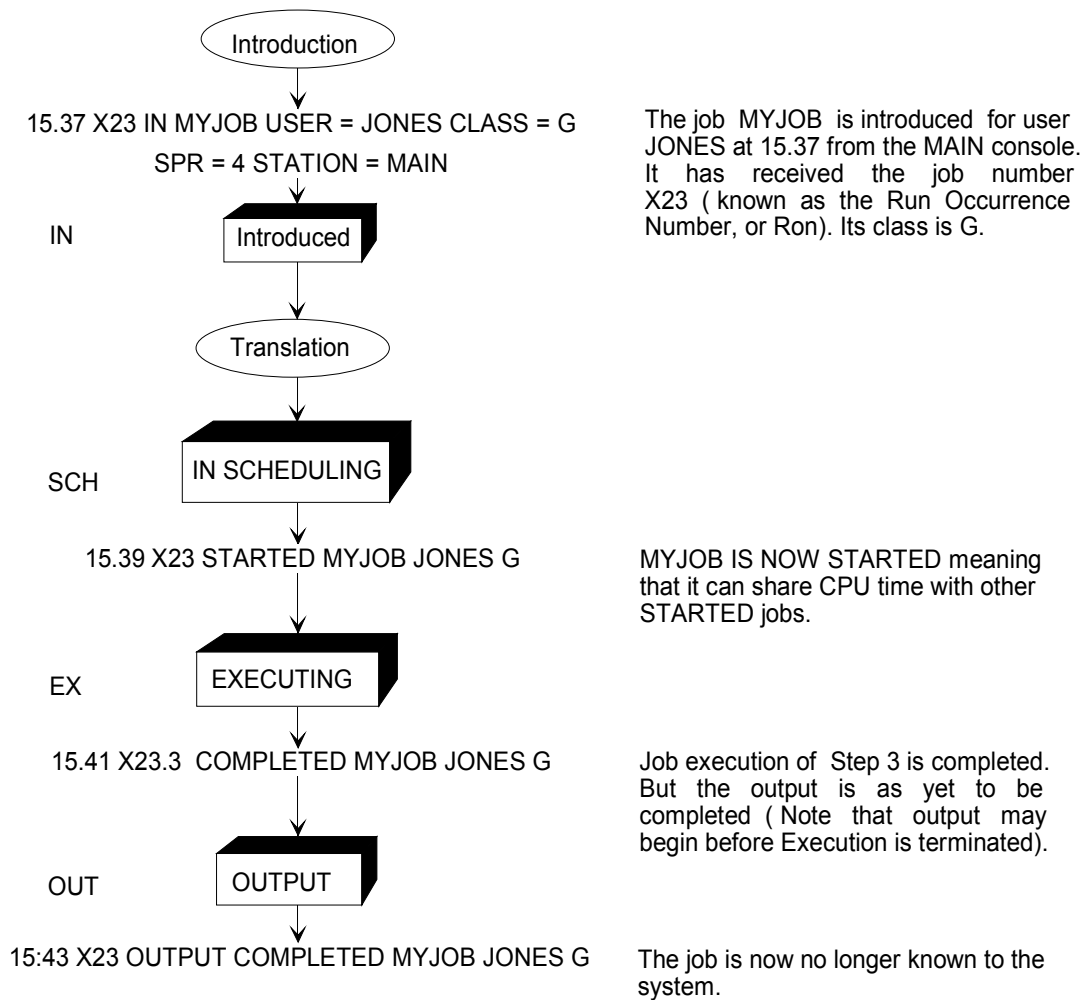


Figure 5-2. Job State Messages

This figure shows some of the messages displayed by the system throughout a job's life. Note that messages relative to HELD and SUSPENDED states are not given here.



5.3 Multiprogramming

The system allows more than one job to be executing at any one time. The number of jobs in execution (or suspended) at any one time is called the System Load. The maximum system load (the maximum number of "J") is defined at system configuration time.

The system load includes user jobs and service jobs. User jobs are batch or interactive jobs, and service jobs are submitted by the system in order to achieve a service function (for example: Output Writer, Stream Reader, JCL Translator).

If a job is suspended, it is still taken into account for the System Load. Service jobs are not considered in the multiprogramming level.

It is part of the operator's duty to monitor and control the system resources used by different jobs in the system by means of the following commands:

Table 5-2. Job Display and Control Commands

Form	Command	Function
DLD	DISPLAY_LOAD	To display information on all jobs in order to monitor the system.
SLD MDLD TLD	START_LOAD MODIFY_LOAD TERMINATE_LOAD	To control user jobs through job class or job class group, scheduling and execution priorities. Some service jobs can also be regulated by commands involving job classes.
FJ HJ MDJ RJ	FORCE_JOB HOLD_JOB MODIFY_JOB RELEASE_JOB	To control user jobs through state, scheduling and dispatching priorities.



5.3.1 The Job Class Concept

Job classes are a means of organizing jobs into manageable groups according to their priority, profile, resource needs, and operational constraints. This simplifies job management and also leads to a more effective use of system resources.

Job classes are identified by a name of one or two characters, A through Z and AA through ZZ.

All have the following attributes:

- a scheduling priority (PRIORITY)
- an execution priority (XPRTY)
- a multi-programming limit
- a state at system startup (STARTED or NSTARTED).

The system operator can:

- use the MODIFY_LOAD command to modify the scheduling priority, execution priority, and multi-programming limit (unless defined as non-modifiable at CONFIG time).
- use the CONNECT_LOAD command and DISCONNECT_LOAD commands to connect or disconnect a job class from a job group.

Standard Job Classes

Classes A through Z plus RB are delivered pre-configured. By default, they are automatically initialized (started) at system startup.

A through Q: are intended for user jobs.
P is the default job class for Batch jobs.
Q is the default job class for IOF jobs.

R through Z, and
RA through ZZ: are reserved for service jobs. They are not available to user jobs.

The default values for scheduling priority (PRIORITY), execution priority (XPRTY), and multiprogramming limit, are shown in the following table, together with the recommended usage.



Table 5-3. Job Classes: Standard Attributes and Recommended Usage

Job Class	Default PRIORITY	Default XPRTY	Default MAXLOAD	Recommended Usage
A-C	7	9	1	batch jobs
D	1	5	1	batch jobs
E	2	4	1	batch jobs
F	3	7	1	batch jobs
G	4	9	1	batch jobs
H	6	0	1	communications
I	7	9	1	GTWRITER
J	6	0	1	TDS
K-O	7	9	1	batch jobs
P	7	9	40	batch (default)
Q	0	1	100	IOF jobs
R	0	1	6	Input readers
RA	*	*	*	Hardware TD_MVH job
RB	**	**	**	Mirrored disk REFILL jobs
S	0	0	12	TNS and/or FEPS (telecom monitoring, QMON for MCS), RAEH, IOSER, ARS, SECCL, OCS, SAV_SRV
T	7	2	6	File transfers FTF, UFT and SERVER2
U	0	1	6	DJP operator's IOF jobs and RBF
V	7	9	1	TRCCL (Trace)
W	0	1	8	Output writers
X	0	0	4	JCL Translator JTRA, High Availability jobs (CMSR, RECOV, and JRU)
Y	7	2	6	GTP, VCAM, JPPC
Z	0	0	2	MAIN operator's IOF jobs, LAEH

* Fixed by OLTD

**Fixed by SYSTEM

User-created Job Classes

These are two-character classes, either created statically by the CONFIG statement JBCLASS2, or dynamically created by the operator command START_LOAD.

Two-character classes (with the exception of the service job class RB) can be deleted using the TERMINATE_LOAD command.



5.3.2 The Job Class Group Concept

A Job Class Group (JCG) is a means of organizing a number of usually similar job classes into a single manageable entity. Just as job classes are a form of job management, JCGs are a form of job class management.

A JCG can be created dynamically using the command `CONNECT_LOAD` (CNLD). Alternatively, it can be configured at CONFIG time, using the `JOBCLASS` statement (mono-character) or `JCLASS2` statement (two-character).

The connection of a class to a Job Class Group can be changed using the Main Operator commands `DISCONNECT_LOAD` (DISLD), and `CONNECT_LOAD` (CNLD).

A Job Class Group consists of a 'parent' job class and one or more 'dependent' job classes. The group is known by the name and attributes of the parent job class. No job class group can be both a parent job class and dependent job class at the same time.

Not all job classes necessarily belong to a Job Class Group, so there may be JCGs, dependent JCGs, and independent JCGs.

The multiprogramming limit of the group is always equal to the multiprogramming limit of its 'parent' job class. Thus, if the total number of jobs currently executing in the group is equal to the MPL limit of the 'parent' class, no other jobs can be started anywhere in the group.

The operator can prevent jobs of a specified class from being scheduled for execution with the `TLD` (`TERMINATE_LOAD`) command.

For more information, see the *Automatic Resource Management (ARM) User's Guide*.



Table 5-4. Job Class Commands

Form	Command	Function
DLD	DISPLAY_LOAD CLASS	To display class or class group status (started or not), current load, multiprogramming level, number of scheduled jobs, number of non scheduled and non executing jobs, priority default values, and whether multiprogramming level is modifiable or not.
SLD TLD	START_LOAD TERMINATE_LOAD	To make each class available or not for scheduling. No job in class held by TLD can start executing unless FJ (Force Job) is used.
MDLD	MODIFY_LOAD	To modify Multiprogramming Level and scheduling and executing priorities, which, if not modified, are retained between sessions. Not effective on classes of jobs already executing.
MDJ	MODIFY_JOB	To alter the class of a job that is held or in scheduling.
DLD	DISPLAY_LOAD CONNECT	To display all the connections between the classes and class groups.

5.3.3 Priorities

Three types of priority are used by the system:

- | | |
|---------------------|--|
| Scheduling Priority | controls the order in which jobs are queued while waiting to be run. |
| Execution Priority | controls the availability of CPU time for each job (step)while executing (also called Dispatching Priority). |
| Output Priority | controls how and when an output can be processed by WRITER. |



5.3.3.1 Scheduling Priority (SPR)

This priority is mainly used with job classes to control the order in which jobs are queued and selected for scheduling (see *Job Class Concept* above).

Jobs are queued in priority order while waiting for translation, scheduling and machine resources. Jobs with the same scheduling priority are handled on a first in, first out (FIFO) basis.

The scheduling priority range is from 0 - 7, with 0 being the highest. The fact that a job is held or its class is not started does not affect the job's position in the queue (apart from jobs leaving the queue to be run or being inserted in the queue).

Operator commands which involve scheduling priority are as follows:

Table 5-5. Scheduling Priority Commands

Form	Command	Function
DJ DS	DISPLAY_JOB DISPLAY_SUBMITTER	Display scheduling priorities.
MDLD MDJ	MODIFY_LOAD MODIFY_JOB	Can be used to change the scheduling priority.
\$JOB	(JCL statement)	Used to give an explicit scheduling priority to a job.
EJR	ENTER_JOB_REQUEST	Otherwise, a job will assume the default priority of its class.
FJ	FORCE_JOB	Used to place a job at the front of the job scheduling queue.



5.3.3.2 Execution Priority (XPR)

This priority controls the sharing of the CPU between executing steps, during multiprogramming. Execution priorities are used to streamline step-execution flow and thus make the best use of the CPU.

The XPR range is from 0 - 9 (0 being the highest). GCOS 7 itself has an XPR higher than 0 and thus can always gain control of the CPU.

Operator commands which can affect execution priority are as follows:

Table 5-6. Execution Priority Commands

Form	Command	Function
DJ DS	DISPLAY_JOB DISPLAY_SUBMITTER	Display execution priorities.
MDLD MDJ	MODIFY_LOAD MODIFY_JOB	Alters the execution priority of a class. To alter the XPR of a job.
STEP EXEC_PG	(JCL statement) EXEC_PG	Used to give a step an explicit dispatching priority. Otherwise it has the default priority of its job class.

Priorities are developed in further detail in Section 5.10, *Automatic Resource Management*.

5.3.3.3 Output Priority

Output Priority (OPR)

This priority is described in Section 5.8.1.5 *Output Priority (OPR)*.



5.3.4 Service Jobs

A service job is a job that the system runs in order to execute certain system functions.

Service jobs are not always running, because they provide temporary functions or because they need to be active only for certain types of work.

The following are service jobs:

- Automatic Restore and Save (ARS)
- Interactive Operation Facility (IOF)
- Input Stream Reader (Reader)
- Disk Move Head T&D (TD_MVH)
- Automatic refilling of mirrored disks (REFILL)
- Queue Monitor (QMON)
- Front End Processing Support (FEPS)
- Front End Control and Management (FECM)
- Transport and Network Subsystem (TNS)
- Networking - Remote Administrative Exchange Handler (RAEH)
- Open Communication Subsystem (OCS)
- Input/Output Server for either Large Memory Cache or Mirror Disks or both (IOSER)
- DJP File Transfer Server (SERVER2)
- Remote Batch Facility (RBF)
- System Trace (TRCCL)
- JCL Translator (JTRA)
- HA Complex Management Server (CMSR)
- HA Takeover File Recovery (RECOV)
- HA Journal Recovery Utility (JRU)
- Virtual Communications Access Method (VCAM)
- Generalized Transfer Processor (GTP)
- TDS Segment Management (JPPC)
- AUDIT7 Security Server (SECCL)
- Local Administrative Exchange Handler (LAEH)

- High Speed Link Server (HSLV7000)

All service jobs are assigned a ron (run occurrence number) when introduced to the system. The job class is exclusive to that particular service job (See Table 5-7 *Service Jobs*).

There can be only one JTRA job executing at any one time.

There is one READER job for each device on which Input Reader has been started.



Table 5-7. Service Jobs (1/2)

FUNCTION	JOBNM	USERNM	LOAD MODULE NAME USED	DEFAULT CLASS	POSSIBLE COMMANDS	COMMENTS
IOF user	IOF	logon	H_IOF plus any system or user load module name	Q	SLD Q TLD Q MDLD Q multi	
Stream Reader	READER	OPERATOR	H_RUN	R	MDLD R multi	class R is always started
OLTD	TD_MVH	OPERATOR	H_MVHEAD	RA		
Mirror	REFILL	OPERATOR	H_UTILITY	RB		
Network and queue management	QMON FECM FEPS TNS RAEH OCS	submitter submitter submitter submitter submitter submitter	H_QMON H_FECM H_FEPS H_TNS H_DSAC H_VCPD7	S		class S is always started
IO management	IOSER	OPERATOR	H_IOSER	S		class S is always started
DJP	SERVER2	submitter	H_SERVER2	T	SLD T ;TLD T MDLD T multi	
RBF operator control box	RBF	logon	H_IOF plus any system or user load module name	U	SLD U TLD U MDLD U multi	
Resource Management	ARS	SYSADMIN	H_ARS	S		class S is always started
Security Audit	SECCL	AUDIT7	H_SECCL	S		class S is always started
System Trace	TRCCL	OPERATOR	H_TRCCL	V	SLD V TLD V	class V multiprogramming level = 1
Output Writer	WRITER	submitter	H_WRITER	W	MDLD W multi	uses IDLE state class W is always started
Dynamic Save Server	SAVE_SRV	SYSADMIN	H_DFSRD_SRV	S	SDYNSVR DDYNSVR TDYNSVR	



Table 5-7. Service Jobs (2/2)

FUNCTION	JOBNM	USERNM	LOAD MODULE NAME USED	DEFAULT CLASS	POSSIBLE COMMANDS	COMMENTS
JCL Translator	JTRA	JTRA	H_EXECUTE	X		uses IDLE state class X is always started. class X multiprogramming level = 4
High Availability	CMSR RECOV JRU	OPERATOR OPERATOR OPERATOR	H_CMSR H_RECOV H_RECOV	X		class X is always started. class X multiprogramming level = 4
DJP	VCAM GTP	OPERATOR OPERATOR	H_JVCAM H_GTP	Y	MDLD Y multi	class Y is always started. uses IDLE state
Segment server for TDS	JPPC	OPERATOR	H_JPPCi i = 1 to 5	Y	MDLD Y multi	class Y is always started
Main Operator	IOF	logon	H_IOF	Z		class Z is always started
Local Administration Exchange Handler	LAEH	OPERATOR	H_OMH	Z		class Z is always started
High Speed Link server	HSLV7000	SYSADMIN	H_HSL_V7000	S		class S is always started

Service jobs do not display log messages concerning run states except in error messages.

Service jobs go through the same states as user jobs, but with certain differences.

GTP, JTRA and WRITER use a specific state called IDLE. They enter the IDLE state when they run out of work. However, as soon as work is available (i.e. transfers to be made, jobs to be translated, or outputs to be printed), they automatically enter the IN SCHEDULING state. The IDLE state economizes the use of system resources and removes the need for operator intervention.

Service-job scheduling is determined by class Scheduling Priority, and the multiprogramming limit for that class. It can be limited by the multi-programming level of a class group attached to the service class.



Service jobs use the scheduling and execution priorities of their class. These can be modified by using the following commands:

Table 5-8. Service Job Commands

Form	Command	Function
MDLD	MODIFY_LOAD	Alters the priorities for a class (except classes S, U, X Z and RB).
MDJ	MODIFY_JOB	Alters the priorities for some service jobs (e.g. READER and WRITER).

Most service jobs never wait for scheduling or system resources. If they cannot start executing at once, they abort and have to be reintroduced. A message is displayed, either on the log or the appropriate terminal, that the job failed to start.

Some service jobs create an output. If the job aborts, a dump of the job is taken. The job remains in the OUT state until the job report and dump have been printed.

Accounting information for service jobs is stored in the accounting file in the same way as for user jobs.

When dedicated to the Large Memory Cache function, H_IOSER is started by the operator command START_IO_CACHE (SIOC). When the system has been configured for Mirror Disks usage (CONFIG MIRROR=YES), H_IOSER is automatically started at GCOS READY.

Refer to *Large Memory Cache User's Guide* and to *Mirror Disks User's Guide* for further details.

5.3.5 The Input Reader

The Input Reader has two main functions:

- To read the input stream (that is, one or more job or data enclosure descriptions).
- To translate job descriptions and put them into the IN SCHEDULING or HOLD state. (Jobs are only put into HOLD, if HOLD was specified as a parameter of either the EJR command or the \$JOB statement, or as result of a HOLD_JOB (HJ) command).

These two functions are performed by the READER service jobs and the JCL translator.



5.3.6 Input Stream Introduction

An input stream may be introduced to the system in one of the following ways:

- an ENTER_JOB_REQUEST (EJR) command,
- "spawning" a job during the execution of another job (by use of the RUN JCL statement),
- from a terminal using either the RBF or IOF service jobs.

5.3.7 The Reader Service Jobs

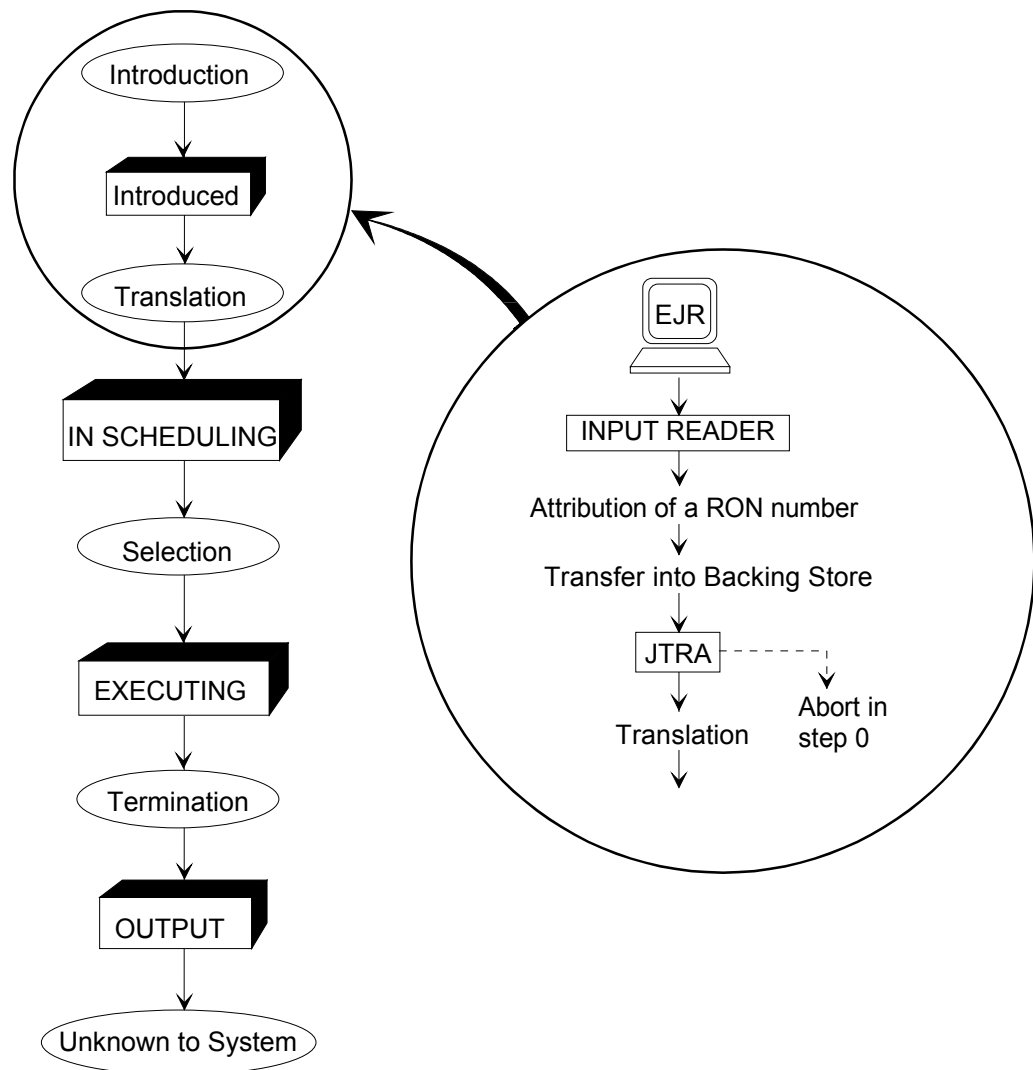
The preceding actions or commands initiate Reader service jobs that read the input stream from the specified file or device.

There may be several Reader service jobs active at the same time although their number is limited to the multiprogramming level of job class R to which they belong.

They can read the input stream from either sequential disk or magnetic tape files, or a member of a disk library.

To improve system performance, pre-initialize the load module activated by the Reader service jobs. This is done by issuing the command:

```
MDC PLM H_RUN.
```



This Figure gives the first phases of the life of a job. The service jobs active during this phase are the INPUT READER and JTRA, (the JCL translator). If translation fails, the job is aborted in step 0. After translation, the job is either IN SCHEDULING, or possibly HELD, if requested by the HOLD parameter of EJR command, for example.

Figure 5-3. Job Introduction



When a Reader has been initialized, by the EJR command, it proceeds in the following way:

1. If it finds that the specified device contains nothing to read, it issues a message asking the operator to mount the volume containing the Input Stream.
2. The Reader then reads the job and data enclosure descriptions and displays a message for each input.
3. When an End of File is encountered, or a TERMINATE_SYSTEM command is received, the reader stops reading at the end of the job it is currently reading and warns the JCL translator that there are jobs to be translated.



5.4 Job Introduction

When a job is introduced the following message is displayed:

```

(INO1)hh.mm ron1 { IN } {ron2 station }
                  { } jmm USER=usernm CLASS=aa SPR=spr { }
                  {HOLD} {STATION=station}

```

The job is given a ron number by the system at job introduction (ron1).

If requested, the job is put into HOLD after translation. The job is not started until a RELEASE_JOB command is issued, or another job executes a RELEASE JCL statement.

Ron2 indicates an input stream spawned by the job ron2 (by the use of RUN JCL statement)

A limit on the number of jobs that the system may know simultaneously is defined during configuration. When this limit is reached, the system displays the following message:

```
IN14 cmdnm dvrnm JOB READING STOPPED: JOB MAX. NUMBER
```

After a TERMINATE_SYSTEM command has been issued, new jobs cannot be introduced to the system (see Section 3.5 *Terminating the Session*).

Service jobs are introduced by specific operator (or terminal user) commands or actions. No message is displayed on their introduction.

5.4.1 Data Enclosure Introduction

An input stream contains job descriptions and data-enclosure descriptions (defined by \$DATA and \$ENDDATA statements).

The data enclosure contains data which is picked out from the input stream and moved into libraries without any job execution being performed.

For identification purposes each data enclosure receives a ron, although this again causes no job execution. A report is produced only if an incident occurs during its introduction, or if specified by the user.

Once a data enclosure has been introduced the operator loses control over it.

Each data enclosure introduction displays the message:

```
(IN02) hh.mm ron DATA MB=mbname IN USER=Usernm station
```



5.4.2 Operator Dialog During Job Introduction (\$SWINPUT JCL Statement)

Some jobs may ask for input using the \$SWINPUT JCL Statement. This should be used with caution and only when the operator is familiar with the jobs which require input in this manner. It is advisable, where possible, to use the EJRC command rather than this feature.

At JCL introduction time, the reader stops and an input is asked for with the following prompt:

```
nn/ron text
```

where text is taken from the \$SWINPUT JCL statement. The operator, taking into account the site information, enters:

```
nn reply
```

The same \$SWINPUT JCL statement may issue several prompts to the operator.

Two forms of \$SWINPUT JCL statements are available:

The first (which uses the ANSWERS parameter) normally asks for a single reply. The reply must belong to a user-defined list of possible replies. If a wrong reply is given then the prompt is displayed again. After three unsatisfactory answers the user job aborts with relevant diagnostics in the JOR.

EXAMPLE:

```
15/X25 TYPE 1 IF UPDATE, 2 IF CREATION  
□
```

The second form of \$SWINPUT JCL statement asks for input from the operator until a user-defined reply terminates the process. (END parameter of \$SWINPUT.)

This second form should be used only in exceptional circumstances, for instance, to enter JCL using a main operator console as a rescue operating mode when the input device is out of order, or if the person at the operator console is the author of the job. It is recommended that the user-defined prompts to the operator specify clearly how to terminate the process, or that some common conventions are used on site. After this input has been sent to the user job, the reader resumes reading.

If the operator has not received clear guidance, or if the user prompts are not clear, he may terminate the reader using a TERMINATE_INPUT_READER (TIR) command with the STRONG option if the job was submitted by START_INPUT_READER (SIR). If the job was submitted by an ENTER_JOB_REQUEST (EJR) with a \$SWINPUT, he should first display information about the jobset request using the DISPLAY_USER_REQUEST (DUR) ALL usernm command, giving a Jobset Occurrence number, Jxxxx. He may terminate the reader using CANCEL_USER_REQUEST (CUR) Jxxxx.



5.4.3 The JCL Translator

The JCL Translator (JTRA) : The JCL Translator is one of the system service jobs. It is always known to the system; it does not terminate but enters the IDLE state. Only one JTRA can be known to the system at any one time.

The JTRA uses class X and it always has a Scheduling Priority of 0.

The JTRA is activated whenever there is an INTRODUCED job present. If more jobs are INTRODUCED while the JTRA is executing, these are also translated in turn, before JTRA enters the IDLE state.

If, after a Warm Restart, there are non-translated jobs in the INTRODUCED state, they cause JTRA to enter the IN SCHEDULING state and then they are translated.

After translation JTRA puts the jobs into the scheduling queue; they go either into the HOLD state or into the IN SCHEDULING state, depending upon the job description.

JTRA does not normally display messages or generate an output. Occasionally messages with a key of "TNnn" may be output, these should be referred to the Service Center. Users may cause a message to be displayed at translation by use of the MESSAGE JCL statement.

If during translation, a JCL error is discovered or a JCL "INVOKED" volume is not ready or is being used by another job, the job in error aborts. Log messages tell the operator when a job has aborted. Refer to the *Console Messages Directory* for the corrective actions. To improve system performance the load module activated by the JTRA service job should be pre-initialized using the command:

```
MDC PLM H_EXECUTE
```

5.4.4 Hold State

After introduction and before execution, a job can be put into the HOLD state. While in this state the job cannot be started unless a RELEASE_JOB command (RJ), or a FORCE_JOB command (FJ) is used.

A job can be placed in the HOLD state on introduction by use of the HOLD parameter in the \$JOB JCL statement or with the HOLD parameter in the EJR command. While the job is in the IN SCHEDULING state the operator can use the HOLD_JOB (HJ) command to put a job or all the jobs into the HOLD state after their translation:

```
HJ ronlist
```

When the list of run occurrence numbers (rons) is omitted, all jobs not executing are held.



A job can be put back to the IN SCHEDULING state by the operator using the RELEASE_JOB (RJ) command or by a job executing a RELEASE JCL statement:

```
RJ ronlist [STRONG]
```

When the list of run occurrence numbers is omitted, all held or suspended jobs are released.

NOTES:

The HOLD parameter in the \$JOB JCL statement may be used in two different ways:

If HOLD is used as a Self Identifying Value, the job is normally released by the operator using an RJ command or by another job using a RELEASE JCL statement.

If HOLD is used as a keyword, it specifies a hold-count (HOLD=hold-count), and it is assumed that the user has introduced a series of jobs and that he wants control of the synchronization of these jobs, without any operator intervention. Each time a RELEASE JCL Statement is used upon that job the current hold count is decrement by one, and when it reaches zero value the job is put into the IN SCHEDULING state. To prevent manipulation errors, jobs using the hold-count facility may not be released by the operator unless a RJ ron STRONG command has been issued, except if the current hold-count is one. Hold-count is decreased by one with the RJ command or the RELEASE JCL statement.



5.5 Job Scheduling

Whenever a job terminates or a new job is introduced, the system tries to start another job. Jobs in the IN SCHEDULING state are queued for selection in scheduling priority order and on a FIFO basis for jobs of the same priority. The system tries to start the first job in the scheduling queue provided that the following conditions are satisfied:

- The job is not in the HOLD state and its associated job class and group class, if they exist, are started.
- The multiprogramming limit for that class has not been reached.
- The multiprogramming limit of any class group attached to the job has not been reached.
- The system multiprogramming level has not been reached.
- The user batch job limit has not been reached.

If the first job in the queue cannot start then the system tries to start the next job, and so on. This is repeated until no more jobs can start due to the above limitations, or because no more jobs are waiting to be started.

A job can be selected immediately with the use of the FORCE_JOB command (FJ). This job is said to be forced.

Forced jobs ignore all restrictions except for the maximum number of started jobs (which is set at configuration time). Thus a forced job may overload the class multiprogramming and the class group multiprogramming levels. In this case no new jobs, dependent upon these levels, start until the levels return to their normal limits.

User jobs display the following message when they start:

```
(JB01) hh.mm ron STARTED jnm usernm jclass
```

Service jobs do not display any message, except for the service job which manages a main operator console.

Operator commands concerning job scheduling are as follows:

Table 5-9. Job Scheduling Commands

Form	Command	Function
DJ DS	DISPLAY_JOB SCH DISPLAY_SUBMITTER SCH	Display the order of jobs waiting to be started.
MDJ	MODIFY_JOB	To alter this waiting order.



In summary, job scheduling tests the following relations:

1. Total number of executing and suspended jobs (both service and user) is less than or equal to the maximum number of jobs.
2. Class group load is less than the class group multiprogramming level.
3. Class load is less than the class multiprogramming level.

Service jobs ignore case 2.

Forced jobs ignore cases 2 and 3.

The priorities are:

1. Forced jobs
2. Released suspended jobs
3. Other jobs.



5.6 Job Execution

Once started, the execution of a job proceeds sequentially through the JCL statements (unless the user, using JUMP statements, has requested some alternative).

The main part of the job execution is the execution of steps, but the user may:

- Introduce new jobs by using RUN JCL statements, which have the same function as the ENTER_JOB_REQUEST (EJR) command.
- Release jobs in the HOLD state by using the JCL statement, RELEASE.
- Send messages to the operator by using the SEND JCL statement. These messages are also shown in the corresponding Job Report (JOR or JOB_REP). These messages appear in the format:

```
SP06 ron msg-txt
```

- Execute operator commands as if the EXDIR statement were entered from the console using GCL statements.

Finally the job execution terminates.

5.6.1 Events Occurring During Job Execution

An event occurring during job execution normally is caused by a device or volume. Other events occurring during execution are reported to the user in the Job Report (prepared by the system).

Occasionally intervention from the operator is required and, when this is the case, he receives console messages to that effect. However some console messages are purely for information and do not require any operation action.

The rest of this Section (5.6) deals with the components of a job run and, where appropriate, certain possible events are also described.



5.6.2 Step Execution

1. Resources are needed for each step to be executed. The job may be queued to wait for a resource, in which case one of the following messages is displayed:

1) A file is being used by another job and cannot be shared:

```
DF01 ron FILE efn volnm USED BY: ron1, ron2, ...
RS04 ron.ssn WAITS FOR FILE efn ON dvnm
```

A job, having the Run Occurrence Number ron waits for a file which is being used by jobs ron1, ron2, ...

2) (Coupled systems): the dual sharing conditions of the file "efn" on a resident volume do not allow access to it:

```
RS04 ron.ssn WAITS FOR DUAL SHARING OF FILE efn ON volnm
```

The corresponding job remains in the WAITING state and is reactivated when the dual sharing conditions change on the other system.

```
DF02 ron WAITS FOR FILE efn RSDT FOR DUAL SHARING CONDITIONS
```

The DF02 message is displayed when the RS04 message has been displayed on the other system.

3) A volume identified by its name (volnm) and characteristics (dvtyp, dvattr, labeltyp) is not available. It is either:

- a) being used by another job and cannot be shared, or
- b) it is a non-removable disk that is OUT or is incorrectly identified, with the result that the step is waiting for a disk that does not exist.

Check dvattr and labeltyp. The label type is specified only if volume labels are non-standard (\$NSTD).

```
RS04 ron.ssn WAITS FOR VOLUME volnm:dvtyp [/dvattr] [labeltyp]
```

4) A device is needed and none is currently free:

```
RS04 ron.ssn WAITS FOR DEVICE dvnm
```

5) More resources are needed in the dimension than are currently available:

```
RS04 ron.ssn WAITS FOR MPL IN DIMENSION dimname
```

6) The step initiation is waiting for a process to be allocated to it. (The maximum number of active processes is decided at system configuration time):

```
RS04 ron.ssn WAITS FOR PROCESS
```

7) Not enough free space exists on the backing-store file to load the step:

```
RS04 ron.ssn WAITS FOR LOADING
```



8) Too many sequence processes are loading non pre-initialized Load Modules simultaneously:

```
RS04 ron.ssn WAITS FOR LOADING PROCESS
```

9) Before Journal is currently fully used:

```
RS04 ron.ssn WAITS FOR BEFORE JOURNAL
```

10) In a DJP environment, the ron is waiting for a remote site:

```
RS04 ron.ssn WAITS FOR SITE sitenm
```

11) No more resources for non-pre-initialized Load Modules are currently available:

```
RS04 ron.ssn WAITS FOR NON PREINITIALIZED LOAD MODULE RESOURCE
```

12) In DJP environment, the ron is waiting for a remote file:

```
RS04 ron.ssn WAITS FOR REMOTE FILE efn ON volnm
```

13) In DJP environment, the server is not available:

```
RS04 ron.ssn WAITS FOR TELECOM SERVER tsvrnm
```

2. The list of jobs waiting for resources can be displayed by using the DJ or DS command.

As soon as the resource becomes available, the message:

```
RS06 ron REACTIVATED
```

is displayed and step execution proceeds. No operator intervention is necessary.

3. The operator can however react to the conflict indicated by message RS04 in various ways. He may:

Prevent other jobs from entering the resource conflict.

Suspend or terminate one of the jobs in conflict, if the conflict is found to be abnormal. This is done by using the MODIFY_LOAD, MODIFY_JOB, HOLD_JOB or CANCEL_JOB commands described in Section 7.

4. Volume mounting may also be requested and execution waits for these mounting operations to be completed before proceeding.



5. The execution of each step may also cause events such as:

Creation of an operator command:

```
OP04 ron.ssn COMMAND: cmdnm CREATED
```

Deletion of a previously created operator command:

```
OP05 ron.ssn COMMAND: cmdnm DELETED
```

Printing of a user message or question. These messages are stamped with the ron of the job:

```
hh.mm USER ron message-text
```

Transformation of a work tape into a normal tape (in particular in the case of dynamic extension of a tape file):

```
FP01 dvmn ron.ssn TAG volnm FOR EFN efn
```

6. The step execution can be temporarily suspended by using the HOLD_JOB (HJ) command (see Section 5.6.6 *Suspended Jobs* below).
7. If requested by an SLD command the following message is displayed when each step starts to execute:

```
JB08 ron.ssn STEP Im-nm XPR=xpr J=jnb
```
8. If the step execution terminates normally, the job execution proceeds. If it terminates abnormally, the job is aborted, unless the step has been declared as being repeatable (particularly with checkpoint restart) or the user has used JUMP JCL statements to take care of the situation.



5.6.3 Step Job Repeat

If a step terminates abnormally and it has been declared as being repeatable, the following question is asked (with a deferred reply):

```

                                [KILLED ]
(SP07) hh.mm ron.ssn [          ] lm-nm [SEV sevnb] [statustp] [ REPEAT]
                                [ABORTED]
                                [FROM CHECKPOINT ckptnb] {Y|N|}, DEFAULT=Y

```

xx	if the deferred reply code must be used in the reply.
REPEAT	<p>In the case of a system crash, the repeat question is sent to the system operator during warm restart. If there is no crash, this question is sent to IOF users with an abnormal job termination. If the IOF user has been disconnected, the job aborts, otherwise the option functions as follows:</p> <ul style="list-style-type: none"> – if the user has declared the execution of the step as repeatable, execution is resumed from the beginning of the step. – if the step is not declared repeatable but the job is repeatable, execution is resumed from the beginning of the job. – if a checkpoint is declared, execution is resumed from the indicated checkpoint.
ckptnb	if checkpoints were taken during the last execution of the step ckptnb indicates the number of the last checkpoint taken. Checkpoint 0 indicates the beginning of the step.
SEV sevnb status	sevnb indicates the severity of the step abort and status indicates the origin. These are summarized in the table below.

NOTE:

The ROLLBACK question is no longer asked; the rollback is automatic if a Before Journal is used.



5.6.4 Pre-initialized Load Modules

Load modules (or "programs") can be transferred to a permanent BKST file. They are then said to be pre-initialized. This speeds up the loading of the load module when it is requested for execution in a job step. When a load module has been pre-initialized, the original copy (in a load module library) is not used.

NOTE:

Pre-initialized Load Modules are lost if a RESTART (CLEAN) is performed.

It is the operator's responsibility to manage a permanent BKST file:

- To find out which load modules are pre-initialized, using the DISPLAY_CONFIGURATION (DC PLM) command.
- To pre-initialize a load module, using the MODIFY_CONFIGURATION (MDC PLM) command.
- To delete a load module from a permanent BKST file, using the MODIFY_CONFIGURATION (MDC CLM) command.

The DC PLM, MDC PLM, and MDC CLM commands are described in detail in Section 7.

Data Management Utilities are used very often, so you may wish to pre-initialize them. The following table gives the correspondence between each utility and the load module it executes:

Utility	Load Module
FILREST	Before TS7254: H_DFSRD_LM
FILSAVE	From TS7254: H_UTILITY
VOLREST	Before TS7254:
VOLSAVE	If FAST option H_DFSRD_LM
	Else H_UTILITY
	From TS7254: H_UTILITY
VOLPREP	H_VOLPREPARE
VOLMODIF	
SORTIDX	H_SORTIDX_LS
TAPEPREP	H_P RTP

All other Data Management Utilities execute the load module H_UTILITY.



5.6.5 SYSOUT Overflow

OU13 ron.ssn SYS.OUT OVERFLOW

This message indicates that the specified step of the job was unable to write to the SYS.OUT file as it was full and could not be extended (message OU25).

If outputs can be processed by the Output Writer this message is only a warning; the step resumes its execution, without any operator intervention, when some space becomes available.

Sometimes operator intervention is required:

- To find out the current contents of the SYS.OUT file by using the `DISPLAY_LOAD OUTPUT SPACE` command.
- To issue a `START_OUTPUT_WRITER` command on classes and devices for which outputs are waiting.
- To use the `RELEASE_OUTPUT` command with the `STRONG` parameter on outputs in the `READY` or `HOLD` state.
- To prevent other jobs from generating outputs by using the `HOLD_JOB` (HJ), `MODIFY_LOAD` (MDLD), `MODIFY_JOB` (MDJ), `CANCEL_JOB` (CJ) commands as appropriate.
- Under certain circumstances (e.g. printer failure) the Output Writer should be started on a tape file to free SYS.OUT space using the `START_OUTPUT_WRITER` (SOW) command. The tape may then be printed later using the GCL command `DPRINT`.

NOTE:

It is possible to prevent a step from using too much SYS.OUT space by use of the `LINES` parameter of the `STEP JCL` statement. Action should be taken to prevent SYS.OUT overflow from happening again; e.g. submit jobs in a different order, use private SYS.OUT files, correct a possible endless loop in a program.



5.6.6 Suspended Jobs

Executing jobs can be suspended using the HOLD_JOB (HJ) command:

```
HJ [ronlist [ENDSTEP]]
```

If the ENDSTEP parameter is used, the job is suspended at the end of the current step.

Suspending a job prevents any further execution of the job.

Suspending a job does not lower either the system or class load.

If the job is suspended at the beginning or in the middle of a step, before devices have been allocated, no device(s) are allocated.

Swappable memory is also freed when a job is suspended, but other memory such as I/O buffers is not.

When a job is suspended between steps, it frees all resources except "passed" files.

A suspended job can be released by the RELEASE_JOB (RJ) command. When a job is released it starts to execute immediately, in accordance with its Execution Priority.

Service jobs should not be suspended except under exceptional circumstances. JTRA cannot be suspended.

NOTES:

1. The TERMINATE_SYSTEM (TSYS) command is not effective until all suspended jobs are released.
2. Jobs can be suspended or released by the system (see Section 5.9 *Automatic Resource Management (ARM)*).



5.6.7 Before Journal

The Before Journal is a disk-based system file containing all information needed to ensure the integrity of the journalized files. This file is used to restore a user's file to its previous state (last checkpoint, last commitment unit, beginning of the step) after a step abort or a system crash. The use of the Before Journal is not normally visible to the operator except when the file becomes full and the application programmer has not previously given a list of media (same device class as the volume supporting SYS.JRNAL) on which to extend the file. In these circumstances, the following message is displayed:

```
JL02 ron MORE SPACE NEEDED FOR BEFORE JOURNAL. WHERE?
```

There are three possible answers:

volnm	the operator replies with the name of a disk with room for the file extension and the job continues. The extra area used is freed when the job or step terminates.
NO	this is the normal reply if no more space is available. The step is aborted.
CONTINUE	if the operator replies CONTINUE, the job continues, but it does so as if no checkpointing has been used. This reply should only be used with the user's express permission. CONTINUE is not a valid reply if the step concerned is under GAC (Generalized Access Control). In such a case, the reply CONTINUE is ignored and the message JL02 appears again; the only possible answers are volnm or NO.

If the extension is not possible on the specified disk, the following message is received:

```
JL03 ron EXTENSION ON volnm NOT POSSIBLE [RC = retcode]
```

and the JL02 message is repeated.

If an I/O error occurs on the Before Journal, thus making its further use impossible, the following message is sent to the step submitter (who may be the main operator):

```
JL05 ron BEFORE JOURNAL I/O ERROR. CONTINUE?
```



Possible replies are:

- | | |
|-----|---|
| NO | This is the normal response which aborts the step. This is the default, assumed when the step submitter is not connected. |
| YES | With this response, the job continues without the Before Journal. |

If no Before Journal exists, the following message is sent to the step submitter:

```
JL06 ron SYSTEM WITHOUT BEFORE JOURNAL. CONTINUE?
```

In this case, the step submitter must answer YES or NO:

- | | |
|-----|---|
| YES | Causes the step to continue without the Before Journal. |
| NO | Aborts the step. This is the default, assumed when the step submitter is not connected. |

TDS uses the Before Journal in a slightly different manner. The part of the Before Journal used by a TDS is an additional journal file dedicated to this TDS which resides on 1 to 4 disks. After each TDS generation, if the Before Journal location was not specified in the JCL, the following message is sent to the step submitter if he is connected, if not to the main operator:

```
JL07 ron SPACE NEEDED FOR BEFORE JOURNAL FOR TDS tdsnm WHERE?
```

The answers are:

- | | |
|---------|---|
| VOL-LST | This specifies a list of up to 4 disk names and an optional dvc parameter. The dvc parameter is used to specify the device class of the disks if it is different from the device class of the media used for the SYS.JRNAL Before Journal file. |
|---------|---|

```
VOL-LST = volnm1 [volnm2 [volnm3 [volnm4]]] [dvc]
```

- | | |
|----|---|
| NO | This is the normal reply if no more space is available. The TDS session is aborted. |
|----|---|

If insufficient space is available on one of the indicated disks of VOL-LST, the message:

```
JL08 ron SPACE ALLOCATION ON vol-lst NOT POSSIBLE  
[RC = retcode]
```

is received and the JL07 message is repeated.



When you change the media on a TDS warm restart, the following message is sent to the step submitter:

```
JL09 ron BEFORE JOURNAL FOR TDS tdsnm ON vol-1st?
```

The two possible answers are:

- | | |
|-----|---|
| YES | This is the normal reply ; the same volume is used as in the previous session of this TDS. It is also the default, assumed if the step submitter is not connected. |
| NO | To change the volume name for the Before Journal. Question JL07 is repeated to ask which volume name is to be used. |

5.6.8 Journalization Advanced Service (JAS) and the After Journal

The Journalization Advanced Service (JAS) and the After Journal are magnetic tape or disk files which are used for restoring a file's updated contents from a previous copy (save) after a failure. It is also used dynamically by a Transaction Driven System (TDS) to handle unsuccessful transactions after an abort or a system crash. The disks or tapes used for the journal are normally specified in the MAINTAIN_JAS utility which provides the facilities for their maintenance. The system Administrator can use MAINTAIN_JAS to extend the volumes used by the journal.

NOTE:

MAINTAIN_JAS replaces the JAGEN utility of GCOS 7-V5 and earlier releases.

The Journalization Advanced Service (JAS) protects the contents of a user file against loss by:

- Rolling back the incomplete file recovery units and rolling forward the file recovery units to be completed.
- Rolling forward destroyed user files using the ROLLFWD utility in conjunction with SAVE and RESTORE operations.

JAS protects user files by immediate update with the Before Journal, by deferred update and the After Journal. The After Journal should be used for protection against hardware or recovery incidents.



The use of the After Journal requests the mounting of volumes for ron X1, i.e. for the system. They must be satisfied prior to other mounting requests, since they may impact all jobs using the after journal. Each time a volume is used for the journal, a message is displayed to show which is the current and next volume allowing the operator to prepare or pre-mount the next volume to save time at the next switching:

```
JP12 jas name: AFTER JOURNAL IS ON MEDIUM cur_volnm_name
          CURRENT AFTER JOURNAL FILE IS efn
          NEXT MEDIUM WILL BE next_volnm_name
```

When the media list is exhausted, the following message is displayed:

```
JP13 jas name: AFTER JOURNAL IS ON MEDIUM cur_volnm_name
          CURRENT AFTER JOURNAL FILE IS efn
          WARNING: LAST MEDIUM.
```

It is advisable to run the MAINTAIN_JAS utility as soon as possible to enter a new volume list, or extend the existing volume list. See the *File Recovery Facilities User Guide*.

When the jas-name.JADIR directory is almost full, the message:

```
JP22 jas name:THE SYS.JADIR DIRECTORY FILE IS ALMOST FULL.
          ANY NEW JOURNALIZING ACTION WILL ABORT
          DELETE IT AND CREATE IT LARGER
```

is displayed. Every step using the After Journal is aborted.

NOTE:

When the After Journal is used, ensure that the system date is accurate because discrepancies between the system date and last After Journal use are checked.

Two situations may occur:

- Predating

The system date is earlier than the last After Journal date. In this case the following message is displayed:

```
JP31 jas name: WARNING: PREDATING:
          LAST RECORDED AFTER JOURNAL DATE: date or time
          FUTURE ROLLFORWARDS ARE NOT GUARANTEED,
          CONTINUE? (Y,N)
```

The consequence of a predating situation is that some files may not be restored to the correct state (this operation is called a rollforward).



- Postdating

The system date is later than the last After Journal use. The consequence of a postdating situation is that it may cause a predating at the next session when a correct date is given. This is a normal situation, but if one work day has passed since the last use of the After Journal, the system detects inconsistency in the date.

If the system date is considered to be incorrect, reply NO to this question. NO aborts the step which was trying to use the After Journal. Terminate the system and perform ISL again, to provide the correct system date, as soon as possible.

NOTES:

From the GCOS 7-V6 release:

1. The MAINTAIN_JAS utility replaces JAGEN and the message key JP replaces JA.
2. The operator can use the DISPLAY_JAS command to display all the steps connected to a JAS, meaning all steps that use files protected by a JAS.



5.7 Job Termination

Whenever a user job terminates, the following message is displayed:

```

_____
{ COMPLETED }
(JB02) hh.mm ron.ssn { ABORTED } jnm usernm jclass [SEV sevnb]
_____
{ KILLED }

```

COMPLETED	the job has been executed and terminated normally.
ABORTED	means that the job has been aborted for one of the following reasons: <ul style="list-style-type: none"> - A fatal error was found in the job description during translation. - An abnormal condition occurred during execution. - A CANCEL_REQUEST command was used to reply to a system request for a certain device or volume.
SEV sevnb	indicates the severity of the step termination.
KILLED	means that the job has been terminated by the CANCEL_JOB (CJ) command which is described in Section 7.

NOTE:

Service jobs do not normally display this message.

When a job is terminated it does not disappear from the system until all its outputs have been processed by the Output Writer.

When a job is terminated, GCOS 7 looks for another job to run, unless a TERMINATE_SYSTEM command has been issued.

If no jobs can start (i.e. there are no jobs in the IN SCHEDULING state) and there are no more jobs executing, the following message is displayed:

```
(SH06) hh.mm GCOS: NO MORE JOBS RUNNING
```

More jobs should be introduced or made available for scheduling, unless the operator wants an empty machine for a specific reason.



5.8 Output Writer

During a job run various outputs are prepared for printing. Instead of printing these outputs directly the standard method is to store them on an intermediate file. This allows more jobs to be run simultaneously than there are printers on the site. It leads to a more efficient use of CPU time and of the printers.

The Output Writer is part of GCOS 7 and is in control of selecting outputs for printing, and printing them via the WRITER service job.

When all of a job's outputs are printed the job is no longer known to GCOS 7, and the Output Writer displays the following message:

```
(OU14) hh.mm rrr OUTPUT COMPLETED jnm usernm
```

The Output Writer may also be started on a sequential tape or disk file instead of a printer. Each output is considered as being printed as soon as it has been transferred to the file. A file created in this manner can be printed later by a job using the WRITER JCL statement or the GCL command DPRINT.

The maximum number of outputs for a job that can be known to the Output Writer is 254. A job can produce more than 254 outputs while it is running, since outputs that have been printed are deleted.

Certain outputs created by jobs introduced via the RBF (Remote Batch Facility) are sent back to the issuing station and not printed on the main site. These "special" outputs are handled by remote writers and sent to the remote devices or files. The maximum number of started writer jobs is 57.

Their number may be limited by the multiprogramming level of jclass W.

With the DJP facility (Distributed Job Processing), output may be directed or spooled towards other DPS 7000 or DPS 6000 sites, using a WRITER started on a remote file or a GTP service job. They are managed and selected in the same way as other writers.



5.8.1 Outputs

5.8.1.1 Standard and Permanent Outputs

An output which is waiting to be printed can be stored in either the system file SYS.OUT or in a permanent file - temporary user files are not permitted. In the first case the output is said to be standard and in the second to be permanent.

Once a standard output is printed, the area it used in SYS.OUT is freed. Since all jobs producing standard outputs share the SYS.OUT file an overflow condition may arise (see "*Sysout Overflow*", 5.6.5 above). Permanent outputs may be on non-resident volumes, tape or disk, and a mounting request is issued if necessary. Permanent outputs are not erased when they are printed and can be printed again by use of the WRITER JCL statement or the GCL command DPRINT, unless the DELETE option is used.

5.8.1.2 Output Names and Sequence Numbers

Outputs created by a job may have a specific name given by the user and they are also given a sequence number (within the ron) by GCOS 7. Therefore the naming of an output can take one or both of the following forms:

X17:TRUC (ron:ounm)	An output name (ounm) has been given by the user. NOTE: avoid giving the same name to several outputs of the same job, as commands such as CO X17:TRUC only affect the first output encountered by the system
X17:1 (ron:ouseqnb)	This output is the JOR or JOB_REP (Job Output Report) in which all the information concerning the execution of the job is given. This report is a virtual memory file printed by the Output Writer. Even though it is one of the outputs, it is not a member of the SYS.OUT file and it is not in SYS.OUT format. <u>Do not confuse X17:1 and X17.1!</u> (X17.1 is the execution of step 1 of job X17. Refer to "Job execution" Section in this section for further information).
X17:2 (ron:ouseqnb)	This output is the JOB_OUT, and contains all the output of job using the same output characteristics as JOB_REP (media, class and priority). X17:2 may contain outputs from several different steps of X17.



X17:3 This is an output with different output characteristics from the JOB_OUT.

X17:DUMP_3 Some cases (especially for service jobs dumps) dumps are identified in this way.

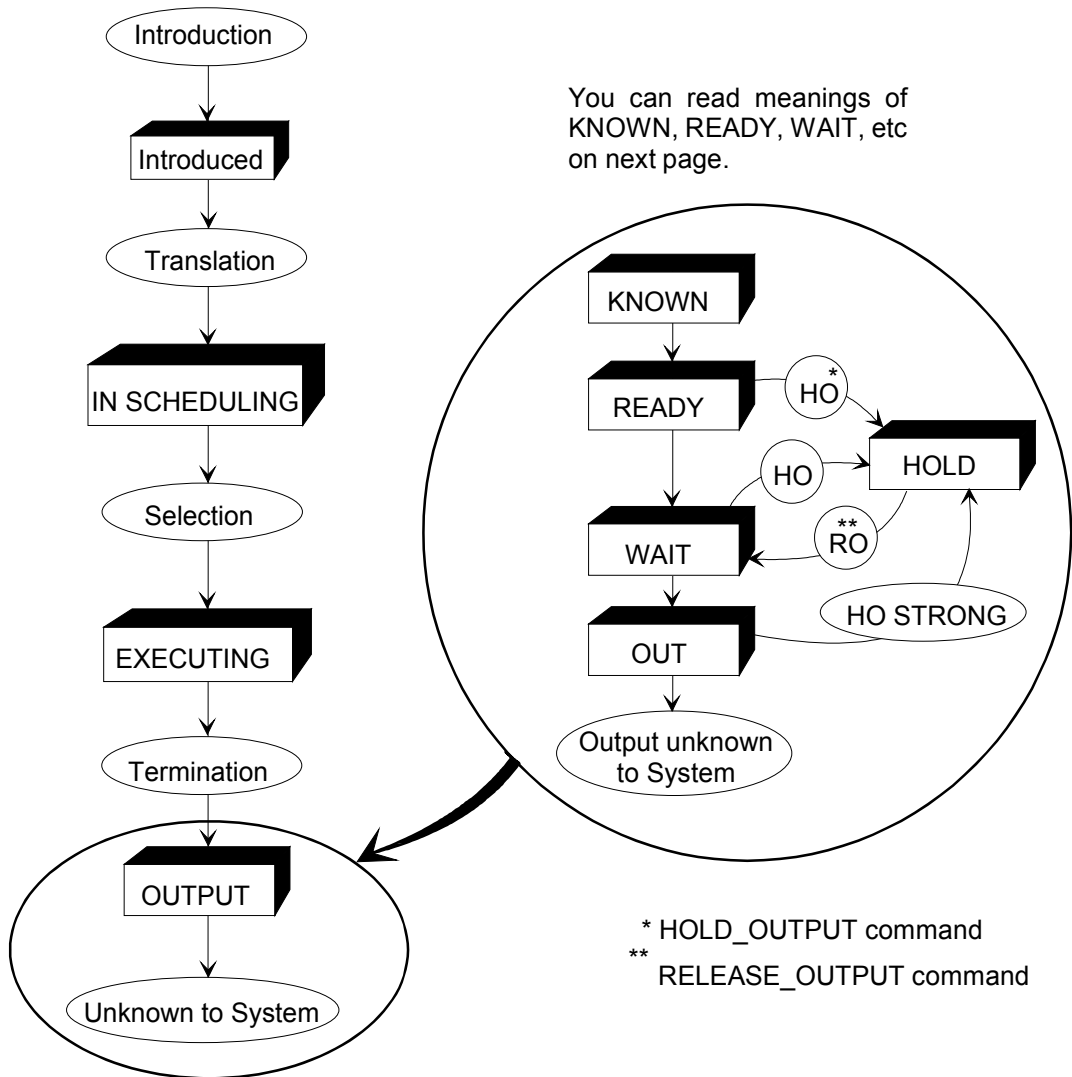


Figure 5-4. Output States



5.8.1.3 Output States

KNOWN	the life of an output begins when first created by a job. From this time until it is printed by the Output Writer, the output is known by the system.
READY	when the job closes the output file it is then said to be READY. However, it is not yet eligible for selection by the Output Writer.
WAIT	as soon as the "when condition" indicated in the JCL statements is met the output enters the WAIT state; it is then eligible for selection to be processed by the Output Writer. The "when condition" can be the end of job (which is the default), end of step, closing of the file. If the <code>HOLDOUT</code> parameter is used with the <code>EJR</code> command or on the <code>\$JOB</code> command line, all the outputs for that job remain in the HOLD state when the "when condition" is met. These outputs remain in the HOLD state until released by the <code>RELEASE_OUTPUT (RO)</code> command.
OUT	while the output is actually being processed by the WRITER service job (i.e. printing) it is in the OUT state. At the end of this operation the output is no longer known to the system.
HOLD	an output can be put into the HOLD state by the <code>HOLD_OUTPUT (HO)</code> command, or by using the <code>HOLD</code> parameter in the <code>EJR</code> command. An output in the HOLD state is not selected until a <code>RELEASE_OUTPUT (RO)</code> command is issued. An output that is being processed can be put into the HOLD state by use of the <code>HO STRONG</code> command, and question <code>OU02</code> is then asked (see <code>HOLD_OUTPUT (HO)</code> command).
WTFL	this is the "Wait for File" status. It is indicated after a <code>DISPLAY_OUTPUT QUEUE</code> command when an output has been selected by the OUTPUT WRITER but the input file for it cannot be accessed. When the input file is freed, the status of this output is automatically changed to WAIT and it is then printed.

Note that a `DISPLAY_OUTPUT xron` command does not return WTFL, but WAIT in the same situation.



SCAND

means that an output is in a scanned state, that is to say the result of scanning by the SCANNER processor. It is the response to a DISPLAY_OUTPUT OUTPUT=output | ron command. After scanning, the state is that requested by the user.

While an output is in the SCAND state, a CANCEL_OUTPUT, RELEASE_OUTPUT, or FORCE_OUTPUT command will be rejected.

When an output is processed by the SCANNER processor, it is automatically held at the start of scanning. It is therefore in the HOLD state as far as OUTPUT WRITER is concerned, so a DISPLAY_OUTPUT STATE=QUEUE | HOLD command shows this output in the HOLD state if it is enqueued.

5.8.1.4 Output Commands

Table 5-11. Output Commands

OUTPUT COMMANDS	MEANING
CANCEL_OUTPUT (CO)	Cancels the notification to the Output Writer to process one or several outputs.
DISPLAY_CONFIGURATION [OUTDV] [OUTCLASS]	Displays software information on the status of selected output devices, or output classes.
DISPLAY_LOAD [OUTPUT] [ALLOUT]	Displays selected information on the current output load.
DISPLAY_OUTPUT (DO)	Displays the state of outputs, output queues, and output devices.
FORCE_OUTPUT (FO)	Positions a specified output as the first to be treated.
HOLD_OUTPUT (HO)	Puts a specified output or outputs into HOLD.
MODIFY_OUTPUT (MDO, MO)	Modifies one or several characteristics, such as output classes, priorities, destinations, and number of copies.
RELEASE_OUTPUT (RO)	Makes one or several outputs available for processing.
START_OUTPUT_WRITER (SOW)	Starts the Output Writer on devices or files.
TERMINATE_OUTPUT_WRITER (TOW)	Terminates the Output Writer on devices or files.
For full information on the relevant commands, see Section 7 <i>Description of the Operator Commands</i> .	



5.8.1.5 Output Priorities (OPR)

Outputs are assigned priorities in the same way that jobs are assigned Scheduling Priorities. The OPR is used to queue outputs waiting to be processed by the WRITER service job.

Output Priorities are in the range of 0 - 7, with 0 being the highest. The outputs are queued in priority order and on a FIFO basis for the same priority. Outputs waiting to be processed are said to be in the WAIT state.

Output Priorities are either given explicitly in the job description or implicitly by default in relation to the output class. Operator commands affecting output priorities are as follows.

Table 5-12. Output Priority Commands

Form	Command	Function
DO	DISPLAY_OUTPUT	displays output priorities.
MDO (MO)	MODIFY_OUTPUT	modifies output priorities.

5.8.1.6 Output Class (OC)

There are 26 Output Classes in the range of A - Z. These are used to control outputs in the same way that Job classes are used to control jobs.

Each Output Class has a default output priority and a default output media at system configuration time. Operator commands affecting output class priorities are as follows:

Table 5-13. Output Class Priority Commands

Form	Command	Function
DC OC	DISPLAY_CONFIGURATION	displays output class priority and the started writers in the specified or submitter station.
MDC OC	MODIFY_CONFIGURATION	modifies default priority of an output class.



The class of an output can be given explicitly in the job description. If not, then a default class is assumed. The class to be used as default is specified by the CONFIG job. Operator commands affecting output classes are as follows:

Table 5-14. Output Class Commands

Form	Command	Function
DO	DISPLAY_OUTPUT	to see the details of an output.
MDO (MO)	MODIFY_OUTPUT	to change the class of output.

Each Output Class can be started or terminated on each output device. It is also possible to start more than one class on a device. This is the usual case (see START_OUTPUT_WRITER and TERMINATE_OUTPUT_WRITER commands in Section 8 *Description of the Operator Commands*).

If at system configuration no values are specified, the following default values are assumed by GCOS 7:

```

-----
OUTPUT CLASS      - A B C D E F - Z
                   |_____|
                   |----> CLASS "C" IS THE DEFAULT CLASS

OUTPUT PRIORITY  - 1 2 3 4 5 6

OUTPUT MEDIA     - I10000
-----

```

NOTE:

Double-letter class (AA...ZZ) do not apply to outputs, but only to job classes.



5.8.1.7 Output Selection

When an output enters the WAIT state it is put last in the output queue of its priority. This is also true for an output in the HOLD state.

Whenever an output device is free and its associated WRITER job is IDLE, the Output Writer determines which output, if any, will be processed. The output selected is the first (highest priority) to satisfy the following conditions:

- Not be in the HOLD state.
- Belong to a class started on the device.
- Be directed to a device class that correlates the desired output format with a compatible device.

It is possible to force an output with the FORCE_OUTPUT (FO) command; this output is selected prior to any other, although it must still satisfy the second and third conditions shown above.

Therefore, the order in which outputs are processed can be controlled by use of:

- Output Priorities.
- Output Classes (started or not on each device).
- HOLD_OUTPUT Command.
- FORCE_OUTPUT Command.

5.8.1.8 Question OU02

If the printing of an output is interrupted by:

- TERMINATE_OUTPUT_WRITER STRONG,
- or a HOLD_OUTPUT STRONG,
- or a CANCEL_REQUEST command was used in reply to a device incident, etc., then question OU02 is asked.

The question OU02 is sent to the operator who submitted the SOW command. Therefore, if the command is submitted in the startup system or if the operator is no longer connected, the question cannot be asked and the reply "CUR" is assumed.

In the reply the operator can specify where the processing of the output should resume from, if at all:

```
nn/OU02 dvmn OUTPUT ron :ouseqnb RESTART FROM?  
(CURRENT PAGE NUMBER is current_page)
```



current_page is the page number at which the incident occurred for OUTPUT-WRITER. This is the page number used to compute the restart page in the answers FOR, BACK, CUR below. The page number is not necessarily the last printed page of printer buffering.

Possible Answers:

- BEG [HOLD] [TERM] resume from the beginning of the output.
- BACK nn [HOLD] [TERM]
resume nn pages before the current page.
- CUR [HOLD] [TERM] from the beginning of the current page (do not use this option after an I/O incident on a PRU 412x printer - see Section 6 *Peripheral Operations* for further information).
- CONT from the next line. This is used only to bypass the current line; i.e. if that line caused the failure.
- FOR nn [HOLD] [TERM] nn pages after the current page. Jumps over nn - 1 pages without printing them.
- CAN [TERM] the output is cancelled; it will not be resumed at all.
- FROM nn[HOLD][TERM]
start output from page number nn.

NOTE:

BACK nn can be abbreviated to -nn and FOR nn to +nn.

If the length of the answer is zero, then CUR is taken as default.

HOLD indicates that the output is to be put into the HOLD state; in this case a RELEASE_OUTPUT is necessary before the output can be resumed. If the question was asked because of a HOLD_OUTPUT (HO) command, HOLD is implicit.

TERM indicates that all output classes are to be terminated on this device. If the question was asked because of a TERMINATE_OUTPUT_WRITER (TOW) command, then TERM is implicit.

If the current output is held after question OU02 is replied to and TERM was not used, then the Output Writer tries to start a new output; if no output is suitable the WRITER service job goes into the IDLE state.



5.8.1.9 Output Writer Incidents

During its activity the Output Writer may encounter certain incidents which require operator intervention.

1. Incident on the Output Device

The operator is informed of the incident by message DV12, a recovery request. The operator recovers the device (as described in Section 6, Peripheral Operations) and the Output Writer resumes normal activity without further operator intervention. See note below.

If the incident happened on a printer and a number of pages need reprinting the operator issues a CANCEL_REQUEST command in response to message DV12, and question OU02 is asked. After switching the printer to READY the operator replies to OU02 indicating the number of pages to be reprinted and the output processing is resumed at the specified point.

If for some reason output processing should not be immediately resumed after an incident, the procedure in the above paragraph should be carried out. The only difference is that the operator uses the HOLD option in the reply to question OU02.

If the incident was a power-off, GCOS 7 reacts as if a CANCEL_REQUEST (CR) command had been issued in response to the DV12 message, and question OU02 is asked.

The incident may be more serious and message OU27 is displayed and followed by question OU02.

The general format of OU27 is:

```

-----
OU27 dvmn ron :ouseqnb { HELD          } [WRITER TERMINATED]
                       {                }
                       { CANCELLED     }

      RC = cccccc --> siu,return code [ON SYS.OUT FILE]
-----

```

After the operator has recovered the device output processing resumes at the point specified in the reply to question OU02. Normally message OU27 ends after HELD or CANCELLED. However, if the output has been HELD, it must be released before processing is resumed (In fact, the Output Writer may begin processing another output as soon as question OU02 is answered).

If CANCELLED is shown, the output has been cancelled and in the case of a standard output it will have been erased.



If WRITER TERMINATED is shown, the Output Writer must be restarted on that device.

NOTE:

If question OU02 was not asked, the Output Writer starts printing after recovery from the current line. This may result in that line being duplicated. In this case mark the line which was printed twice and inform the user.

2. Incidents on the Volume Containing the Output File

These incidents are treated like any other tape or disk incident (see Chapter 6, *Peripheral Operations*) and if recovery is successful the output continues without any further operator action. Tapes are automatically repositioned if the recovery was successful.

If the incident could not be recovered, the operator issues a CANCEL_REQUEST (CR) command. Message OU27 is displayed indicating that the output has been HELD, question OU02 is then asked. After the operator has replied to OU02, the Output Writer selects the next available output. The output must be released before it can be reprocessed by the Output Writer.

3. Paper Form or Print Belt Cannot Be Used.

If the output volume cannot be used, this may be due to an error in the WRITER or OUTPUT JCL parameters. In this case message OU27 is displayed indicating that the output is HELD. The return code displayed contains information concerning the erroneous parameter and therefore determines the operator response. Refer to the *JCL Reference Manual*.

If the output volume cannot be used for any other reason, in particular after the issuing of a CANCEL_REQUEST command in reply to a mount volume request, then message OU28 is displayed:

```
-----  
OU28 dvm [WRITER TERMINATED]  
          [RC = ccccc -> siu, return code]  
-----
```

The Output Writer is terminated on the specified device and the output being processed is left in the WAIT state. The Output Writer can be started again on that device but only for outputs that do not need the unavailable volume. It is thus advisable to hold these other outputs before starting the Output Writer.



4. The SYS.OUT File Cannot be Used

If a Permanent SYS.OUT file cannot be found, i.e. the wrong file was specified or the file has been deleted, then message OU26 is displayed, indicating the output sequence number of the incorrect file.

```
OU26 dvmn ron :ouseqnb CANCELLED
```

```
RC = cccccc -> siu, return code  
EFN = [SUBFILE = sfn]
```

The output is cancelled and Output Writer selects the next output to be processed.

In any other case where the SYS.OUT file cannot be used, i.e. the operator refused to mount the volume containing the file, then message OU27 is displayed showing that the output has been HELD and that the Output Writer has been terminated on the indicated device.

5. The SYS.OUT File is Incorrect

Sometimes unedited Permanent SYS.OUT files have errors that prevent the Output Writer from printing a line. In these circumstances, message OU27 is displayed and then question OU02 is asked. The operator should reply CONT to OU02 and print resume on the next line, bypassing the line at fault. For each line that is at fault the above messages are displayed. If there are a large number of these errors, use the HOLD option in reply to OU02 and consult the user.



5.9 Automatic Resource Management (ARM)

This manual only gives summary information about ARM. For more details, refer to the *ARM User's Guide*, or to the *System Administrator's Manual*.

5.9.1 What is ARM?

The Automatic Resource Manager (ARM) is a GCOS 7 system administration and regulation tool.

ARM as "administrator" is a set of main operator commands for communicating operational and optimization objectives to the system.

ARM as "regulator" dynamically measures, reports and controls the system's resource consumption.

ARM dynamically controls the distribution and usage of the three main system resources:

- memory,
- CPU,
- multiprogramming levels.

At regular time intervals, ARM measures the current workload on each of these resources and then, on a feedback basis, adjusts the control rules to obtain the best results possible within the specifications given. The general mechanism is summarized in Figure 5-5 *ARM Mechanism*.

From ARM's point of view, the system is made up of a number of dimensions each competing in some degree or another for a share of the system's resources. ARM's overall objectives are to:

- distribute the resources between the different dimensions according to the level of importance of each dimension,
- distribute resources within a dimension according to the resources allocated to it and the rights and needs of the jobs running under it,
- balance the overall distribution of resources according to the global system load.

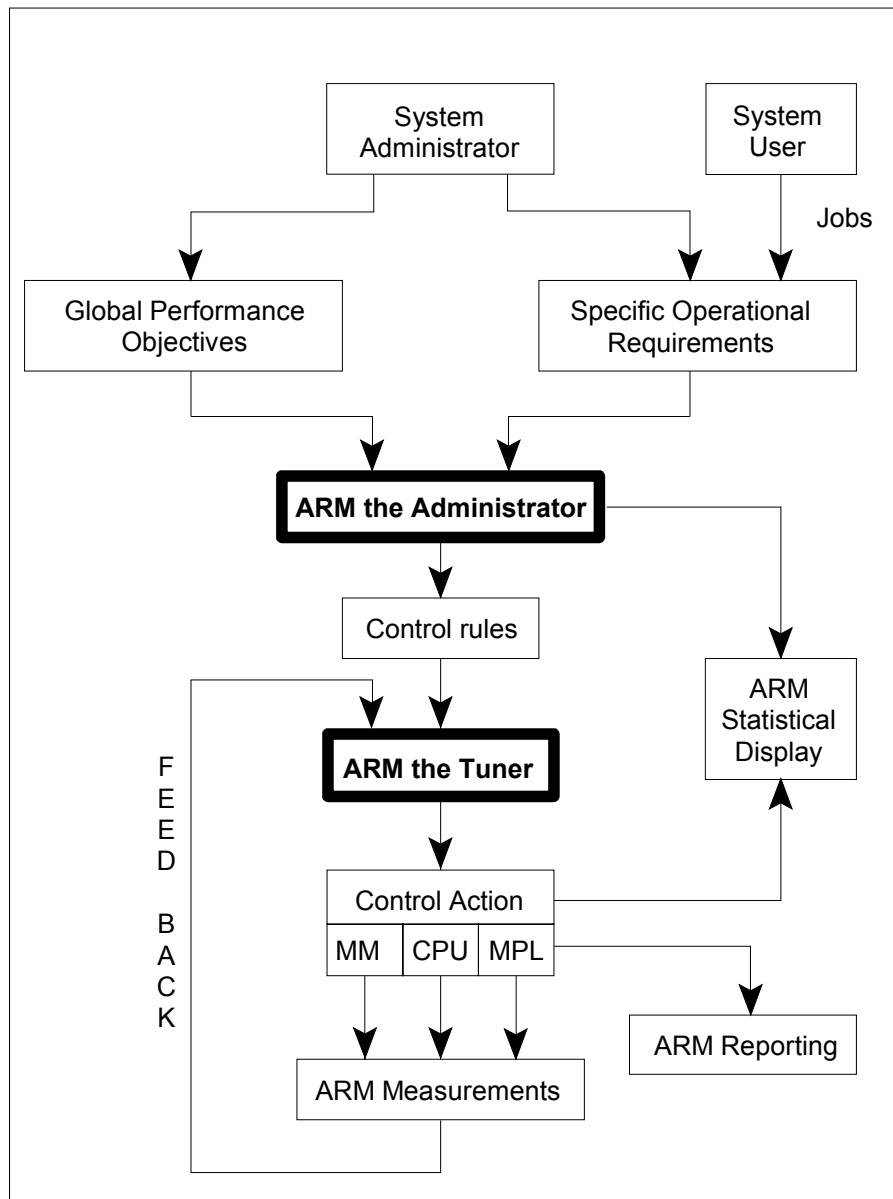


Figure 5-5. ARM Mechanism



5.9.2 Basic ARM and Full ARM

ARM has two levels of operation, called Basic ARM and Full ARM. Basic ARM is available with all DPS 7000 configurations. Full ARM is an option for the larger HPS systems.

All features, commands and procedures described in this document apply to systems equipped with Full ARM.

To find out whether your system is operating with Basic ARM or Full ARM, type in the DISPLAY_REGULATION_PARAMETER (DRP) as follows:

```
S: DRP;
```

This gives the following display:

```
MEM = xxxMB
```

```
LEVEL = (BASIC or FULL) STATE = (NORMAL or DEGRADED)
```

Memory size is given in Megabytes, the level of ARM that has been installed on the system is given and the present operational state of the ARM. FULL ARM in degraded state means that the BASIC ARM features are operational. BASIC ARM cannot be in DEGRADED state.

Basic ARM regulates the rules concerning memory management and working set protection. This leads to acceptable results where there is a moderate workload, behaving in a moderate manner.

Full ARM regulates not only rules concerning main memory, but also those concerning CPU time, and multiprogramming levels. This resolves many traditional performance problems which, in the past, could be corrected only through operator surveillance and intervention. Brief examples of the kinds of regulation performed by ARM are given below.



5.9.3 Concepts

5.9.3.1 Execution Level (XL)

The Execution Level (XL) is the level of importance of a given process, expressed as a number from 0 to 100. It is computed dynamically by ARM according to:

- the amount of resources already consumed by the step,
- the RELWEIGHT, XL range, and PA attributes of the dimension concerned.

The XL computed by ARM determines directly VMM's working set decisions (memory allocation) and CDAC's dispatching policies (CPU allocation) for the process concerned.

The XL of a dimension is the mean of the XLs of its steps. The XL of a step is the mean of the computed XLs of its processes. To display the current XL of a dimension, use the DDIM command with PERF option.

You cannot modify directly the XL of a process but you can modify the XL range of its dimension, using the MDDIM command.

5.9.3.2 Multiprogramming Limit (MPL)

This refers to the maximum number of steps which can be in execution at the same time in a given dimension or a given Job Class Group (see below).

With Basic ARM, the MPL is static and can be modified only by the operator commands MDDIM and MDLD.

With Full ARM, the MPL is dynamically modified according to the current system load. The MPL cannot, however, go beyond the bounds set by the MPL range attribute for the dimension. You can modify the MPL range of a dimension using the operator command MDDIM.



5.9.3.3 Execution Level Class (XLC)

Certain predefined XL ranges correspond to an Execution Level Class (XLC). Associated with each XLC is a CPU dispatching policy, a minimum CPU consumption rate and a hardware priority range.

Each XLC corresponds to:

- a hardware priority range,
- a fixed XL range,
- a CPU dispatching policy,
- a maximum CPU consumption rate.

The following table gives the default settings of XLC characteristics.

XL Range	XLC	Policy	Max.Rate	HPR
100	0	F	100%	0-F
90-99	1	F	100%	8
80-89	2	FS	100%	9
70-79	3	FS	100%	10
60-69	4	FS	100%	11
50-59	5	O	30%	11-12
40-49	6	FS	40%	12
30-39	7	O	50%	12-13
20-29	8	O	60%	13-14
10-19	9	O	80%	14-15
0-9	10	FS	100%	15

F=Fixed non Sliced, FS=Fixed Sliced, O=Optimized.

Note that XLC=0 is reserved for system applications and its characteristics cannot be modified.



5.9.4 Predefined Attributes

The default attributes for the four standard dimensions are given in the following table.

Table 5-15. Standard Dimension Default Attributes

Dim Name	Assignment Criterion	RELWGHT	MPL Range	XL Range	BWGHT	PA	ICA	HO
SYS	JC = R-ZZ	100	50-50	0-100	0			
TDS	Unassigned TDS steps	100	50-50	60-99	0		1	
IOF	JC = Q-QZ	70	5-200	30-69	20			
BATCH	Default	50	2-200	10-49	30	1		1



5.9.5 ARM Commands

5.9.5.1 Table of FULL ARM commands

Full ARM commands are, in alphabetic order, the following:

Table 5-16. FULL ARM Commands

Abbreviation	Command Name	Command Action
CNDIM	CONNECT_DIMENSION	Adds new assignment criteria to a dimension.
CNFUNC	CONNECT_FUNCTION	Connects a privileged function to a given resource domain.
CRDIM	CREATE_DIMENSION	Creates a user-defined dimension, modelled on an existing dimension.
DLDIM	DELETE_DIMENSION	Deletes a user-defined dimension. No applications can be assigned to it.
DISFUNC	DISCONNECT_FUNCTION	Disconnects a privileged function from a given resource domain.
DFUNC	DISPLAY_FUNCTION	Displays information about functions and domains.
DISDIM	DISCONNECT_DIMENSION	Removes specified assignment criteria from a dimension.
DDIM	DISPLAY_DIMENSION	Displays information and statistics relating to one or all dimensions.
DXLC	DISPLAY_XL_CLASS	Displays the current values associated with an XL class.
MDDIM	MODIFY_DIMENSION	Modifies the attributes of a dimension.
MDXLC	MODIFY_XL_CLASS	Modifies one or more characteristics of an Execution Level Class.



5.9.5.2 Availability of ARM commands

NOTE:

ARM commands are available only to the MAIN operator. The table below gives the availability of command in full and basic levels.

The access rights of the different ARM commands are summarized in this table:

Table 5-17. Availability of ARM Commands

COMMAND	'FULL' LEVEL	'BASIC' LEVEL
MDXLC	F	F
DXLC	F	F
CRDIM	F	U
DLDIM	F	U
MDDIM	F	P
CNDIM	F	U
DISDIM	F	U
DDIM	F	U
CNFUNC	F	F
DISFUNC	F	F
DFUNC	F	P

NOTES:

- F= totally available
 U= unknown command.
 P= partially available in BASIC Level. The commands and keywords in BASIC are:
 MDDIM dimension MPL=dec3
 DDIM dimension OPTION=MPL.
- The use of the CRDIM, DLDIM, CNDIM and DISDIM commands on a system at BASIC Level results in the following message being displayed:
 AR45 UNKNOWN COMMAND cmdnm
 where cmdnm is the name of the command used.



5.9.6 ARM Operator Messages

We give here a list of all the ARM generated operator messages. Each is followed by a brief explanation of its significance.

JOB SUSPENDED BY SYSTEM

A job running in an overloaded dimension, whose HO attribute is on, has been suspended by ARM.

JOB RELEASED BY SYSTEM

A previously suspended job has been released for execution.

WAIT FOR RESOURCES IN DIMENSION xxxxx

A step has been queued in the BATCH dimension"

JOB REACTIVATED

A step has left the BATCH dimension queue.

RESOURCES NOT AVAILABLE NOW IN DIMENSION xxxxx

New IOF steps are rejected, and only trivial requests (at S: level) are allowed.

WARNING: SYSTEM ENTERS THRASHING

The system has been diagnosed as overruled.

SYSTEM LEAVES THRASHING

ARM has dealt with the cause of thrashing.

ARM ENTERS DEGRADED INTO BASIC RC=cccccccc -> SIU, FC

ARM has made an error and cannot recover from it. Only Basic ARM facilities are currently available.

Please contact your support if this happens.

GCOS 7 V9

WARNING: SYSTEM ENTERS NORMAL USE

This message is followed by the reason of the change of mode.

SYSTEM BECOMES UNDERUSED

The CPU consumption is low.



5.10 Domain and CDP (Customer Dedicated Processor) Overview

These functions are available only on the DPS 7000/MTxx models.

5.10.1 CDP (Customer Dedicated Processor) Overview

The physical CPU resources that can be installed at the client's site can be any of the following processors and co-processors:

IPU: Instruction Processor Unit	Standard processor
EPU: Extended Processor Unit (type 1)	CDP1
FPU: Extended Processor Unit (type 2)	CDP2
GPU: Extended Processor Unit (type 3)	CDP3

Previous names for co-processors were:

HRP	High Relational Processor (first co-processor for Oracle), introduced on DPS 7000/4x0/5xx/7xx series.
X-HRP	High Relational Processor (co-processor for Oracle and OPEN 7), introduced on DPS 7000/4x5/8xx series.
BES	Back End Server (board containing 4 EPUs in an DPS 7000/8xx/Mxx/Dxx)
PSP	Parallel Specialized Processor, introduced on DPS 7000/Cxx/Dxx/Mxx series.



5.10.2 Domain Overview

A domain represents a kind of CPU resource. There are four domains:

- standard
- CDP1
- CDP2
- CDP3

A function is a set of applications that can be privileged, or not. A privileged function can be assigned to a domain. The following functions are privileged:

- SYSTEM
- ORACLE
- OPEN7
- DW7
- IQS
- OPEN7_F1
- OPEN7_F2

A software MI must be bought to allow a privileged function to be assigned to a domain.

There are four types of hardware resources and domains: see the Table below.

Hardware Resource	Domain
IPU: standard processor for non-privileged functions	STANDARD domain
EPU: processor for all privileged functions not assigned to another domain	CDP1
FPU: processor for a privileged function specified by the administrator	CDP2
GPU: processor for another privileged function specified by the administrator	CDP3

A domain exists if the corresponding hardware configuration exists. For example:

- if your DPS 7000/MTxx has an IPU and an EPU, STANDARD and CDP1 domains exist.
- if your DPS 7000/MTxx has an IPU, an EPU and a GPU, STANDARD, CDP1 and CDP3 domains exist.



Assignment Rules

1. All functions can be executed using the STANDARD domain resources. This domain cannot be assigned to a function.
2. The CDP2 or CDP3 domain resources can be dedicated to one privileged function if the administrator has connected the domain and the function. The connection is established using the `CONNECT_DOMAIN` command.
3. The CDP1 domain resources are available to all privileged functions not connected to the CDP2 and CDP3 domains.
4. A function assigned to CDP1 (using CDP1 resources) can be disconnected. In this case it can use only the resources of the STANDARD domain.



6. Peripheral Operations

6.1 Introduction

This section describes the functioning of the system peripheral devices under GCOS 7 V7, V9 and V10.

The following topics are discussed:

- **Device Handling Overview:**
introduces the terminology, characteristics, commands and messages common to all devices.
- **Disk, Magnetic Tape and Tape Cartridge Operations:**
describes the operational characteristics of each device class.
- **Printer Operations**
- **Hardware Incidents:**
describes incidents, recovery, and error logging.
- **Monitoring Device Incidents:**
describes use of Device Trace commands to monitor device incidents.



6.2 Device Handling Overview

Operators must understand the following terminology:

DEVICE TYPE (dvtyp)	Each device belongs to a device type identified by two letters.
Examples:	MS (Mass Storage disk) CT (magnetic tape CarTridge) MT (Magnetic Tape reel) PR (PRinter)
	See Appendix F <i>Device Types and Classes</i> for more information.
TAPE	Magnetic tape cartridge (cartridge tape) or magnetic tape reel (reel tape). Cartridge tapes can be in a Cartridge Tape Library.
DEVICE NUMBER	Each device of a device type is identified by a device number (two digits or letters).
DEVICE IDENTIFIER (dvnm)	The device identifier is made up of 2 letters for the device type and of 2 characters for the device number (for instance PR01, MSA3).
{ VOLUME { { FILE	Disks and tapes are referred to as volumes. A volume contains one or more files. Each file is, for the user, a set of information.
REMOVABLE VOLUME	Tapes.
NON-REMOVABLE VOLUME	Non-removable disks. All disks supported by GCOS 7 are non-removable.
VOLUME DESCRIPTION	Volumes are identified by their volume name and their device class.
VOLUME NAME (volnm)	6 characters which can be alphabetic, alphanumeric or numeric.
DEVICE CLASS (dvc)	Symbolic name used to describe the characteristics of a device. Its first two letters are the device type.



Examples:	MS/FSA (FSA disk) MS/B10 (B10 disk) ✓ (not supported on DPS 7000/XTA) CT/M6 MT/D6250 (6250 BPI density reel tape) ✓ Seen as CT/M6 on DPS 7000/XTA See Appendix F <i>Device Types and Classes</i> for more information.
PREMOUNTING	For a removable volume: the action of loading a volume on a device before it is actually needed or requested. For a non-removable volume: the action of switching the device to the READY state before it is actually needed or requested.
MOUNTING	For a removable volume: the action of loading a volume on a device when it is requested by a job. For a non-removable volume: the action of switching the device to the READY state when it is requested by a job. If a mount request from a job cannot be satisfied, the operator indicates this to the system by using the CANCEL_REQUEST (CR) command.
AVR	Whenever a device is put into the READY state, the system tries to identify the volume. This operation is called Automatic Volume Recognition. AVR identifies native volumes (volumes labelled according to the GCOS 7 standard). The facility is also available for print belts which are identified by a hardware "label" on each belt. For these volumes, no other operator action is required. (Non-standard volumes have to be presented to the system by an operator using the NAME_VOLUME command).



MDHW command
DEVICE INCIDENT

All incidents are recorded in the SYS.ERLOG system file. The system provides functions for handling this file. When necessary, the MODIFY_HARDWARE_CONFIGURATION (MDHW) command with the (OUT) parameter can be used to remove a faulty device from the configuration. The MDHW command with the (IN) parameter enables the operator to reinsert the device into the configuration.

6.2.1 Device States

A device (known within the system) is always in one of the following device states. To find the state of a device, use the DISPLAY_HARDWARE_CONFIGURATION or DISPLAY_CONFIGURATION command. States which are displayed by DHW or DC are denoted below in upper case.

- **OUT (or HELD)**
The device has been removed from the system configuration by use of the MDHW (OUT) command. This can be performed when the device is free. It can be put back into the configuration by use of the MDHW (IN) command. (These two commands are described in Chapter 8 *Description of the Operator Commands*).
- **free**
The device is not being used. It may be in the hardware state "off-line" (seen as STANDBY), STANDBY or READY. If it is READY, a volume may be premounted (DHW or DC then displays PREMOUNTED). When a native volume is premounted the device may become allocated without any operator action.
- **ALLOCATED**
The device is being used by one or several users. Its normal hardware state is READY. If an incident has occurred, a RECOVER request may be pending (DHW or DC then displays RECOVER). A MOUNT request may also be pending for this device (DHW or DC then displays MOUNT). The CANCEL_REQUEST command can be used to cancel this request, but in this case, if the device is allocated to several jobs, they are aborted.



On a system configured for Mirror Disks usage, the MIRROR attribute and specific MIRROR states are displayed by DHW or DC for mirrored disks. Refer to the *Mirror Disks User's Guide*, this option is no more available on DPS 7000/XTA platforms.

NOTE:

RESIDENT disks are invariably allocated to X1 and to X1 only. It is therefore not possible to see who are the users of files on such disks.

6.2.2 Device Allocation in Device Mode

When a job needs the use of a device out of usual GCOS 7 rules, it can request (in its JCL) that the device be allocated in device mode.

From that instant the handling of the device is entirely under the responsibility of the user (which can request mounting or dismounting of volumes through specific user messages).

Incident handling and device availability are still the system responsibility.

This allocation mode is, typically, used by the Test and Diagnostic modules.

6.2.3 Unknown Device

When a device unknown to the system (due to a lack of conformity between the System Resource and Status Table (SRST) and the GCOS 7 Software Device Tables or to a hardware incident) is made READY or set to STANDBY, the following message is displayed:

```
AV54 UNAVAILABLE PATH (PC = pcnb, LC = lcnb)
```

Action to be taken depends on the conditions which caused the message to be displayed:

- re-initialize with the correct SRST;
- or run the CONFIG utility to have the actual hardware configuration taken into account at the next ISL;
- or try to insert the device into the GCOS 7 configuration by the MDHW IN command. See the *Console Messages Directory* for more details.



6.3 Disk and Tape Operations

This subsection is divided into 3 parts:

- **Volume Characteristics:**
describes the logical characteristics of the various types of volumes.
- **Premounting, mounting and dismounting:**
describes the system responses to physical operations.
- **Incidents:**
discusses various non-fatal incidents which can occur when the system is reading or writing to disk or tape.

6.3.1 Volume Characteristics

The following is a description of the logical characteristics of the different categories of volumes supported by the system.

Characteristics which are displayed by the DHW or DC operator command are denoted below in upper case.

6.3.1.1 Disks

When the system first tries to access a disk, it looks for the disk volume label (which contains the volume name). If the volume label is present and in the correct format, the disk is identified as native; otherwise it is considered as non-standard (NSTD).

The system can recognize several categories of native disks:

1. **SYSTEM disk**

The SYSTEM disk is recognized as such because it is the disk from which Storage Load is performed; it supports the SYS.BKST0 file. It cannot be removed from the configuration by the MODIFY_HARDWARE_CONFIGURATION (MDHW) command.

2. **RSDT disks (resident disk)**

The operator may define a disk as resident during System Loading. RSDT disks are disks which, like the SYSTEM disk and the backing store disks, are required in the configuration. (See question IL01 in Chapter 3 *Starting and Terminating a GCOS 7 Session*).



NOTES:

1. There is no necessity to define any RSDT volume, provided that all the system files are held on the SYSTEM disk and provided that there is enough space for the temporary files.
2. RESIDENT disks are invariably allocated to X1 and to X1 only. It is therefore not possible to see who are the users of files on such disks.
3. **BKST disks (backing store disks)**
These disks support backing store files. They are declared as BKST, LIB, PVMF or TVMF during system loading. (See question IL01 in Chapter 3 *Starting and Terminating a GCOS 7 Session*).
4. **SRV disk (service disk)**
The service disk contains SPOS service files (SYS.FW.*) used at system initialization time on DPS7000/An systems. It is handled by GCOS 7 like BKST disks.

NOTE:

The SYSTEM, RSDT, BKST and SRV disks cannot be mirrored disks (refer to Section 3). They are allocated to X1 (system) and must not be put in WRITE PROTECT, STANDBY or be powered off. They must keep available to the system throughout the GCOS 7 session.

5. SHARED disks

These disks are shared between coupled systems. They are declared during System Loading or dynamically with the MDHW command. SHARED disks are allocated to X1,

6. Other disks

User disks. They contain user files.

In addition to the above characteristics, the DHW or DC command shows:

- NVTOC for volumes having the FBO volume format which are not high reliability disks
- HRD, followed by the reliability characteristic for high reliability disks. For RAID-1 disks, RD1 is displayed, for RAID-S disks, RDS is displayed.

On a system configured for Mirror Disks usage, the MIRROR attribute and specific MIRROR states are displayed by DHW or DC for mirrored disks. Refer to the *Mirror Disks User's Guide*, this option is no more available on DPS 7000/XTA platforms.



6.3.1.2 Tapes

When the system finds that a tape contains standard tape labels (which contain the volume name) in the correct format, it treats the tape as native; otherwise it is considered as non-standard (NSTD).

Native reel tapes can use either EBCDIC or ASCII recording code. A native cartridge tape is a cartridge tape with a standard volume label in EBCDIC recording code. Cartridge tapes with a label in ASCII recording code are processed as non-standard tapes. Tapes may have various physical characteristics (density, etc.) which can vary a great deal from one tape to another. It is therefore essential that it be premounted/mounted on a drive which has the corresponding physical characteristics. If not, or if nothing at all can be read on the tape, it is treated as a non-standard volume (but messages displayed on the operator console show UNKNOWN VOLUME).

Native tapes belong to one of two types:

1. **WORK tapes**

These are created by the VOLPREP utility. They are recognized as WORK tapes by their tape labels. They may also be prepared with the PREPARE_TAPE (PRPTP) GCL command. (See the *IOF Terminal User Reference Manual*.)

Any job may ask for a WORK tape. A suitable tape is then mounted by the operator and is normally assigned only for the duration of that job. After this it reverts to a pool of WORK tapes which is controlled by the Operations Manager.

If a WORK tape is used for dynamic permanent file extension it loses its WORK quality. The operator is informed by the following message:

```
FP01 dvmn ron.ssn TAG volnm FOR EFN efn
```

The operator should then note that the tape belongs to the indicated file and inform the user accordingly.

A work tape must be in WRITE PERMIT, otherwise the following message is displayed:

```
(DV11) hh.mm dvmn MOUNT "WORK" VOLWR FOR ron
```



2. User tapes

These are created by user jobs and are used to store permanent files. These tapes belong to the user who created them either until the expiration date (specified in the tape labels) has been reached, or until they are released by a user action.

NOTE:

Any user tape whose expiration date has been reached can be overwritten by a job which specifies the correct volume name.

6.3.2 Premounting, Mounting and Dismounting

6.3.2.1 Premounting

The operator is able to save computer time by anticipating job requests, by premounting the volume before the job has asked for it.

For removable volumes, the volume is loaded on any available device (with the required characteristics) and the device is set to READY; for non-removable volumes, the device is set to READY. The system now tries to recognize the volume and it may or may not succeed:

1. If the volume has been recognized, the following message is displayed:
2. If the volume has not been recognized, the system replies with one of the following:

```
(DV21) hh.mm dvnm PREMOUNTED volnm [NVTOC | ...]
```

```
(DV21) hh.mm dvnm PREMOUNTED [volnm] NSTD [FBO] [36T]  
DV24 dvnm UNKNOWN VOLUME
```

The name of the volume must be given by the NAME_VOLUME command. After that the following message is displayed:

```
(DV21) hh.mm dvnm PREMOUNTED volnm NSTD [36T]
```

NOTE:

The NAME_VOLUME command is mandatory, even if a volume name appeared in the displayed message. The volume name given in the NAME_VOLUME command can be a new one or the same one.



6.3.2.2 Normal Mounting

The following does not apply to non-removable volumes, Cartridge Tape Libraries and Any Media mode for tapes.

A job is aborted if it asks for more devices, either IN (inserted in the configuration) or OUT (removed), than there are in the system. It can be queued if the devices exist, but are not available by now (allocated or OUT).

When there is an available device, if a job requires a volume and the operator has not premounted it, the following message is displayed:

```
hh.mm dvmn MOUNT [volnm] ["WORK"] [NSTD] ... FOR ron
```

(WORK indicates that a native WORK tape is required).

The following information can also appear:

- density,
- S (streamer reel tape),
- C (cartridge tape with compaction),
- IMP (when mounting on that device is imperative),
- VOLWR (WRITE PERMIT),
- PROTECT (WRITE PROTECT).

The operator should then take the following actions:

1. If necessary, dismount a volume currently loaded on any device.
2. Mount the requested volume on any device of the same type and with compatible characteristics.
3. Switch the device to READY.
4. For non-standard volumes, issue a NAME_VOLUME command.

If the mounting request cannot be satisfied, the operator has to answer the mounting request with a CANCEL_REQUEST command. This causes the job to be aborted, unless the corresponding file was an 'optional' file.

NOTE:

It can occur that a DC command is followed by the message: "dvmn MOUNT volnm ... FOR ronlist" while there has been no request to mount a volume. This occurs when DEFER has been used in the ASSIGN command, or in the case of a dynamic ASSIGN, or for a WORK tape.



6.3.2.3 Mounting of Non-removable Volumes

Usually, all non-removable volumes are premounted. It can nevertheless happen that some devices are on STANDBY, or have been removed from the configuration.

For STANDBY devices, the processing is similar to the one described for removable volumes (if the device has been previously READY, which allows the system to know the volume name): a MOUNT message is displayed, which can be answered by switching the device to READY, or by a CANCEL_REQUEST command.

There are specific operating rules for mirrored disks concerning MOUNT requests and CANCEL_REQUEST commands. Please refer to the *Mirror Disks User's Guide*.

For OUT devices (or devices which have never been READY), the job can be queued. If there are no such devices, it is aborted, because that means that a wrong volume name has been given. The system knows the volume names of all the volumes which can be present in the configuration.

6.3.2.4 Mounting Cartridge Tape Library Volumes

Sites using the CLC (Cartridge Library Client) component

This component is not available on DPS 7000/XTA platforms.

If the CLC component is operational, mounting is performed automatically by the system. (For non-standard cartridge tapes, the system automatically performs the equivalent of a NAME_VOLUME). If the mount request cannot be satisfied because the volume is not present in the Cartridge Tape Library, or because the volume cannot be overwritten, the system automatically cancels the mount request, which causes the step to be aborted.

If the CLC (Cartridge Library Client) component is not operational, display the CLC state with the DISPLAY_CLC (DCLC) command and then, depending on the information displayed, either:

- start the CLC component with the START_CLC (SCLC) command, if it has not already been started,
- or establishes communications with the Cartridge Library Server (CLS), if the CLC has already been started.



Sites using the CLX (Cartridge Library Unix) component

If the CLX component is operational, mounting is performed automatically by the system. (For non-standard cartridge tapes, the system automatically performs the equivalent of a NAME_VOLUME). If the mount request cannot be satisfied because the volume is not present in the Cartridge Tape Library, or because the volume cannot be overwritten, the system either cancels the request or waits for operator intervention. This action depends on the declaration made in the configuration file CLX_CONFIG.

If the CLX (Cartridge Library Unix) component is not operational, display the CLX state using the command DCLX. If necessary, start CLX with the SCLX command. Starting CLX requires OPEN 7 to be operational.

For more information about the CLX system, see the *CTL-UNIX Server User's Guide*.

6.3.2.5 Mounting of Tapes in Any Media Mode (media = *)

NOTE:

This mode is not allowed for cartridge tapes in a Cartridge Tape Library.

Normally user programs specify the names of the volumes to be used and the operator receives the requests specifying the volnm of the volume to be mounted. In some cases the user program is ignorant of the number of volumes, or of the volnms which are to be used. The user may then use the "MEDIA=*" parameter in the ASSIGN JCL statement specifying that the responsibility and the order of mounting are left entirely to the operator. Each volume within a set of volumes must have the same device attributes.

The volumes to be used (possibly restricted to 1) are delivered to the operator as a set of ordered volumes, each of which has an external identification, that reflects the order in which they are to be mounted by the operator.

The system asks the operator to mount the first volume of the set (IMP shows that the mounting on that device is imperative):

```
*hh.mm dvnM MOUNT *1 ... IMP ... FOR ron.
```

The operator should then take the following actions:

1. If necessary, dismount the volume currently loaded on the required device
2. Mount the first volume of the required set on the required device
3. Switch the device to READY.



After processing the first volume of the set, the system asks the operator to mount the second volume in the same way:

```
*hh.mm dvnM MOUNT *2 ... IMP ... FOR ron.
```

The operator then proceeds as previously stated until the last volume of the set has been processed. A CANCEL_REQUEST command specifies that the requested volume is not to be mounted and the running job step is to end normally.

Whenever possible the system checks that the mounting sequence is correct (multivolume file) or that the user-specified file is found on the volume. When the system detects a problem, the operator is asked to confirm the mounting action:

```
nn/FP04 dvnM vol-occur-nb. VOLUME SEQUENCE CORRECT?
```

The possible answers are:

- | | |
|-----|---|
| YES | The operator confirms that the nth volume of the set has been mounted, and that the volume mounting sequence is correct. Depending on the case, the user program is notified or the operator is asked to mount the next volume. |
| NO | The operator made a mounting error; therefore a mount request for the correct volume is re-issued by the system. |

6.3.2.6 Dismounting

A volume can be dismounted (or removed from the configuration) when the device is not allocated. The dismounting operation is not acknowledged by the system. The DISPLAY_HARDWARE_CONFIGURATION command may be used to ascertain whether a particular device is deallocated.

A user may specify in the JCL that a tape is to be dismounted after its use, in which case the tape is fully rewound.



6.3.3 Incidents

Incidents involving disk or tape devices can be of the following nature:

- An input/output operation attempt is unsuccessful, but the affected device remains in the READY state.
- Transition of the device to STANDBY (or power off). This may result either from a physical action (the operator pressing the appropriate button and/or dismounting the volume) or from a hardware/firmware problem.
- WRITE PROTECT violation on disk or tape. This leads to a system request for operator action which might involve the START_DEVICE command or the CANCEL_REQUEST command.

6.3.3.1 Input/output Operation Incidents

Several retries (the actual number is dependent upon the type of operation) are attempted by hardware and software to complete an unsuccessful I/O operation in which the affected device has remained READY.

If a retry is successful:

- There is no notification, although the original incident is reported if the START_DEVICE_TRACE command (described later in this section) was entered.
- The event is recorded by Error Logging in SYS.ERLOG file.
- There is no effect on the step to which the I/O operation applies (the retries are invisible to the step).

If all retries fail, this is an irrecoverable I/O error. The step attempting the operation is notified of the irrecoverable condition and is then usually aborted.

NOTE:

An irrecoverable I/O error involving a system file can cause a system crash.



6.3.3.2 Transition to STANDBY (or power off)

NOTE:

a STANDBY push button does not exist on all devices.

A disk or tape device which is allocated to a user or to the system may be switched to the STANDBY (or power off) state either by an operator action or as the result of hardware failure. Three cases may occur:

1. The drive contains a SYSTEM or BKST disk. The following message is displayed:

```
AV04 dvnm SWITCH volnm TO READY
```

The operator must switch the device to READY. If this is successful, no further action is requested and no message is displayed.

If this is unsuccessful AV05 is displayed:

```
AV05 dvnm MOUNT volnm IMPERATIVE
```

There is clearly a hardware problem with the device. Hardware initialization must now be performed (see also *Hardware Failure*).

NOTE:

As long as the SYSTEM disk is in STANDBY, system activity is limited by its inability to access system files.

2. The device is in a Cartridge Tape Library.

The following message is displayed:

```
DW12 dvnm RECOVER volnm FOR ron
```

The system then automatically performs a CANCEL_REQUEST, and the step which is using the volume is aborted.

3. Otherwise, the following message is displayed:

```
DV12 dvnm RECOVER volnm FOR ron
```

The operator has a choice at this point and may:

- a. Try to recover

For removable volumes, remount the volume, either on the same device or on any other device of the same type and characteristics. For non-removable volumes, switch the device back to READY.

If recovery is possible, the I/O operation is retried. Tapes are repositioned before a retry except during open or close operations or if they are used for a VOLSAVE, VOLREST or FILREST utility.

**NOTE:**

If the mounting request for that volume was IMP (imperative), it must be mounted again on the same device.

b. Give up recovery

Use the CANCEL_REQUEST command to indicate this to the system. All the steps which have allocated the volume are usually aborted, since they all try to access the unavailable volume.

It is also possible to use the CANCEL_JOB (CJ) command to kill the job for which the volume is allocated. Be careful; using CJ leaves the device in STANDBY if the cancelled job was the only one which had allocated the volume. If there are others jobs present, the device remains in the RECOVER state.

There are specific operating rules for mirrored disks concerning RECOVER requests, CANCEL_REQUEST and CANCEL_JOB commands. Please refer to the *Mirror Disks User's Guide*.

6.3.3.3 Write Protect on tapes

If a reel tape is mounted without its permit ring, it is said to be in WRITE PROTECT and no writing can be done on the tape. If the permit ring is present, the tape is in WRITE PERMIT.

For a cartridge tape, the protection depends on the position of the file-protect selector on the side of the cartridge. A cartridge tape from a Cartridge Tape Library is in WRITE PROTECT when a white dot appears in a dark background above the thumbwheel.

1. If PROTECT has been indicated for the tape, it is mandatory to mount the tape in the WRITE PROTECT state. If this is not done, the following message is displayed:

```
* DV06 dvm SWITCH volnm FOR ron TO PROTECT
```

2. If a job attempts to write on a tape in WRITE PROTECT the following message is displayed:

```
DV07 dvm SWITCH volnm FOR ron TO PERMIT [AND REPLY (SDV,  
CR, ...)]
```

The operator action depends on whether or not it is normal (that is requested by the user or not) for this tape to be in WRITE PROTECT:

- a. Abnormal WRITE PROTECT (not requested by the user).



The operator:

- dismounts the tape,
- remounts it in the WRITE PERMIT state,
- if requested, indicates this to the system using the START_DEVICE command..

No further action is required and no message is displayed.

b. Normal WRITE PROTECT (as requested by the user, the mounting request indicates PROTECT).

The operator indicates to the system that the tape will not be set to WRITE PERMIT, by the CANCEL_REQUEST command. In this case, the step which attempted to access the tape is aborted.

NOTE:

For a cartridge tape in a Cartridge Tape Library, the following message is displayed:

```
DW07 dvnm SWITCH volnm FOR ron TO PERMIT
```

But the system then automatically performs a CANCEL_REQUEST command and the step which attempted to access the tape is aborted.



6.4 Printer Operations

The printer operations described in this section are general and depend little on the type of the printers.

There are:

- belt printers, which use a print belt as their print medium.
Print belts can be changed so that a different character set can be used. The PR54 (PRU4112, PRU4113) and PR88 (PRU4120, PRU4121) are belt printers.
These printers are no more supported since end of 2000.
- dot matrix printers, which use dot matrix technology.
Characters (and graphics) are formed by assembling groups of dots in matrices (hence the term "dot matrix").
 - PR90, (not supported on DPS 7000/XTA platfprms)
 - PR800, 900,
 - PR 701, 801, 901,
 - PR702, 802, 902,
 - PR702A, 802SA, 902SA

Printer configuration parameters are controlled via the SYS.URCINIT system file:

- character set to use (this corresponds to a belt for belt printers),
- form control information,
- the physical vertical format unit (VFU) used by some printers (see also Section 6.4.1, *Paper Form Operations*).

For more information on the SYS.URCINIT file, see Appendix D *The SYS.URCINIT Printer Control File*.

For more information on the Dot matrix printers, refer to the *Printer User's Guide*.

The alignment of a paper form can be visually checked using the Print test routine.

These aspects of printer operations are described below under the headings:

- Paper Form Operations.
- Print Belt Operations.
- Print Incidents.

NOTE:

During System Loading the printer is handled in a manner different from that described below.



6.4.1 Paper Form Operations

An installation may define its own in-house paper form standards. Each type of paper form has its own characteristics: page length, perforations, preprinting, etc. Each of these form types is entered into a system file (called SYS.URCINIT) and is allocated a 4-character name by which it is known both to the users and to the system.

Thus a user job can call for a particular paper form to be used for its output. If this paper form is not mounted at the time of output - the system remembers which form is mounted - the system asks the operator to mount the correct form.

Once the correct form is mounted, output begins. The handling of page lengths and paper movement is performed automatically by the system which holds all the necessary details in its SYS.URCINIT file.

6.4.1.1 Paper Premounting

To premount a paper form, perform the following actions:

1. Switch the printer to STANDBY (if it is not already in STANDBY).
2. Mount the paper.
3. Preferably leave the printer in STANDBY (see Note below). If the printer is already allocated, switching it to STANDBY causes an incident, and the message

```
DV12 dvrnm RECOVER volnm FOR ron
```

is displayed. This request is periodically repeated until it is either satisfied or cancelled.

At loading, the system treats the printer (if it is switched to READY) as if the standard paper were mounted, and the message:

```
hh.mm dvrnm PREMOUNTED 0000
```

is displayed.

NOTE:

If, after changing the paper, the printer is switched to READY, the system continues as if no paper change had occurred. Therefore, it may well be advantageous to leave the printer switched to STANDBY so that a mounting request will be displayed if the wrong paper is mounted.



6.4.1.2 Paper Mounting

Normal Case

When a paper form is required by a job or the Output Writer (see Note below) and is not already mounted, the system displays the message:

```
hh.mm dvnm MOUNT volnm FOR ron
```

and switches the printer to STANDBY with the PRINT CHECK indicator on. The operator should perform the following actions:

1. Mount the paper.
2. Then press the START button to switch the printer to READY.

Abnormal Case

The following abnormal case may occur during paper mounting:

The request cannot be satisfied (for example, the paper form is not available). The CANCEL_REQUEST command should be used. If the paper form was requested by the Output Writer, the WRITER service job is terminated and the output being processed is left in the WAIT state. If it was requested by a user, the user step is aborted.

NOTE:

If the user requests a print belt together with a paper form, mounting messages are issued for both; thus the required belt can then be mounted at the same time as the paper.



Print Test Routine

The Print Test routine allows the operator to verify that the paper is correctly aligned. It is started using the TEST_DEVICE (TSTDV) command (see Chapter 8 *Description of the Operator Commands*).

NOTE:

On some printers, there is a PRINT TEST pushbutton to start the Print Test routine.

The standard Print Test (a user can specify another Print Test, but it must be stored in the SYS.URCINIT file) is the printing of a full line under the paper fold. The line is:

```
SYS.URCINIT CREATION DATE mm/dd/yy  
EEE.....EEE
```

During the mounting of a paper form the system sets the printer to STANDBY with the PRINT CHECK indicator on. Once the paper is mounted, the operator starts the Print Test routine (TSTDV command or PRINT TEST pushbutton), with the following effect:

- The system sets the printer to READY.
- The Print Test routine is executed.
- The system resets the printer to STANDBY with PRINT CHECK on and displays the message:

```
DV45 dVnm SWITCH TO READY
```

The operator can then visually check that the mounting is correct:

- If the paper form alignment is not correct, it can be corrected and Print Test can be used again.
- If it is correct, switch the printer to READY.

The Print Test routine can be performed at any time by switching the printer to STANDBY (if necessary) and using the TSTDV command or pressing the PRINT TEST button. The result is the same as that described above.



6.4.2 Printer Incidents

Line Printer Incidents

Most printer incidents (end of paper, etc.) result in the printer being set to STANDBY. Some more serious incidents result in the printer being switched to POWER OFF.

When an incident occurs, and the printer is being used, the following message is displayed:

```
DV12 dvm RECOVER volnm FOR ron
```

The operator may either:

- Try to recover

Correct any obvious cause (end of paper, paper jam, printer open), then make the printer READY. Printing resumes with the line at which the incident was detected, which may result in a line or two being duplicated. If the incident involves a disengaged tractor, press the SKIP button prior to making the printer READY, to avoid a line overlapping the previously printed line.

- Give up the recovery

Inform the system of this choice through the CANCEL_REQUEST command. If the printer is being used by a step, that step is aborted. This is also used to ask the Output Writer for a page recovery (see Section 5.8.1.9 *Output Writer Incidents*). For POWER OFF conditions the job reacts as if a CANCEL_REQUEST command had been issued.

In certain cases (normally after a POWER OFF incident) the system may indicate that it is unable to recognize the printer by giving the message:

```
DV17 dvm MEDIA RECOGNITION IMPOSSIBLE  
RC = XXXXXXXX, siu module_id,return-code
```

If this happens, press the STOP and then the START pushbuttons on the printer.

NOTE:

During the printing of the PRLOG utility report on a non-standard paper form (i.e., one whose length is not 11 inches), the printer may enter the STANDBY state with the NOT SYNCHRONIZED indicator lit. Such an incident can be avoided by redefining the printing parameters of the PRLOG utility REPORT file. For more details, refer to the Note at the end of Section 6.5.3 *Error Logging*.



6.5 Hardware Incidents

Hardware incidents, and the way in which the system reacts to them, are discussed below at three levels:

- Device Incidents.
- Peripheral Processor Incidents.
- Error Logging.

6.5.1 Device Incidents

The system provides a set of functions which help in maintaining devices in a good operational state. When incidents do occur, these functions allow the system to handle them with as little damage to user programs as possible.

The handling of incidents on each particular type of device has been described in the corresponding incident paragraph above.

More serious incidents might cause the peripheral processor to enter the "STOP" state - specific operator action is then required.

All incidents on devices, or peripheral processors are recorded by the Error Logging mechanism, for further use by the Service Center. Specific commands are provided for Error logging control (see Section 6.5.3 *Error Logging*).

The MODIFY_HARDWARE_CONFIGURATION command (MDHW) is provided to take a device out of the configuration (for handling by the Service Center, or to prevent its use by the system) and then to put it back in the configuration.

The Display Device Trace mechanism provides an operator with a record of device marginal conditions.

On-Line Test and Diagnostics programs are provided to help detect and diagnose problems. For further information see the *Operator Reference Manual*. No OLTD programs are available on DPS 7000/XTA platforms.

The operator is encouraged to use these programs before informing the Service Center of device incidents.



6.5.2 Peripheral Processor Incidents

If an incident occurs on one of the peripheral processors, the system switches the processor to STOP and reacts as follows:

- Failure of the Mass Storage Processor or the IOP/PXU. When the SYSTEM disk has a single access path through the MSP, the MSP or IOP/PXU controls the drive the system disk is mounted on.

Because the system disk can no longer be accessed, no more useful work can be performed and the following message is displayed:

```
* AV01 GCOS : CRASH. NO DUMP followed by:
```

```
* AV01 CHANNEL EXCEPTION ON MCnn PCpp
```

- Magnetic Tape or Mass Storage Processor failure. The following message is displayed:

```
* hh.mmm dvrnm PSI pcnb FAILED
```

It is still possible for the system to run, but the devices controlled by the failed processor cannot be used; if these devices have a single access path, they are automatically removed from the configuration.

NOTE:

If only the console keyboard has failed, there is no indication on the SOP.

In any of the above cases the operator should:

1. Note any abnormal indications on the SOP and other Maintenance Panels.
2. Take any necessary action as described in the appropriate *Operator Reference Manual*.
3. Try to recover the system. Perform a complete initialization, if necessary.
4. Run the available operator tests and diagnostics. Contact the Service Center.



6.5.3 Error Logging

Every event which could be of help to the Service Center is recorded in a system file called SYS.ERLOG.

Examples of such events are the successful retry of an I/O operation or a marginal condition detected by a peripheral.

It is essential to maintain enough free space in the SYS.ERLOG file, if not, information may be overwritten and lost.

When half the file is full, the following message appears

```
EL01 SYS.ERLOG FILE 50% FULL
```

The PRLOG utility is now used to print the information stored in the SYS.ERLOG file and to free the corresponding space.

If the PRLOG utility is specified in the Config. job (see the *GCOS 7 System Installation Configuration and Updating Guide*), you can specify a percentage other than 50% and/or specify a command to be executed automatically; for example, `EJR PRLOG LIB=SYS.HSLLIB VALUES = ('PRLOG-options')`. In this case, the system automatically prints the contents of the SYS.ERLOG file.

If the PRLOG utility is not specified in the Config job, the following message appears

```
EL03 RUN PRLOG
```

The operator must now run the PRLOG utility and save its output for later analysis by the FER.

If the PRLOG utility is not run before the SYS.ERLOG file is full, the message:

```
EL01 SYS.ERLOG FILE FULL
```

is displayed, and if PRLOG is still not run before SYS.ERLOG has been completely overwritten then the message:

```
EL01 SYS.ERLOG FILE FULL DATA LOST UP TO date
```

is displayed, indicating all the information in the SYS.ERLOG file up to the indicated date has been lost.

Certain abnormal situations cause the following messages to be displayed:

```
EL01 SYS.ERLOG FILE UNAVAILABLE
```

This message, indicating an error (probably on the SYS.ERLOG file), causes Error Logging to work in "Reduced Mode". In this state PRLOG cannot be run and error messages are displayed automatically on the console.



It can also indicate that due to an error the whole set of Error Logging functions are unavailable. This is called "Trace Only Mode" because all events that would have been stored on the SYS.ERLOG file are displayed on the console instead.

```
EL01 SYS.ERLOG FILE REINITIALIZED
```

This message is issued at GCOS 7 initialization when the SYS.ERLOG file is not initialized or when the disk supporting the SYS.ERLOG file does not match the system disk under which the file has been initialized (this may happen after volume duplication). This means that the previously stored events are erased as they do not relate to the current system.

The operator can write comments on the SYS.ERLOG file using the SEND_ERROR_LOG (SNDRLOG) command. The DISPLAY_ERROR_LOG (DERLOG), MODIFY_ERROR_LOG (MDERLOG), TERMINATE_ERROR_LOG (TERLOG) and START_ERROR_LOG (SERLOG) commands are used by the Service Center. See Chapters 7 and 8 *Description of the Operator Commands*.

NOTE:

During the printing of the PRLOG utility report on a paper form whose length is not 11 inches, the printer may enter the STANDBY state with the NOT SYNCHRONIZED indicator lit. Such an incident can be avoided by redefining the printing parameters of the PRLOG utility REPORT file.

For example, when using a paper form 12 inches long, the JCL values of the standard PRLOG Job in the SYS.HSLLIB should be modified as follows:

```
VALUES='PROLOG OPTIONS', FORMHT=96, FF1=94, HOF=8, PRDEN=8;  
STEP H_PRLOG .....;  
    STEP H_PRLOG .....;  
    DEFINE REPORT PRINTER=(FORMHT=96 FF1=94 HOF=8);  
    ENDSTEP;
```

For other paper form heights, contact the Service Center.



6.6 Monitoring Device Incidents (Device Trace)

For each device in the configuration, a threshold value is declared at system generation which defines the acceptable number of logged events on a given device. When that threshold is reached, the message:

```
AV16 dvm log-ev-pct % EVENTS (ev-nb/io-nb)
```

is displayed, giving the number of logged events over the number of input/output operations (and the corresponding percentage).

If the threshold declared at generation proves to be unrealistic, the value can be revised for a device class through the `MODIFY_DEVICE_TRACE` command. The operator can ascertain the present value set for a device class (or all classes) by using the `DISPLAY_DEVICE_TRACE` command, which also displays the number of events detected so far for each device of the class (or all classes). The number of events can be reset to zero by use of the `MODIFY_ERROR_LOG` command or by use of the `PRLOG` utility.

It may also be appropriate at some time (with the help of the Service Center, for instance) to use the `START_DEVICE_TRACE` command, which requests an information message each time an event is detected. A `TERMINATE_DEVICE_TRACE` command discontinues this activity.

The `START_DEVICE_TRACE` command causes the following message to be displayed for the original I/O incident and for the last retry of the I/O operation.

```
AV17 dvm IOCC = ioccv1 EV = evv1 PSB = psbv1 DBS = dsbv1
```

The Device Trace commands are described in Chapter 8 *Description of the Operator Commands*.





7. Description of the Operator Commands (Part 1)

7.1 Introduction

7.1.1 GCL Operator Commands

The whole set of operator commands has been redesigned using the GCL syntax.

This section describes the GCL commands most used by the GCOS 7 system console operator.

The command parameters can be specified in two ways:

- as a keyword parameter, that is, a keyword, followed by an equals sign and the parameter value,
- as a positional parameter, that is, just a parameter value with no keyword.

NOTE:

The conventions, notations and rules used to define the syntax of GCL commands are in the Preface.



7.1.2 Network Commands

The following Network Operator commands are described briefly in Section 4.8 *Managing the Network*.

They are fully described in the *GCOS 7 Network User's Guide*.

```
DISPLAY_MAILBOX (DMBX)
DISPLAY_NETWORK (DNET)
DISPLAY_TELECOM_SERVER (DTSVR)
MODIFY_NETWORK (MDNET)
MODIFY_QUEUE_EDIT (MDQED)
SEND_QUEUE (SNDQ)
START_TELECOM_SERVER (STSVR)
START_TELECOM_TRACE (STTR)
TERMINATE_TELECOM_SERVER (TTSVR)
TERMINATE_TELECOM_TRACE (TTTR)
```



7.1.3 TDS Commands

The following TDS commands are described in the *TDS Administrator's Guide*:

```
ALLOW_NEW_TDS_COR (ALNTC)
CANCEL_TDS_COR (CTC)
CANCEL_TDS_SPAWN (CTSPW)
CHECK_TX_CONVERSATION (CKTXCONV)
CLOSE_COR_POOL (CLCPPOOL)
CLOSE_TDS_FILE (CLTF)
DISPLAY_TDS (DTDS)
DISPLAY_TX (DTX)
DUMP_TDS (DPT)
EXEC_TDS (EXECT)
HELP
LIST_COR_POOL (LSCPOOL)
LIST_TDS_COR (LSTC)
LIST_TDS_FILE (LSTF)
LIST_TDS_SPAWN (LSTSPW)
LOAD_TDS_IQS (LDTIQS)
LOAD_TDS_MEMORY (LDTMEM)
MODIFY_COR_POOL (MDCPOOL)
MODIFY_TDS (MDTDS)
MODIFY_TDS_MOT (MDTMOT)
MODIFY_TDS_RESTART_OPTION (MDTRSO)
MODIFY_TX
OPEN_COR_POOL (OCPOOL)
OPEN_TDS_FILE (OTF)
PREVENT_NEW_TDS_COR
SEND_TDS (SNDT)
SEND_TDS_USER (SNDTU)
SUPERVISE_TDS
TERMINATE_TDS (TTDS)
UNLOAD_TDS_IQS (UNLDTIQS)
UNLOAD_TDS_MEMORY (UNLDTMEM)
```



7.2 Alphabetical List of GCL Commands

The commands described in this section are listed below with an indication of the type of operators to which they apply. The notation is as follows:

A Any operator (IOF)
M Main operator only
S Both Main and Station operators

(A) ATT (ATTACH_CATALOG)
 (A) ATTACH (ATTACH_CATALOG)
 (A) ATTACH_CATALOG

(A) CANCEL_DUMP
 (A) CANCEL_JOB
 (A) CANCEL_OUTPUT
 (M) CANCEL_REQUEST
 (A) CANCEL_TERMINAL_OUTPUT
 (A) CANCEL_USER_REQ
 (A) CDP (CANCEL_DUMP)
 (A) CJ (CANCEL_JOB)
 (M) CNDIM (CONNECT_DIMENSION)
 (M) CNFUNC (CONNECT_FUNCTION)
 (M) CNLD (CONNECT_LOAD)
 (A) CO (CANCEL_OUTPUT)
 (M) CONNECT_DIMENSION
 (M) CONNECT_FUNCTION
 (M) CONNECT_LOAD
 (M) CR (CANCEL_REQUEST)
 (M) CRCXGEN (CREATE_COMPLEX_GENERATION)
 (M) CRDIM (CREATE_DIMENSION)
 (M) CREATE_COMPLEX_GENERATION
 (M) CREATE_DIMENSION
 (M) CREATE_FILTER
 (M) CREATE_FILTER_SET
 (M) CRFLT (CREATE_FILTER)
 (M) CRFLTST (CREATE_FILTER_SET)
 (A) CTO (CANCEL_TERMINAL_OUTPUT)
 (A) CUR (CANCEL_USER_REQ)

(A) DARS (DISPLAY_ARS)
 (A) DASGF (DISPLAY_ASSIGNED_FILES)
 (M) DASGMIR (DEASSIGN_MIRLOG_FILE)
 (A) DC (DISPLAY_CONFIGURATION)
 (A) DCLC (DISPLAY_CLC)
 (M) DCLOG (DISPLAY_CONSOLE_LOG)
 (M) DCLX (DISPLAY_CLX)
 (M) DCX (DISPLAY_COMPLEX)
 (M) DDIM (DISPLAY_DIMENSION)



(A) DDJP (DISPLAY_DJP)
(A) DDP (DISPLAY_DUMP)
(A) DDTIME (DISPLAY_DEVIATION_TIME)
(M) DDVTR (DISPLAY_DEVICE_TRACE)
(M) DDYNSVR (DISPLAY_DYNSAVE_SVR)
(M) DEASSIGN_MIRLOG_FILE
(M) DELETE_DIMENSION
(M) DELETE_FILTER
(A) DERLOG (DISPLAY_ERROR_LOG)
(M) DFUNC (DISPLAY_FUNCTION)
(M) DGC (DISPLAY_GCL_CACHE)
(A) DHW (DISPLAY_HARDWARE_CONFIGURATION)
(M) DIOC (DISPLAY_IO_CACHE)
(A) DIOF (DISPLAY_IOF)
(M) DISCONNECT_DIMENSION
(M) DISCONNECT_FUNCTION
(M) DISCONNECT_LOAD
(M) DISDIM (DISCONNECT_DIMENSION)
(M) DISFUNC (DISCONNECT_FUNCTION)
(M) DISLD (DISCONNECT_LOAD)
(A) DISPLAY_ARS (DARS)
(A) DISPLAY_ASSIGNED_FILES
(A) DISPLAY_CLC
(M) DISPLAY_CLX (DCLX)
(M) DISPLAY_COMPLEX
(A) DISPLAY_CONFIGURATION
(M) DISPLAY_CONSOLE_LOG
(A) DISPLAY_DEVIATION_TIME
(M) DISPLAY_DEVICE_TRACE
(M) DISPLAY_DIMENSION
(A) DISPLAY_DJP
(A) DISPLAY_DUMP
(A) DISPLAY_ERROR_LOG
(A) DISPLAY_FUNCTION
(M) DISPLAY_GCL_CACHE
(A) DISPLAY_HARDWARE_CONFIGURATION
(M) DISPLAY_IO_CACHE
(A) DISPLAY_IOF
(A) DISPLAY_JAS
(A) DISPLAY_JOB
(A) DISPLAY_LOAD
(M) DISPLAY_MEMBER
(M) DISPLAY_MIRROR_MODE
(M) DISPLAY_MOT
(A) DISPLAY_OUTPUT
(M) DISPLAY_POWER_TIMER
(A) DISPLAY_REGULATION_PARAMETER
(A) DISPLAY_REQUEST
(M) DISPLAY_RESTART_OPTIONS
(M) DISPLAY_RMS
(M) DISPLAY_SECURITY_OPTIONS



(M) DISPLAY_SERVICE
(M) DISPLAY_SSI (DSSI)
(A) DISPLAY_STORAGE_FACILITIES
(A) DISPLAY_SUBMITTER
(M) DISPLAY_SYSTEM_TRACE
(A) DISPLAY_TERMINAL_OUTPUT
(A) DISPLAY_TERMINAL_WRITER
(A) DISPLAY_TIME
(A) DISPLAY_USER_REQ
(M) DISPLAY_XL_CLASS
(A) DJ (DISPLAY_JOB)
(A) DJAS (DISPLAY_JAS)
(A) DLD (DISPLAY_LOAD)
(M) DLDIM (DELETE_DIMENSION)
(M) DLFLT (DELETE_FILTER)
(M) DMB (DISPLAY_MEMBER)
(M) DMIR (DISPLAY_MIRROR_MODE)
(M) DMOT (DISPLAY_MOT)
(A) DO (DISPLAY_OUTPUT)
(A) DPR (DPRINT)
(A) DPRINT
(M) DPWR (DISPLAY_POWER_TIMER)
(M) DPWRM (DISPLAY_POWER_TIMER)
(A) DR (DISPLAY_REQUEST)
(M) DRMS (DISPLAY_RMS)
(A) DRP (DISPLAY_REGULATION_PARAMETER)
(M) DRSO (DISPLAY_RESTART_OPTIONS)
(A) DS (DISPLAY_SUBMITTER)
(A) DSF (DISPLAY_STORAGE_FACILITIES)
(M) DSO (DISPLAY_SECURITY_OPTIONS)
(M) DSRV (DISPLAY_SERVICE)
(M) DSSI (DISPLAY_SSI)
(M) DSYSTR (DISPLAY_SYSTEM_TRACE)
(A) DTO (DISPLAY_TERMINAL_OUTPUT)
(A) DTM (DISPLAY_TIME)
(A) DTW (DISPLAY_TERMINAL_WRITER)
(A) DUR (DISPLAY_USER_REQ)
(M) DXLC (DISPLAY_XL_CLASS)

(A) EFTR (ENTER_FILETRANS_REQ)
(A) EJ (ENTER_JOB_REQ)
(A) EJR (ENTER_JOB_REQ)
(A) ELTR (ENTER_LIBTRANS_REQ)
(A) ENTER_FILETRANS_REQ
(A) ENTER_JOB_REQ
(A) ENTER_LIBTRANS_REQ
(A) EXCLS (EXEC_CLS_COMMAND)
(A) EXDIR (EXECUTE_DIRECTIVE)
(A) EXEC_CLS_COMMAND
(M) EXEC_CLX_VOLRPT (VOLRPT)



(M) FJ (FORCE_JOB)
(S) FO (FORCE_OUTPUT)
(M) FORCE_JOB
(S) FORCE_OUTPUT
(A) FORCE_SERVER_TIMEOUT
(S) FORCE_USER_REQ
(A) FSVRTO (FORCE_SERVER_TIMEOUT)
(S) FUR (FORCE_USER_REQ)

(M) GET_CLX_FILE (GTCLXF)
(M) GTCLXF (GET_CLX_FILE)

(M) HGC (HOLD_GCL_CACHE)
(A) HJ (HOLD_JOB)
(A) HO (HOLD_OUTPUT)
(M) HOLD_GCL_CACHE
(A) HOLD_JOB
(A) HOLD_OUTPUT
(A) HOLD_TERMINAL_OUTPUT
(A) HOLD_USER_REQ
(A) HTO (HOLD_TERMINAL_OUTPUT)
(A) HUR (HOLD_USER_REQ)

(M) INIT_SYSTEM_TRACE
(M) ISYSTR (INIT_SYSTEM_TRACE)

(A) LIST_DUMP
(M) LIST_FILTER
(A) LSDP (LIST_DUMP)
(M) LSFLT (LIST_FILTER)

(A) MAIL
(S) MDC (MODIFY_CONFIGURATION)
(A) MDCLC (MODIFY_CLC)
(M) MDCLX (MODIFY_CLX)
(M) MDDIM (MODIFY_DIMENSION)
(M) MDDJP (MODIFY_DJP)
(M) MDDVTR (MODIFY_DEVICE_TRACE)
(M) MDERLOG (MODIFY_ERROR_LOG)
(M) MDFLT (MODIFY_FILTER)
(M) MDHW (MODIFY_HARDWARE_CONFIGURATION)
(M) MDIOC (MODIFY_IO_CACHE)
(M) MDIOF (MODIFY_IOF)
(M) MDIR (MODIFY_INPUT_READER)
(A) MDJ (MODIFY_JOB)
(M) MDLD (MODIFY_LOAD)
(M) MDMIR (MODIFY_MIRROR_MODE)
(M) MDMOT (MODIFY_MOT)
(A) MDO (MODIFY_OUTPUT)
(A) MDOM (MODIFY_OPERATING_MODE)
(M) MDPWR (MODIFY_POWER_TIMER)



(M) MDPWRTM (MODIFY_POWER_TIMER)
(M) MDRSO (MODIFY_RESTART_OPTIONS)
(S) MDST (MODIFY_STATION)
(M) MDSYSTR (MODIFY_SYSTEM_TRACE)
(M) MDTIME (MODIFY_TIME)
(M) MDTM (MODIFY_TIME)
(A) MDTO (MODIFY_TERMINAL_OUTPUT)
(A) MDTW (MODIFY_TERMINAL_WRITER)
(A) MDUR (MODIFY_USER_REQ)
(M) MDV (MODIFY_VOLUME)
(M) MDXLC (MODIFY_XL_CLASS)
(A) MODIFY_CLC
(M) MODIFY_CLX (MDCLX)
(S) MODIFY_CONFIGURATION
(M) MODIFY_DEVICE_TRACE
(M) MODIFY_DIMENSION
(M) MODIFY_DJP
(M) MODIFY_ERROR_LOG
(M) MODIFY_FILTER
(M) MODIFY_HARDWARE_CONFIGURATION
(M) MODIFY_INPUT_READER
(M) MODIFY_IO_CACHE
(M) MODIFY_IOF
(A) MODIFY_JOB
(M) MODIFY_LOAD
(M) MODIFY_MIRROR_MODE
(M) MODIFY_MOT
(A) MODIFY_OPERATING_MODE
(A) MODIFY_OUTPUT
(M) MODIFY_POWER_TIMER
(M) MODIFY_RESTART_OPTIONS
(S) MODIFY_STATION
(M) MODIFY_SYSTEM_TRACE
(A) MODIFY_TERMINAL_OUTPUT
(A) MODIFY_TERMINAL_WRITER
(M) MODIFY_TIME
(A) MODIFY_USER_REQ
(M) MODIFY_VOLUME
(M) MODIFY_XL_CLASS
(M) MSYSTR (MODIFY_SYSTEM_TRACE)
(A) MWENVT (Modify Working Environment)

(M) NAME_VOLUME
(M) NV (NAME_VOLUME)

(M) RELEASE_GCL_CACHE
(A) RELEASE_JOB
(A) RELEASE_OUTPUT
(A) RELEASE_TERMINAL_OUTPUT
(A) RELEASE_USER_REQ
(A) REP (REPLY)



(A) REPLY
(M) RESYNC_MIRROR_SHARED
(M) RGC (RELEASE_GCL_CACHE)
(A) RJ (RELEASE_JOB)
(A) RO (RELEASE_OUTPUT)
(M) RSCMIR (RESYNC_MIRROR_SHARED)
(A) RTO (RELEASE_TERMINAL_OUTPUT)
(A) RUR (RELEASE_USER_REQ)

(M) SARS (START_ARS)
(A) SCLC (START_CLC)
(M) SCLOG (START_CONSOLE_LOG)
(M) SCLX (START_CLX)
(M) SCMSR (START_CMSR)
(M) SDV (START_DEVICE)
(M) SDVTR (START_DEVICE_TRACE)
(M) SDYNSVR (START_DYNSAVE_SVR)
(A) SEND
(M) SEND_ERROR_LOG
(M) SERLOG (START_ERROR_LOG)
(M) SHC (START_HARDCOPY)
(M) SIOC (START_IO_CACHE)
(S) SIR (START_INPUT_READER)
(M) SLD (START_LOAD)
(M) SMB (START_MEMBER)
(M) SNDRLOG (SEND_ERROR_LOG)
(M) SPWRM (START_POWER_TIMER)
(S) SO (START_OUTPUT_WRITER)
(S) SOW (START_OUTPUT_WRITER)
(M) SRMS (START_RMS)
(M) SSRV (START_SERVICE)
(M) SSSI (START_SSI)
(M) SSTR (START_SYSTEM_TRACE)
(M) START_ARS (SARS)
(A) START_CLC
(M) START_CLX (SCLX)
(M) START_CMSR
(M) START_CONSOLE_LOG
(M) START_DEVICE
(M) START_DEVICE_TRACE
(M) START_ERROR_LOG
(M) START_HARDCOPY
(S) START_INPUT_READER
(M) START_IO_CACHE
(M) START_LOAD
(M) START_MEMBER
(S) START_OUTPUT_WRITER (SO)
(M) START_POWER_TIMER
(M) START_RMS
(M) START_SERVICE
(M) START_SSI (SSSI)



(M) START_SYSTEM_TRACE
(A) START_TERMINAL_OUTPUT
(M) START_TERMINAL_WRITER
(A) STO (START_TERMINAL_OUTPUT)
(M) STW (START_TERMINAL_WRITER)

(M) TAKEOVER_MEMBER
(M) TARS (TERMINATE_ARS)
(M) TCLC (TERMINATE_CLC)
(M) TCLX (TERMINATE_CLX)
(M) TCLOG (TERMINATE_CONSOLE_LOG)
(M) TCMSR (TERMINATE_CMSR)
(M) TDVTR (TERMINATE_DEVICE_TRACE)
(M) TDYNSVR (TERMINATE_DYNSAVE_SVR)
(M) TERLOG (TERMINATE_ERROR_LOG)
(M) TERMINATE_ARS (TARS)
(A) TERMINATE_CLC (TCLC)
(M) TERMINATE_CLX (TCLX)
(M) TERMINATE_CMSR
(M) TERMINATE_CONSOLE_LOG
(M) TERMINATE_DEVICE_TRACE
(M) TERMINATE_ERROR_LOG
(M) TERMINATE_HARDCOPY
(S) TERMINATE_INPUT_READER
(M) TERMINATE_IO_CACHE
(M) TERMINATE_LOAD
(M) TERMINATE_MEMBER
(S) TERMINATE_OUTPUT_WRITER
(M) TERMINATE_POWER_TIMER
(A) TERMINATE_RMS
(A) TERMINATE_SERVICE
(M) TERMINATE_SSI (TSSI)
(M) TERMINATE_SYSTEM
(M) TERMINATE_SYSTEM_TRACE
(A) TERMINATE_TERMINAL_OUTPUT
(M) TERMINATE_TERMINAL_WRITER
(M) TEST_DEVICE
(M) THC (TERMINATE_HARDCOPY)
(M) TIOC (TERMINATE_IO_CACHE)
(S) TIR (TERMINATE_INPUT_READER)
(M) TLD (TERMINATE_LOAD)
(M) TMB (TERMINATE_MEMBER)
(S) TOW (TERMINATE_OUTPUT_WRITER)
(M) TPW (MODIFY_POWER_TIMER)
(M) TPWRTM (TERMINATE_POWER_TIMER)
(A) TRMS (TERMINATE_RMS)
(M) TSRV (TERMINATE_SERVICE)
(M) TSSI (TERMINATE_SSI)
(M) TSTDV (TEST_DEVICE)
(M) TSYS (TERMINATE_SYSTEM)
(M) TSYSTR (TERMINATE_SYSTEM_TRACE)



(A) TTO (TERMINATE_TERMINAL_OUTPUT)
(M) TTW (TERMINATE_TERMINAL_WRITER)

(M) UNLMIR (UNLOCK_MIRROR_VOLUME)
(M) UNLMIRF (UNLOCK_MIRLOG_FILE)
(M) UNLOCK_MIRLOG_FILE
(M) UNLOCK_MIRROR_VOLUME
(M) UNPAIR_MIRROR_VOLUME
(M) UNPMIR (UNPAIR_MIRROR_VOLUME)

(M) VOLRPT (EXEC_CLX_VOLRPT)

The following commands are also accessible to all operators:

(A) AI (ALTER_INPUT)
(A) ALTER_INPUT

(A) BYE

(A) CALL
(A) COM (COMM)
(A) COMM

(A) DELETE_GLOBAL
(A) DISPLAY_PROFILE
(A) DLGB (DELETE_GLOBAL)
(A) DP (DISPLAY_PROFILE)

(A) EJECT

(A) GB (GLOBAL)
(A) GLOBAL

(A) LET
(A) LIST_GLOBAL
(A) LOG
(A) LSGB (LIST_GLOBAL)

(A) MDOM INT (MODIFY_OPERATING_MODE INT)
(A) MDP (MODIFY_PROFILE)
(A) MODIFY_PROFILE

This last set of commands is not detailed in the present manual. Refer to the *IOF Terminal User's Reference Manual Part III* for a complete description.



7.3 Using the GCOS 7 Help Mechanism

The help mechanism is one of the most frequently used forms of documentation on DPS 7000 systems. It is described in more detail in the multi-volume *IOF Terminal User's Reference Manual*.

Types of Console

Local Console: The help texts are available as described below.

Other Consoles: Synchronous Telecom terminals can sometimes be used as MAIN OPERATOR consoles. This gives access to formatted help screens. These are completely described in the *IOF Terminal User's Reference Manual*.

Using GCOS 7 Helps

The GCOS 7 helps are called by entering a question mark (?). Using the DERLOG command as an example, it would look like this:

```
S: ?DERLOG
```

Putting the question mark in front of the command gives a text explaining the command. In this case, there is a four-page text about the DERLOG command. But the most useful feature of GCOS 7 helps are the menu prompts, obtained by putting the question mark after the command:

```
S: DERLOG?  
DISPLAY_ERROR_LOGGING  
    OPTION STATE, LAST, SET, PRINT (PRINT):
```



You are then prompted to key in each one of the command's parameters. The first parameter of DERLOG command is OPTION; its default value is PRINT.

Key in:

?	For a help text about the current parameter (OPTION in the example).
<ENTER> or <TRANSMIT >	To select the default value (PRINT) and to go on to the next parameter.
/	To cancel the command and go back to S:
()	To cancel the default value without giving a new one in its place. (Does not work if the parameter is mandatory.)
;	To end prompting and execute the action, skipping the remaining parameters
<	To restart prompting from first value again. Useful if you have already selected some values and have changed your mind.
1 or 0	Yes or No. Only works if prompt finishes with a question mark, such as: ABN abnormal termination?
Value	The value to give to the parameter



7.4 Detailed Description of GCL Commands

In the following pages each GCL command, in alphabetical order, is fully described under the following headings:

- Purpose,
- Syntax,
- Parameters,
- Rules,
- Examples.

7.4.1 Command Prefixing

Operators can prefix commands with a station-name as follows:

ALL/command	for information on jobs at all stations.
MAIN/command	for information on jobs at the main station.
STNx/command	for information on jobs at the named station: STNx.

Refer to the *IOF Terminal User's Reference Manual, Part 1*.

7.4.2 Three Levels of Command Syntax

GCOS 7 commands have two standard forms:

1. The full command (verb_object): **ATTACH_CATALOG**
2. The abbreviation: **ATT**
3. Sometimes there is a third possible command syntax. This is most often a previous syntax that has been left intact for the convenience of users who have the habit of using it.

7.4.3 Hidden Parameters or Commands

Certain commands and parameters, usually those that are rarely used, are not included in the menu and/or prompt mechanism. The parameters or commands in this category are listed in this section below a dashed line: -----

You have these parameters listed in the menu by entering a plus (+) after its name, rather than a question mark (?). There are no help texts for these parameters.



7.5 "A" To "C" Commands

7.5.1 ATTACH_CATALOG (ATT) (ATTACH)

Purpose

Indicates to the system that in the following commands any reference to a cataloged file must be looked for in the specified catalog.

Syntax

```
{ ATTACH_CATALOG }  
{ ATT             }  
{ ATTACH         }
```

```
[CAT = file78]
```

Parameters

CAT The identification of the catalog in which references to cataloged files must be looked for. When omitted, the effect of any previous ATTACH_CATALOG command is negated and SITE.CATALOG is assumed again.

Rules

- The name of the catalog must end with .CATALOG. When not so, this suffix is automatically assumed.
- The command ATTACH_CATALOG, entered without parameters, cancels the effect of any previous ATTACH_CATALOG CAT=<catalog-name>.
- But it has no effect in the MAIN OPERATOR environment on catalogs automatically attached by a MWINLIB command. (This is assuming that the catalog in question was created with the AUTOATT attribute, and therefore that the assignment of one of the cataloged files implies the attachment of the catalog). In the MAIN OPERATOR environment, the only way to detach an auto-attachable catalog is to log off.



7.5.2 CANCEL_DUMP (CDP)

Purpose

Cancels a binary dump. The main operator and an IOF user working under project SYSADMIN can cancel any dump. Other users can cancel dumps pertaining to jobs for which they are submitter or user. For further information, refer to Appendix B *Index of Commands by "Object"*.

Syntax

```
{ CANCEL_DUMP }  
{  
{ CDP }
```

```
DUMP_ID = { Xnnnn[:dsn[:dpindex ]] | ALL | * }
```

Parameters

DUMP_ID identifies the dump or dumps to be cancelled. Xnnnn specifies the RON (Run Occurrence Number) of the job; dsn is the dynamic step number (default is all dumps for the RON); and dpindex is the number of the dump within the step (default is all dumps within the step). If ALL or "*" is specified, all dumps of SYS.SPDUMP are cancelled.

Constraints

None.

Examples

CDP X5432	{cancel all dumps of job X5432}
CDP ALL	{cancel all dumps of SYS.SPDUMP}
CDP X654:3:5	{cancel the dump whose RON is X654, dynamic step number is 3, and dpindex is 5}



7.5.3 CANCEL_JOB (CJ)

Purpose

Cancels the execution of one or more jobs. A job may be cancelled by its submitter, its USER and by main or station operators. Only a main operator can cancel a job that is not attached to his station. A station operator may cancel only jobs that are linked to his station. A main operator's IOF can be cancelled only by a main operator in IOF using the CJ command with the RONS parameter.

Syntax

```
{CANCEL_JOB}
{CJ          }
{TJ          }

[ RONS = (ron [ron] ...) ]

[ {SELECT}   {jclass} [ {usernm} [ {projnm} ] ] ]
[ {          } = {          } [ / {          } [ / {          } ] ] ]
[ {SEL      }   { *          } [ { *          } [ { *          } ] ] ]

[ {          } {1} ]
[ STRONG = {  } ]
[ {          } {0} ]

[ {          } {1} ]
[ FORCE = {    } ]
[ {          } {0} ]

[ {          } {1} ]
[ ENDSTEP = {  } ]
[ {          } {0} ]

[ {          } {1} ]
[ DUMP = {    } ]
[ {          } {0} ]

[ {MSG}      ]
[ {          } = char255 ]
[ {MSG }     ]

[ {SYSTEM}   ]
[ {          } = name4 ]
[ {SYS      } ]

[JOBSTATE = ALL|SCH|EX|SUSP|IN|HOLD]
[JOBNAME = name8]
```



Parameters

RONS	a list of up to 16 Run Occurrence Numbers of jobs to be cancelled.
SELECT	restricts the scope of the command to the jobs that belong to the specified combination of class/username/project. SELECT=* cancels all jobs that are not executing or suspended (this is to avoid that the jobs should be cancelled in incoherent states). Special care should be taken when using this option.
STRONG	if 1, jobs are cancelled and terminate abnormally regardless of whether the jobs are repeatable or not, and regardless of their internal logic (neither JUMP nor any further STEP orders are taken into account). Default is 0.
FORCE	if 1, jobs terminate abnormally, even if executing their exit routines. Default is 0.
ENDSTEP	if 1, cancellation is deferred until completion of the current step. Jobs then abort regardless of their internal logic (neither JUMP nor any further STEP orders are taken into account). Default is 0.
DUMP	if 1, a dump is generated, even if not requested in the control language. if 0, no dump is generated, regardless of the control language. if not specified, a dump is generated or not, according to the STEP option. For IOF jobs, no dump is generated.
MSSG	the text of a message to be included in the Job Occurrence Reports of the cancelled job(s).
SYSTEM	indicates the system where the command will be executed.
JOBSTATE	jobs to be cancelled by current state; jstate must be one of the following: ALL, SCH, EX, SUSP, IN, HOLD.
JOBNAME	jobs to be cancelled by name (name-8).



Rules

When RONS is used, neither SELECT, JOBNAME nor JOBSTATE can be used. Either RONS, SELECT, JOBNAME or JOBSTATE must be specified.

When STRONG=1 FORCE must not be 1.

When STRONG=1 ENDSTEP must not be 1.

When FORCE=1 ENDSTEP must not be 1.

An IOF MAIN operator may cancel the IOF of another MAIN operator if he gives the IOF RON (Run Occurrence Number). If the SELECT keyword is used, the MAIN operator's IOF jobs will not be cancelled.

NOTE:

The effect of a CJ command depends on its issuer and on the job it is aimed at:

if a job is in critical section when the CJ command is issued, its cancellation is deferred until the job exits critical section.

a job cannot cancel itself unless the ENDSTEP parameter is used.

only main and station operators may cancel a job that they did not submit.

only a main operator may cancel a job that is not attached to the same station as he is. A station operator may only cancel jobs linked to his station.

most service jobs cannot be cancelled by a CJ command. Some of them can be terminated by specific operator commands (for example `TERMINATE_SYSTEM_TRACE` for TRCCL job, `TERMINATE_OUTPUT_WRITER` for WRITER jobs; refer to the *System Operator's Guide* or to specific server documentation), others cannot be terminated during a session and are stopped only by system shutdown (e.g. LAEH, VCAM).

main operator's IOF can be cancelled only by another main operator in IOF using the CJ command with the RONS parameter.



Examples

CJ X123 {cancel one job}

CJ (X123 X42 X57) {cancel a series of jobs}

CJ X123 DUMP {cancel and force a dump}

CJ X123 DUMP=0 {cancel and suppress dump}

CJ X123 FORCE=1 {cancel, even if in exit routine}

CJ X123 ENDSTEP MSG='I DID IT BECAUSE OF CONFLICT WITH X324 '
{cancel at end of step and inform with a message}

CJ SEL=AA {cancel all jobs in class AA}

CJ SEL=*/JONES {cancel all jobs of user JONES}

CJ SEL=M JOBNAME=TEST1 JOBSTATE=EX
{cancel jobs named TEST1 which are in execution in
job class M}



7.5.4 CANCEL_OUTPUT (CO)

Purpose

Cancels the request that an output or set of outputs be printed. The outputs cannot be recovered after they have been cancelled.



IMPORTANT:

This command applies only to the outputs on which the submitter has access rights

Syntax

```
{ CANCEL_OUTPUT }
{ CO }

[ { (output14 [ output14 ] ...) } ]
[ OUTPUT = { output14-output14 } ]
[ { * | ALL } ]

[ SELCLASS = a [-a] ]

[ STRONG = { bool | 0 } ]

[ { STATION } { * } ]
[ { } = { ALL } ]
[ { ST } { name8 } ]

[ { SELECT } { user/project } ]
[ { } = { */project } ]
[ { } { */* } ]
[ { SEL } { * } ]

[ { JOBNAME } ]
[ { } = name8 ]
[ { JNM } ]

[ { SYSTEM } ]
[ { } = name4 ]
[ { SYS } ]
```




SELECT	restricts the scope of the command to outputs belonging to a specified user and/or project. Not meaningful when OUTPUT specifies a list of outputs (format 1). By default, only the outputs belonging to the command submitter are selected.
JOBNAME	restricts the scope of the command to the outputs of jobs with the name specified. If JOBNAME is specified, the default value for OUTPUT is *.
SYSTEM	indicates the system on which the command is to be executed.

Rules

- SELCLASS is ignored for names of specific outputs (Xnnnn: nnnnn or Xnnnn: name).
- If several outputs of the same job are given the same name, CO Xnnnn: name only affects the first output with that name encountered by the system.
- CANCEL_OUTPUT does not apply to outputs in the KNOWN state.

Examples

CO X123	{cancel all outputs of job X123}
CO (X123 X124)	{cancel all outputs of jobs X123 and X124}
CO X123 H	{cancel all outputs of job X123 that are of class H}
CO X123 A-S	{cancel all outputs of job X123 that are of classes A to S}
CO X123:2	{cancel specified outputs of job X123}
CO X123: SPR	
CO (X123:1 X123:RES)	
CO X123 STRONG	{cancel all outputs of job X123, even if being printed}
CO X123-X2167	{cancel all outputs, belonging to the command submitter, of the jobs whose RON is in the range X123 to X2167}
CO * Z	{cancel all class Z outputs, belonging to the command submitter, of all jobs}



- CO * SELECT=U1/P1 {cancel all the outputs belonging to user U1 of project P1}

- CO * STATION=BC10 {cancel all outputs, belonging to the command submitter, that are attached to the station BC10}

- CO X123-X2167 A ST=BC10 SEL=U1/P1
 {cancel all class A outputs of jobs in the range X123 to X2167 that are attached to station BC10 and belong to user U1 of project P1}

- CO JOBNAME=JOB1 {cancel all outputs, belonging to the command submitter, of the jobs named JOB1}



7.5.5 CANCEL_REQUEST (CR)

Purpose

Cancels a system request on a device. This is used to indicate that:

- a request volume cannot or will not be mounted,
- a device will not be switched to WRITE PERMIT,
- there is no recovery to be made following an incident,
- the Output Writer recovery mechanism is requested after an incident on a printer.

The operator decision is given to the job originating the request. Usually, the job aborts. For further details, refer to the sections concerning to the device indicated in the request message.

Syntax

```
{ CANCEL_REQUEST }
{ CR }

      { DEVICE }
      { DV } }
```

Parameters

DEVICE Identification of the device whose request is to be cancelled.

Rules

Refer to *Mirror Disks User's Guide* for CR related to a MIRROR disk.

Constraint

A mounting or recovery operation performed after CANCEL_REQUEST has been issued does not stop the effect of the CANCEL_REQUEST command.

Examples

```
CR MS03                      {cancel request on disk unit MS03}
CR PR04                      {cancel request on printer PR04}
CR MT03                      {cancel request on tape unit MT03}
```



7.5.6 CANCEL_TERMINAL_OUTPUT (CTO)

Purpose

Cancels the request that an output or set of outputs be printed by the Generalized Terminal Writer. The outputs cannot be recovered after they have been cancelled. Refer to the *Generalized Terminal Writer User Guide* for a complete description of the Generalized Terminal Writer.

Syntax

```
{ CANCEL_TERMINAL_OUTPUT }  
{ CTO  
  
    OUTPUT = (Rnnnn [ Rnnnn ] ...)  
  
    [ STRONG = { bool | 0 } ]
```

Parameters

OUTPUT	a list of the names of up to 16 outputs to be cancelled. Each output name has the form Rnnnn and is provided as the result of the DPRINT command (with parameter TWDEST).
STRONG	if 1, cancel even if outputs are being printed.

Rules

None.

Examples

CTO R1234	{cancel output with name R1234}
CTO (R1234 R1235)	{cancel two outputs}
CTO R1234 STRONG	{cancel even if being printed}



7.5.7 CANCEL_USER_REQ (CUR)

Purpose

Cancels file transfer or jobset requests.

Syntax

```
{ CANCEL_USER_REQ }
{ CUR
    [ REQUEST = ( name8 [ name8 ] ... ) ]
    [ ALL = { bool | 0 } ]
    [ ALL_USERS = { bool | 0 } ]
    [ RELEASE = { bool | 0 } ]
}
```

Parameters

REQUEST	<p>a list of up to 16 names of requests to be cancelled. Names start with a letter. Each name may take one of the following three forms:</p> <ul style="list-style-type: none"> – Jnnnnnnn a Jobset Occurrence Number (JON) – Fnnnnnnn a File transfer Occurrence Number (FON) – name8 the symbolic name of a request as specified by the NAME parameter of the ENTER_JOB_REQ or ENTER_FILETRANS_REQ commands. <p>JONs and FONs are provided as results of the ENTER_JOB_REQ or ENTER_FILETRANS_REQ commands: they are unique identifiers for the corresponding requests.</p>
RELEASE	<p>if 1, requests which are waiting for the completion of the request(s) being cancelled are released, as if the cancelled requests had completed execution. If omitted or 0, waiting requests are cancelled as well. Default value is 0.</p>
ALL	<p>This parameter enables a user to cancel all of his or her own requests.</p>



ALL_USERS This parameter can be used by the Main Operator to cancel all the requests of all users, or by a Station Operator to cancel all requests submitted on behalf of the Station.

Rules

RELEASE is ignored for jobset requests.

REQUEST, ALL and ALL_USERS are mutually exclusive.

Examples

CUR J18 (Cancel jobset request J18.)

CUR (F32 J27 RQP) (Cancel three requests.)

CUR F44 RELEASE (Cancel file transfer request F44 and release waiting requests.)



7.5.8 CONNECT_DIMENSION (CNDIM)

Purpose

Connects Job Classes or Load Modules to a dimension. See the *ARM User's Guide* for full details.

Syntax

```
{ CONNECT_DIMENSION }
{ CNDIM                }

DIM=name8

{ { JOBCLASS } }
{ {           } = (name2 [-name2] , name2 [-name2] . . . ) }
{ { JC       } }
{ {           } }
{ {           {name2} {name2} } }
{ { LM= (name31/{ } [, name31/{ } ] . . . ) } }
{ {           { * } { * } } }

[ { SYSTEM } ]
[ {       } = name4 ]
[ { SYS   } ]
```

Parameters

DIM	name of the dimension
JC	list of up to eight job classes or job class groups assigned to the specified dimension. The hyphen convention means that a list of JCs is to be connected to the specified dimension; for instance, (AY-BC) means that the job classes named AY, AZ, B, BA, BB, BC are to be connected.
LM	list of up to eight couples (load module name/job class name). The named LMs are considered as being connected to the present dimension if they are issued from the associated job class. The star convention means that the load module may be issued from any job class in the system.
SYSTEM	This parameter indicates the system where the command will be executed.



Rules

- Available in FULL ARM only.
- JOBCLASS (JC) and LM are mutually exclusive. Either JOBCLASS or LM (but not both) must be used.
- A JC or an LM which has already been connected to a dimension cannot be connected to another dimension.
- If the first object connected to the dimension is a JC or a set of JCs, further attempts to connect LMs are rejected except for the dimension SYS which may have both JCs and LMs connected.
- If the first object connected to the dimension is a LM or a set of LMs, further attempts to connect JCs are rejected except for the dimension SYS which may have both JCs and LMs connected.
- The use of CNDIM does not cancel earlier connections of job class groups or of load modules to the same dimension.
- Some LM connections are done automatically on predefined dimensions when the system is initialized. They cannot be disconnected using DISDIM command.

Example

```
CNDIM DIM=IOF JC= (A, B, C, AA, BM)
```

(Connect the Job Class Groups A,B,C,AA and BM to the IOF dimension.)



7.5.9 CONNECT_FUNCTION (CNFUNC)

Purpose

Connect a privileged function to a given resource domain. When a function is connected to a domain, this function can use the CPU resource of the domain. See Section 5-10 *Domain and CDP Overview*.

Syntax

```
{ CONNECT_FUNCTION      }
{ CNFUNC                }

        DOMAIN    = { name-8 }
        FUNCTION  = { name-8 }

[ { SYSTEM        }           ]
[ {                } = name-4 ]
[ { SYS          }           ]
```

Parameters

DOMAIN	name of the domain. On ARTEMIS1 machines, only the domains CDP1, CDP2 and CDP3 can be specified. On ARTEMIS2 machines, the domains can be only CDP1 and CDP2, depending on the configuration.
FUNCTION	<p>name of the function. This function name is not the name of the product. The function names are: SYSTEM, ORACLE, OPEN7, IQS, DW7, Y2KTEST, OPEN7_F1 and TDS_WEB.</p> <p>On ARTEMIS2, the OPEN7 function is assigned to the domain OPENP and its affectation cannot be changed. On ARTEMIS2, configuration B, the OPEN7_F1 function is assigned to the domain OPENC DP and its affectation cannot be changed.</p> <p>On ARTEMIS2, configuration A, the OPEN7_F1 function is not visible. No function can be connected explicitly to the domains OPENP or OPENC DP, and the functions OPEN7 and OPEN7_F1 cannot be connected to any domain. They are always connected by the system on their respective domains.</p>



SYSTEM the system where the command will be executed.

Rules

- To connect a function to the domain CDP2 or CDP3, this domain must be free or in other words, no other function must be connected to it.
- No function can be connected to the domain STANDARD (corresponds to IPU processors).
- By default all functions are connected to CDP1.
- A function can be connected to one and only one domain.

Examples

```
CNFUNC DOMAIN=CDP2 FUNCTION=OPEN7
```

When OPEN7 jobs start, they will use the CDP2 CPU resources

```
CNFUNC DOMAIN=CDP3 FUNCTION=IQS
```

When IQS jobs start, they will use the CDP3 CPU resources.

NOTE:

The CONNECT_FUNCTION, DISCONNECT_FUNCTION and DISPLAY_FUNCTION commands are part of the MAIN GCL domain.



7.5.10 CONNECT_LOAD (CNLD)

Purpose

Attaches classes to a class group.

Syntax

```
{ CONNECT_LOAD  }
{ CNLD          }

CLASS = aa

CLASSLIST = (aa [-aa] [aa [-aa]] ... )

[ { SYSTEM }           ]
[ {           } = name4 ]
[ { SYS      }           ]
```

Parameters

CLASS	name of the existing or to be defined class group.
CLASSLIST	the list of interval classes to be joined to the class group CLASS. Up to 8 intervals can be specified.
SYSTEM	This parameter indicates the system where the command will be executed.



Rules

- The command is rejected if a class in CLASSLIST is already a class group.
- A class cannot be connected to several class groups.
- The value of CLASS cannot be included in CLASSLIST.



IMPORTANT:

Some service job classes cannot be connected to a job class group.

NOTE:

Any class connected to a job class group assigned to a project by the System Administrator is dynamically assigned to the project.

Example

CNLD AA B-BZ {attaches the classes B,BA,...BZ to class group AA}



7.5.11 CREATE_COMPLEX_GENERATION (CRCXGEN)

Purpose

This command is used to describe and generate a COMPLEX. It is normally for use of the COMPLEX Administrator.

Syntax

```
{ CREATE_COMPLEX_GENERATION }
{ CRCXGEN                      }

SOURCE = subfile-name31

SLLIB = lib78

[ ENABLE = { bool | 0 } ] ;
```

Parameters

SOURCE	Name of the source subfile which contains the input directives of the COMPLEX Description. This subfile must be catalogued.
SLLIB	Name of the source library which contains the SOURCE subfile.
ENABLE	When ENABLE = 1, the COMPLEX Description is syntactically and semantically analyzed, and the COMPLEX is generated if these analyses have succeeded. When ENABLE = 0, only the COMPLEX Description is syntactically and semantically analyzed. The default is 0.



Constraints

- CREATE_COMPLEX_GENERATION command must be syntactically correct.
- If a COMPLEX is already started then CREATE_COMPLEX_GENERATION (CRCXGEN) command must not be entered with the option ENABLE equal to 1.

Example

```
CRCXGEN SOURCE = ABCD SLLIB = P1.MYLIB
```

(Generates a COMPLEX using subfile named ABCD
in library named P1.MYLIB.)



7.5.12 CREATE_DIMENSION (CRDIM)

Purpose

Creation of a non-predefined dimension. A new dimension is created with the current pattern of the named existing dimension. In case the LIKE keyword is not used, the dimension is created with the current pattern of the BATCH dimension. See *ARM User's Guide* for full details.

Syntax

```
{ CREATE_DIMENSION  }
{ CRDIM              }

DIM = name8

[LIKE = { name8 | BATCH } ]

[ { SYSTEM }          ]
[ {           } = name4 ]
[ { SYS      }          ]
```

Parameters

DIM	Name of the new dimension.
LIKE	Name of an existing dimension.
SYSTEM	System where the command will be executed.

Rules

- the ARM Level must be FULL.
- the maximum number of Dimensions is 32.
- the dimension to be created may not be a predefined dimension.
- the existing dimension named in the command must not be SYS.

When a Dimension is created, its pattern is identical to the current pattern of the LIKE parameter dimension. If LIKE is not used, it is the current pattern of the BATCH dimension.

Example:

```
CRDIM POOL4 LIKE=IOF {Create a new dimension, named POOL4, with IOF
                    current parameter values}
```



7.5.13 CREATE_FILTER (CRFLT)

Purpose

Creates an elementary filter within a filter set. An elementary filter defines a set of filtering criteria. A generic message must meet all criteria defined by a filter to be acceptable to that filter. The filters belonging to a given filter set are arranged according to rules defined by the FLT_TYPE parameter.

Syntax

```
{ CREATE_FILTER }
{ CRFLT          }

      { FILTERSET }
      {           } = fltst
      { FLTST     }

      { FILTER    }
      {           } = flt
      { FLT       }

      { FLT_TYPE  } { I }
      {           } = { E }
      { FTYPE     } { M }

[ KEY = ( name8 [ name8 ] ) ]

[           { * } ]
[ TYPE = { R } ]
[           { M } ]

[ JOBID = name8 ]

[ RON   = Xnnnn   ]

[ { SYSTEM }           ]
[ {           } = name4 ]
[ { SYS   }           ]
```

**Parameters**

FILTERSET	the name of the filter set to which the filter being defined belongs.
FILTER	the name of the filter being defined.
FLT_TYPE	the type of the filter, as follows: I inclusive E exclusive M mandatory. A message is acceptable to a filter set if: – it meets at least one of its inclusive filters (if any), – it meets none of the exclusive filters, – it meets all its mandatory filters. A message is acceptable to a filter if it meets all criteria defined by the parameters KEY, TYPE, JOBID and RON.
KEY	the key or inclusive range of keys of the acceptable messages. The format of each key is: aaaannnn, where aaaa is a string of 2 to 4 alphabetic characters and nnnn is the numerical part between 1 and 4059. Leading zeros are eliminated.
TYPE	the types of the acceptable messages, as follows: * all messages, R request messages requiring a reply or a manual or non-manual action, M messages requiring a manual action.
JOBID	identification of the Job issuing the message; the messages originate from the load modules. NOTE: The JOBID filter parameter is not a criterion for most system messages.
RON	the Run Occurrence Number (Xnnnn) of the Job issuing the acceptable messages.
SYSTEM	This parameter indicates the system where the command will be executed.



Rules

- One at least of KEY, TYPE, JOBID or RON must be specified.
- No more than 16 filters may belong to a single filter set.
- The filter set must be created (by use of the CREATE_FILTER_SET command) prior to creating the filters it is to contain.

Examples

```
CRFLT A1 A101 I TYPE=*
```

{create inclusive filter A101 within set A1 accepting all types of messages}

```
CRFLT A2 A101 E KEY=(DV01 DV99)
```

{create exclusive filter A101 within set A2 analyzing and rejecting all messages with keys in the range DV01 to DV99}

```
CRFLT A3 A101 M KEY=(AA01 ZZ99) TYPE=M JOBID=TEST RON=X29
```

{create mandatory filter A101 within set A3 accepting all action messages from job TEST with RON X29}



7.5.14 CREATE_FILTERSET (CRFLTST)

Purpose

Creates a set of filters. A set of filters consists of a number of elementary filters. The combination of several elementary filters belonging to the same filter set defines a class of messages that the operator wishes to receive. Filters defined by an operator apply only to himself. By default, when no filter is active the operator receives no generic message from the system. It is sufficient that a message matches one of the defined filter sets to be delivered to the issuer of the command.

Syntax

```
{ CREATE_FILTERSET }
{ CRFLTST          }

      { FILTERSET  }
      {            } = fltst
      { FLTST      }

      { PRIORITY   }
      {            } = dec1
      { PRTY       }

      [ { SYSTEM   }           ]
      [ {           } = name4 ]
      [ { SYS      }           ]
```

Parameters

FILTERSET	the name of the set of filters to be created.
PRIORITY	the priority of the filter set in the range 0 to 9. The lower the value, the higher the priority.
SYSTEM	This parameter indicates the system where the command will be executed.

Rules

No more than 16 filter sets may be defined for any single operator.

Example

```
CRFST A1 3           {create filter set A1 with priority 3}
```



7.6 "D" Commands

7.6.1 DEASSIGN_MIRLOG_FILE (DASGMIR)

Purpose

This command is used when resolving SITE.MIRLOG error conditions related to the file itself, the volume where it is located or the device containing this volume.

It is to be used with great caution, normally under the supervision of your Bull representative.

This command is available only if the system has been configured with MIRROR = YES.

Syntax

```
{ DEASSIGN_MIRLOG_FILE }  
{ DASGMIR }
```

Parameters

None.

Rules

Refer to *Mirror Disks User's Guide*.

Example

DASGMIR

Deassigns the SITE.MIRLOG file and displays the message:

```
MR19 DEASSIGNMENT OF THE SITE.MIRLOG  
SUCCESSFUL
```




7.6.2 DELETE_DIMENSION (DLDIM)

Purpose

Deletes a dimension. See the *ARM User's Guide* for full details.

Syntax

```
{ DELETE_DIMENSION }
{ DLDIM              }

      DIM = name8

      [ { SYSTEM    }           ]
      [ {           } = name4 ]
      [ { SYS      }           ]
```

Parameters

DIM	name of the dimension to be deleted.
SYSTEM	This parameter indicates the system where the command will be executed.

Rules

- the predefined (or default) dimensions (SYS, TDS, IOF, and BATCH) may not be deleted.
- the ARM level must be FULL.
- the command is rejected (message AR85) if there is at least one step allocation in the dimension to be deleted.

Example:

```
DLDIM DIM=IOF3           {delete the dimension named IOF3}
```



7.6.3 DELETE_FILTER (DLFLT)

Purpose

Deletes a single filter, a set of filters, all filters, or all filter sets that belong to the operator.

Syntax

```
{ DELETE_FILTER }
{ DLFLT          }

      { FILTERSET } = { *          }
      { FLTST     } = { name2     }

      [ { FILTER }           ]
      [ {           } = name4 ]
      [ { FLT      }         ]

      [ { SYSTEM }           ]
      [ {           } = name4 ]
      [ { SYS    }           ]
```

Parameters

FILTERSET	When used alone (without FILTER), specifies the set of filters to be deleted (* means all operator defined filter sets). When used in conjunction with FILTER, specifies the filter set to which the filter to be deleted belongs.
FILTER	The name of the filter to be deleted. When omitted, all filters that belong to the set of filters named by FILTERSET are deleted.
SYSTEM	This parameter indicates the system where the command will be executed.



Rules

When FILTERSET is *, FILTER cannot be specified.

Examples

DLFT A1 A101	{delete filter A101 within filter set A1}
DLFLT A1	{delete all filters of set A1}
DLFLT *	{delete all filters and sets of filters that belong to the operator}



7.6.4 DISCONNECT_DIMENSION (DISDIM)

Purpose

Disconnects Job Class Groups or Load Module(s) from a dimension. See the *ARM User's Guide* for full details.

Syntax

```

{ DISCONNECT_DIMENSION }
{ DISDIM }

DIM=name8

[ {JOBCLASS } ]
[ { } = (aa [-aa] , [aa [-aa]] ..) ]
[ {JC } ]

[ {name2} {name2} ]
[ LM=(name31/{ } [,name31/{ } ] ...) ]
[ { * } { * } ]

[ { SYSTEM } ]
[ { } = name4 ]
[ { SYS } ]

```

Parameters

DIM the name of the dimension.

JC list of up to eight Job Class Groups to be disconnected from the specified dimension. If JC=* is specified, all the Job Classes are disconnected from the Dimension. The hyphen convention means that a list of JCs is to be disconnected from the dimension; for instance, (AY-BC) means that the job classes names AY, AZ, B, BA, BB, BC are to be disconnected.



LM	list of up to eight couples (load module name/job class name). The named LMs are considered as being disconnected from the present dimension if they are issued from the associated job class. The star convention means that the specified load module is no longer known to the load module/job class correspondence array of the dimensions. If LM=* is specified, all the Load Modules are no longer known to the Dimension.
SYSTEM	This parameter indicates the system where the command will be executed.

Rules

- Available in FULL level only.
- JC and LM are mutually exclusive. One of the two must be specified.
- LM connections done automatically, on predefined dimensions, during system initialization, cannot be disconnected using DISDIM. See the CONNECT_DIMENSION command for more information.
- For both JC and LM, only one value can be specified when * or ALL are used.

Example

```
DISDIM DIM=TDS2 JC=(TA, TB)
      (Disconnect job classes TA and TB from the dimension
      TDS2.)
```



7.6.5 DISCONNECT_FUNCTION (DISFUNC)

Purpose

Disconnect a privileged function from a given resource domain. *See Section 5.10 Domain and CDP Overview.*

Syntax

```
{ DISCONNECT_FUNCTION      }
{ DISFUNC                  }

        DOMAIN = { name-8 }
        FUNCTION = { name-8 }

[ { SYSTEM      }           ]
[ {              } = name-4 ]
[ { SYS        }           ]
```

Parameters

DOMAIN	name of the domain. On ARTEMIS1 machines, only the domains CDP1, CDP2 and CDP3 can be specified. On ARTEMIS2, the domains can only be CDP1 and CDP2, depending on the configuration.
FUNCTION	name of the function. This function name is not the name of the product. The function names are: SYSTEM, ORACLE, OPEN7, IQS, DW7, Y2KTEST, OPEN7_F1 and TDS_WEB. On ARTEMIS2, the functions OPEN7 (connected to the domain OPENP) and OPEN7_F1 (connected to the domain OPENC DP) cannot be disconnected.
SYSTEM	the system where the command will be executed.

Rules

- To disconnect a function from a domain, the function must have been connected to it and must not be running. In other words, it must have no running Load Module for this function.
- No function can be disconnected from the domain STANDARD (corresponds to IPU processors).



- **When a function is disconnected from CDP2 or CDP3, it is automatically connected to the domain CDP1 (default domain).**
- When a function is disconnected from CDP1, it is automatically connected to the domain STANDARD.

Examples

DISFUNC DOMAIN=CDP2 FUNCTION=OPEN7

The function OPEN7 is connected to the domain CDP2, the command is successfully performed only if the OPEN7 jobs are stopped. In other cases an error message is sent.

If the command is completed, the OPEN7 function returns to domain CDP1.

DISFUNC DOMAIN=CDP1 FUNCTION=IQS

The function IQS is connected to the domain CDP1. The command is successfully performed only if the IQS jobs are stopped. After execution of this command, the function IQS cannot use the resources of the CDP1 domain and is connected to the domain STANDARD.

NOTE:

The CONNECT_FUNCTION, DISCONNECT_FUNCTION and DISPLAY_FUNCTION commands are part of the MAIN GCL domain.



7.6.6 DISCONNECT_LOAD (DISLD)

Purpose

Detaches class(es) from a class group.

Syntax

```
{ DISCONNECT_LOAD      }
{ DISLD                }

CLASS = aa

CLASSLIST = (aa[-aa] [aa[-aa]]...)

[ { SYSTEM }           ]
[ {           } = name4 ]
[ { SYS     }           ]
```

Parameters

CLASS	name of the class group to be modified.
CLASSLIST	the list of interval classes attached to the class CLASS that should be detached (* for all classes). Up to 8 intervals can be specified.
SYSTEM	indicates the system on which the command will be executed.

Rules

None.

Example

```
DISLD AA BC-BE {detach classes BC,B D, BE from Class Group AA}
```




7.6.7 DISPLAY_ARS (DARS)

Purpose

This directive displays the current migration jobs, the activity statistics, and/or the configuration parameters of the ASM 7 File Migrator (ARS).

Syntax

```
{ DISPLAY_ARS }
{
  { DARS
    {
      { CONFIGURATION }
      [ { } = bool ]
      { CONF
        {
          { STATISTICS }
          [ { } = bool ]
          { STAT
            {
              { RSTFILE }
              [ { } = bool ]
              { RSTF
                {

```

```
[ PRTFILE = file78 ]
```

Parameters

CONFIGURATION	If 1, displays the current values of the ARS configuration parameters. For Migration-in, these are the job execution class, time-out period, and operating modes. If 0, this information is not displayed.
STATISTICS	If 1, displays statistics about ARS activity. These are accumulated values starting from ARS activation to the current date. If 0, this information is not displayed.
RSTFILE	If 1, displays the ron, efn, and elapsed time for each Migration-in job currently in progress. If 0, this information is not displayed.
PRTFILE	The execution report file.



7.6.8 DISPLAY_ASSIGNED_FILES (DASGF)

Purpose

Displays on the screen,

if the RON parameter is used:

- the list of all the files used by the job (EFN, IFN, and PMD) and for each of them, if the DETAILED option is specified, MD, DVC, FILESTAT, and CATALOG.

if the EFN parameter is used:

- the list of all the RONs of the JOBS having assigned the file. If the DETAILED option is used, for each RON, the IFN and the PMD will be displayed.

Syntax

```
{DISPLAY_ASSIGNED_FILES}
{
  {DASGF
    {RON = {output5} | EFN = {char44}
      {DETAILED}
      {
        {DTLD
          } = {bool | 0 }
        }
      }
    }
  }
```

Parameters

RON	Run Occurrence Number of the JOB
EFN	The External File Name of the file
DETAILED	Displays more information about files

If the RON parameter is used:

- =0 Only EFN, IFN, and PMD are displayed
- =1 Assign file characteristics are displayed

if the EFN parameter is used:

- =0 Only the list of the RONs of the JOBS having assigned the file is displayed
- =1 For each RON, the IFN and the PMD is displayed



Constraints

Parameters RON and EFN are mutually exclusive.

If the parameter RON is used:

1. Command right execution:
 - a user that belongs to the SYSADMIN project or to a project having the MAIN attribute has all rights; he has the visibility of all JOBS running on the system
 - a user that belongs to other projects has only the visibility of the JOBS submitted by himself.
2. The command is only allowed on the JOBS in executing state.

If the parameter EFN is used: the command execution is reserved to a user belonging to the SYSADMIN project or having the MAIN attribute.

Limits:

- If the parameter IFN is used with the parameter DETAILED, a maximum of 20 IFNs per JOB will be displayed.

The line:

```
IFN= ..... PMD= ..
```

indicates that more than 20 IFNs are assigned to the EFN given in the parameter.

- For cataloged tape files on more than 10 media, only the 10 current media are displayed.

Remarks concerning EFN parameter usage:

The EFN parameter specifies the EXTERNAL_FILE_NAME of the file. (No residency information must appear in this EFN.)

For a temporary file, the EFN must be followed by \$TEMPRY.



Example

To display the RONs of the job having assigned the temporary file
";102213.TEMP.SLLIB":

```
DASGF EFN=TEMP.SLLIB$TEMPRY
```

REMARK:

If several jobs have assigned a temporary file called TEMP.SLLIB, the
DASGF command will display information for each temporary file.

For EFNs with a generation number and/or a version number, only the EFN
syntaxes shown in the example are allowed (G suffix with + or - is rejected).

Examples

```
DASGF EFN=DIR1.FILE/G0001;
```

```
DASGF EFN=DIR.FILE/V01;
```

To use the command with a non-standard EFN (EFN with lowercase characters
with blank characters, etc.) the EFN parameter must be quoted. In these cases, to
specify the EFN in the input field use:

- single quote with formatted screen; e.g., for a temporary file ';000635.myefn
- triple quotes with a serial terminal; e.g., for a temporary file: "';000635.myefn"

In some cases (CATALOG files assigned by the system and "passed" files) no IFN
exists associated with the EFN. In these cases, only the RON for which the file is
assigned will be displayed.

Examples

Use of the RON parameter

```
DASGF X366 DETAILED=0
13.07 EFN                                IFN      PMD
      GEF.FILE003                          IFN003   NA
      GEF.ULFA0001                          IFN004   UP
```

Displays the list of the files assigned by the RON X366.

```
DASGF X366 DETAILED=1
13.07 EFN= GEF.FILE003
      IFN= IFN003                                PMD= NA
      MD= (BFS162, BFS163, BFS164)
      DVC= MS/B10                                FILESTAT= UNCAT
```



```

-----
EFN= GEF.ULFA0001
IFN= IFN004                                PMD= UP
MD= (BFR0AA)
DVC= MS/B10                                FILESTAT= CAT
CATALOG= GEF.CATALOG:BFU047:MS/B10
-----

```

Displays the list of files assigned by the RON X366, and the detailed information for each of them.

Use of the EFN parameter

DASGF EFN=DIR1.FILE1 DETAILED=0

```

13.14 EFN=DIR1.FILE1
MD= (BFS162/BFS163/BFS164)
DVC= MS/B10                                FILESTAT= UNCAT
RON= X1234 X7568

```

Displays the list of RONS having assigned the file DIR1.FILE1

DASGF EFN=DIR1.FILE1 DETAILED=1

```

13.16 EFN= DIR1.FILE1
MD= (BFS162/BFS163/BFS164)
DVC= MS/B10                                FILESTAT= UNCAT
RON= X1234
                                     IFN= IFN001      PMD= UP
                                     IFN= IFN002      PMD= IN
RON= X7568
                                     IFN= INFILE       PMD= IN
                                     IFN= OUTFILE      PMD= OU

```

Displays the list of RONS having assigned the file DIR1.FILE1 and the detailed information for each of them.

**NOTES:**

1. The CATALOG field is optional; it is displayed only when the RON parameter is used and if the file is catalogued in a non-auto-attachable catalog.
2. If the EFN is SYS.OUT or SYS.IN, the MD and DVC are set to RSDT (resident) and the PMD is set to NA (non available).
3. If the EFN is DUMMY, the MD, DVC and PMD are set to NA.
4. For the other EFN, the PMD is set to NA if the file is not opened.
5. PMD is set to ?? if an error is detected during access to the file management table.
6. If the file has been assigned with the residency parameter equal to RESIDENT and if it has never been opened, the DVC is unknown and only "MS" is displayed.
7. If the JOB works with remote files, the PMD of these files is set to RM (remote).

PMD is the processing mode of the file, that is its state when the DASGF command is executed. PMD can be equal to IN (file opened in input), OU (file opened in output), UP (file being updated), NA (not available) or RM (remote).

Error messages:

```
SH46 cmdnm UNSUCCESSFUL: UNKNOWN JOB ron (01)
SH46 cmdnm UNSUCCESSFUL: STATE OF ron MEANINGLESS WITH THE COMMAND (02)
SH46 cmdnm UNSUCCESSFUL FOR ron gr4 (03)
SH46 cmdnm NOT ALLOWED FOR ron: SUBMITTER NOT AUTHORIZED (04)
SH46 cmdnm NOT ALLOWED FOR ron: NO FILE ASSIGNED (05)
SH46 cmdnm NOT ALLOWED FOR efn: FILE NOT ASSIGNED (06)
SH46 cmdnm UNSUCCESSFUL FOR efn gr4 (07)
```



7.6.9 DISPLAY_CLC (DCLC)

Purpose

This command, Display Cartridge Library Client, displays the status of devices connected to the cartridge library.

For the selected units, the command displays the volume currently mounted or the status of the mount or dismount operation in progress, and also a list of the pending mount requests.

Syntax

```
{ DISPLAY_CLC }  
{  
{ DCLC }  
[ { DEVICE } ]  
[ { DV } ]  
} = device ]
```

Parameters

DEVICE: When this parameter is present, the status of the selected device is displayed. When this parameter is not specified, the status of all devices connected to the cartridge library is displayed.

Rules

The command is rejected if the communications link between the Cartridge Library Client (CLC) and the Cartridge Library Server (CLS) has not been established. It is also rejected during the CLC initialization phase.

DEVICE must be the name of a cartridge library device.

Example

DCLC (Displays the status of all devices connected to the cartridge library.)

DCLC DEVICE=CT22 (Displays the status of device CT22.)

DCLC DV=CT22

DCLC CT22



7.6.10 DISPLAY_CLX (DCLX)

Purpose

Displays the status of a device or group of devices integrated within the cartridge tape library:

Syntax

```
{ DISPLAY_CLX }  
{ DCLX      }  
  
[DEVICE = <device> ]
```

Parameters

DEVICE the name of the device.

If DEVICE is not specified, the status of all devices in the cartridge library are displayed.

Rules

- The status displayed represents the device status as seen by the CLX.
- When mounting or dismounting operations are in progress, the information displayed by DC and DCLX commands may not be identical: some actions can be considered completed by Device Manager whereas they are in progress to CLX.



Example 1

The DCLX command is performed on a system with four drives and two volumes mounted.

CT15 and CT16 are 18-track devices (CT/LIB/M5). The pool identifier 3 is associated with CT/LIB/M5 devices (18-track formatted WORK volumes).

CT17 and CT18 are 36-track devices (CT/LIB/36T). The pool identifier 6 is associated to CT/LIB/36T (36-track formatted WORK volumes).

Volume 001863 is dismounted, then remounted. The second DCLX command is processed while the dismounting is in progress and the last mount request is pending.

```
S: DCLX
  14.12 TU15 V2.1 CT15  M5    POOLID =    3 (0,0,1,0) STATUS : 001863 MOUNTED FOR X1981
  TU15 V2.1 CT16  M5    POOLID =    3 (0,0,1,1) STATUS : STANDBY
  TU15 V2.1 CT17  36T   POOLID =    6 (0,0,1,2) STATUS : 001864 MOUNTED FOR X1980
  TU15 V2.1 CT18  36T   POOLID =    6 (0,0,1,3) STATUS : STANDBY
TU14 CLX_DISPLAY COMMAND COMPLETED
* DW10 CT15 DISMOUNT 001863 S
* 14.13 CT15 MOUNT 001863 LIB FOR X1981
S: DCLX
  14.13 TU15 CT15  M5    POOLID =    3 (0,0,1,0) STATUS : DISMOUNT 001863 FOR X1981
  IN PROGRESS
  TU15 V2.1 CT16  M5    POOLID =    3 (0,0,1,1) STATUS : STANDBY
  TU15 V2.1 CT17  36T   POOLID =    6 (0,0,1,2) STATUS : 001864 MOUNTED FOR X1980
  TU15 V2.1 CT18  36T   POOLID =    6 (0,0,1,3) STATUS : STANDBY
  TU15 V2.1 CT15  M5    POOLID =    3 (0,0,1,0) STATUS : MOUNT 001863 FOR X1981 PENDING
  TU14 V2.1 CLX_DISPLAY COMMAND COMPLETED
```



Example 2

Successive DCLX commands are performed while a volume is being mounted, and then while the same volume is being dismounted.

```
S: DCLX CT17
 15.35 V2.1 TU15 CT17 36T POOLID = 6 (0,0,1,2) STATUS : STANDBY
TU14 V2.1 CLX_DISPLAY COMMAND COMPLETED
S: EJ FILSAVE LIB=LCIU.CLX.JCLLIB VL=001863
 15.36 X1887 IN FILSAVE USER=OPERATOR CLASS=P SPR=6 STATION=BCA9
 15.36 X1887 STARTED FILSAVE OPERATOR P
* 15.36 CT17 MOUNT 001863 LIB FOR X1887
S: DCLX CT17
 15.36 TU15 V2.1 CT17 36T POOLID = 6 (0,0,1,2) STATUS : MOUNT 001863 FOR X1887
  IN PROGRESS
TU14 V2.1 CLX_DISPLAY COMMAND COMPLETED
  TU11 V2.1 CT17 IN LIBRARY : 001863 MOUNTED
  JB08 X1887.1 STEP H_UTILITY XPR=8 PGID=22
S: DCLX CT17
 15.38 TU15 V2.1 CT17 36T POOLID = 6 (0,0,1,2) STATUS : 001863 MOUNTED FOR X1887
TU14 V2.1 CLX_DISPLAY COMMAND COMPLETED
* DW10 CT17 DISMOUNT 001863 S
 15.39 X1887.1 COMPLETED FILSAVE OPERATOR P
S: DCLX CT17
 15.39 TU15 V2.1 CT17 36T POOLID = 6 (0,0,1,2) STATUS : DISMOUNT 001863 FOR X1887
  IN PROGRESS
TU14 V2.1 CLX_DISPLAY COMMAND COMPLETED
  TU12 V2.1 CT17 IN LIBRARY : 001863 DISMOUNTED
S: DCLX CT17
 15.39 TU15 V2.1 CT17 36T POOLID = 6 (0,0,1,2) STATUS : STANDBY (ACTION TO UNLOCK
  RESOURCE IN PROGRESS)
TU14 CLX_DISPLAY COMMAND COMPLETED
S: DCLX CT17
 15.40 TU15 V2.1 CT17 36T POOLID = 6 (0,0,1,2) STATUS : STANDBY
TU14 V2.1 CLX_DISPLAY COMMAND COMPLETED
```



7.6.11 DISPLAY_COMPLEX (DCX)

Purpose

Displays information about the whole COMPLEX. This command is a part of the High Availability product (HA). For more information, see the *High Availability Administrator's Guide*.

Syntax

```
{ DISPLAY_COMPLEX }  
{ DCX }
```

Parameters

None.

Rules

The CMSR of the MEMBER on which this command is entered must be running.

Example

DCX Displays all information about the COMPLEX.



7.6.12 DISPLAY_CONFIGURATION (DC)

Purpose

Displays the status of selected elements of the hardware or software configuration.

Syntax

```
{DISPLAY_CONFIGURATION}
{DC

    [ { * } ]
    [ { DEVICE } = {device-id } ]
    [ { DEV } } {device-class } ]
    [ [ DV } } {device-type } ]
    [ {device-id-char2 } ]

    [{OUTCLASS} { * } ]
    [{ } = { } ]
    [{OC } {a[-a]} ]

    [ { * } ]
    [OUTDV = {PR[mn]} ]
    [ {T[nmn]} ]
    [ {W[nmn]} ]

    [{STATION} { * } ]
    [{ } = {ALL} ]
    [{ } = {CURRENT} ]
    [{ST } {name8} ]

    [ { * } ]
    [ PLM = { } ]
    [ { [lib-nm..] lm-nm [:volnm :dvc] =nn PAGES YY/MM/DD} ]

    [BS = name6]

    [{VOLUME} = volume 6 : device-class ]
    [{VOL } ]

    - - - - -
    [LIB = lib78 ]
```



Parameters

DEVICE display the status of the devices. This may be specified as:

*	all devices
device-id	a specified device (e.g., MS03)
device-type	devices of a type (e.g., MS)
device-id-char2	devices of the same type within an interval (e.g. CT01-05)
device-class	all devices that have the specified characteristics (e.g., MS/B10, CT/S75, CT/LIB/M5/C, etc.)

You can specify all characteristics or only some of them:

CT/36T: all class 36T cartridge tape devices not in a library

CT/LIB: all cartridge tape devices in the library

CT/C: all devices with the compaction capability, whether in a library or not.

OUTCLASS display the configuration of the output classes (priority, output media) and output activities for these classes. These may be specified as:

*	all classes
class:	a specified class
class-class:	a range of classes

where class is a letter in the range A to Z. A list of up to 10 classes or ranges of classes may be specified.

OUTDV display the output currently being processed or waiting for the specified device (PRnn, Tnnn, or Wnnn), device class (PR, T or W), or all writers (*, i.e. PR + T + W).



STATION	specifies one of the following: an explicit station name, CURRENT (for currently attached stations) or ALL or * (for all stations). When used in conjunction with OUTCLASS or OUTDV, the information shown is restricted to the station concerned (values ALL and * are then ignored). When neither OUTCLASS nor OUTDV is specified, information about the configuration of the named stations is displayed.
PLM	display information (name, location and number of units) about all (*) or a designated pre-initialized load module (PLM). A single PLM is designated as follows: [lib-name..] Im-name [:volnm :dvc] where: <ul style="list-style-type: none">– lib-name is the library name (default SYS.HLMLIB)– Im-name is the name of the load module– volnm is the volume name (uncataloged library)– dvc is the device class of volnm (uncataloged library)– PAGES is the size of LM– YY/MM/DD is the date of the last PLM of this LM
BS	display the status of the required elements of backing store. This may be specified as: SYS backing store occupation for system, ALL backing store occupation for system and all jobs, MAP which backing stores are used and on which volumes, SM loaded sharable modules, CKPT active checkpoints, VMF active Virtual Memory Files, IO average number of I/Os per second, Xnnnn space used by the named job, volume-name jobs that use the volume as backing store.
VOLUME	display the status of a volume. The volume must be standard. For cartridge, the full device class is needed if several models are present on the site.
LIB	display loaded shareable modules in backing store for the specified library. Used only with BS = BM.



7.6.13 DISPLAY_CONSOLE_LOG (DCLOG)

Purpose

Scans and displays selected portions of the SYS.LOGC file.

Syntax

```
{DISPLAY_CONSOLE_LOG}
{DCLOG                }

[          {BEG                }]
[          {END                }]
[          {ISL                }]
[          {<string15          }]
[          {>string15          }]
[OPTION = {<                }]
[          {>                }]
[          {[mm.dd. [yy]yy/] hh.mm}]
[          {G>string15        }]
[          {G<string15        }]
[          {X                  }]
[          {IDM                }]
```

[SIZE = {dec2|18}]

[PATH = dec4]

[SKIP = dec4]

Parameters

OPTION How the file is to be scanned. Applicable values are:

- BEG: display the first page of the file.
- END: display the last page of the file (also reset the values of PATH, SIZE and of the "current string").
- ISL: position to previous ISL and display its page.
- <string15: move backwards till the given string is found. It may be used to refer to the "current string" as defined in a previous DCLOG command and display corresponding page (also memorize value of string15 as "current string").



- >string15: move forward till the given string is found. It may be used to refer to the "current string" as defined in a previous DCLOG command and display corresponding page (also memorize value of string15 as "current string").
- <: move backwards and display the corresponding page.
- >: move forwards and display the corresponding page.
- [mm.dd.[yy]yy/]. hh.mm: locate the entry that corresponds to the given date or as close as possible to that date and display the corresponding page.
- G>string15: move forward and display all lines containing string15.
- G<string15: move backward and display all lines containing string15.
- X: display the date of the current line being displayed, the value of PATH and the "current string".
- IDM: display the current page again. When omitted, > is assumed.

SIZE	Display page size. Default is 18. Possible values are 3 to 18.
PATH	Scanning is limited to current day or to value specified by PATH. When a search has failed, the next search is not limited by PATH or by the date.
SKIP	Number of SYS.LOGC entries skipped before starting to scan.

Rules

The DCLOG command is only available in Interactive Mode.

When the year is expressed with the digits in the OPTION expression, it must verify the interval:

1961 =< yyyy =< 2060



Examples

DCLOG ISL {move to the preceding ISL}

DCLOG '>JOB START' {locate the string 'JOB START'}

DCLOG {move forwards and display the corresponding page}

DCLOG 'G>X123 AB' {display occurrences of string 'X123 AB' from this point}

DCLOG <ABORTED {look backwards for string ABORTED}

DCLOG G>X123 {display all lines that contain X123 from this point}

DCLOG 12.31.2041/18.08 {locate the entry that corresponds to the December 31, 2041, from 6:08p.m.; and display the corresponding page}



7.6.14 DISPLAY_DEVIATION_TIME (DDTIME)

Purpose

Displays the time deviation of GCOS 7 and the time deviation of the Service Processor. Time deviation is the difference between GMT and the local time. Its value is positive when local time leads (is later or greater than) GMT, and negative when local time lags GMT. If Service Processor support is not available, only the GCOS 7 time deviation is displayed. In the case of an Input/Output failure, the time deviation is reported as UNKNOWN.

Syntax

```
{ DISPLAY_DEVIATION_TIME }  
{ DDTIME }
```

Parameter

None.

Rules

None.

Example

DDTIME (Display the GCOS 7 and the SP time deviations)



7.6.15 DISPLAY_DEVICE_TRACE (DDVTR)

Purpose

Displays information on events logged for a device, a range of devices or devices of a certain class. The displayed information shows for each selected device:

- the current number of I/Os,
- the number of logged events,
- the threshold value (see the MODIFY_DEVICE_TRACE command).

Syntax

```
{ DISPLAY_DEVICE_TRACE }
{ DDVTR }

[ { * } ]
[ { DEVICE } { dvtyp } ]
[ { } = { } ]
[ { DV } { dvnm } ]
[ { } { dvc } ]

[ VOLNAME = name6 ]
```

Parameters

DEVICE	the device(s) for which information is to be displayed, expressed as: *: all devices, dvnm: a specified device, dvtyp: all devices of a type dvc: all devices of a class. When omitted, all devices (*) is assumed.
VOLNAME	information displayed is restricted to the named volume.

Rules

VOLNAME may be used only when DEVICE is dvnm.



Examples

DDVTR *	{display information for all devices}
DDVTR MS	{display information for all disk units}
DDVTR MS/B10	{display information for all MS/B10 disk units}
DDVTR MS03	{display information for disk unit MS03}
DDVTR MS03 DISK5	{display information for volume DISK5 on disk unit MS03}

NOTE:

For more information on this command, see Section 6.6, *Monitoring Device Incidents*, and the EL02 message in the *Console Messages Directory*.



7.6.16 DISPLAY_DIMENSION (DDIM) - BASIC ARM

Purpose

Displays statistical information about dimension(s) on systems with ARM BASIC level. See the *ARM User's Guide* for full details.

Syntax

```
{ DISPLAY_DIMENSION }
{ DDIM }

DIM = [ { name8 } ]
      [ { } ]
      [ { * } ]

[ { MPL } ]
[ OPTION = { } ]
[ { Xnnnn } ]

[ { SYSTEM } ]
[ { } = name4 ]
[ { SYS } ]
```

Parameters

DIM name of the dimension. If DIM=* or ALL, the command applies to all dimensions.

OPTION the type of information required:
OPTION= MPL displays the following:

```
DIMNAME MPL ALLOC USED (MB) FIX (MB) OUT/S FLT/S
```

- the MPL, as set by the MDDIM command
- the current number of steps in execution
- the used size (USED) in megabytes; does not include dedicated I/O cache size
- the fixed size (FIX) in megabytes; does not include dedicated I/O cache size
- the page out frequency (OUT/S)
- and the page fault frequency (FLT/S).

OPTION= Xnnnn: see the Xnnnn option of the Full ARM command (below)



SYSTEM This parameter indicates the system where the command will be executed.

Rules

User must be in a MAIN operator project.

Examples

DDIM * {Display information for each of the dimensions}

DDIM IOF {Display information on the IOF dimension}



7.6.17 DISPLAY_DIMENSION (DDIM) - FULL ARM

Purpose

Displays statistical information about dimension(s) and job activities on systems with ARM FULL. For full information, see the *ARM User's Guide*.

Syntax

```
{ DISPLAY_DIMENSION }
{ DDDIM                }

[ DIM = { name-8 } ]
[       { *       } ]

[       { ATTR    } ]
[       { CN      } ]
[       { MPL     } ]
[       { PERF   } ]
[ OPTION = { RSC   } ]
[         { CPU   } ]
[         { RON   } ]
[         { Xnnn  } ]
[         { *     } ]

[ { SYSTEM } ]
[ {       } = name-4 ]
[ { SYS   } ]
```



Parameters

DIM name of the dimension. If * or ALL is specified, the command applies to all dimensions.

OPTION the type of information required:

* If * is specified, the command applies to all options.

ATTR For more information about the ATTR, CN, MPL, PERF and RSC parameter values, see the *ARM User's Guide*.

CPU displays the following information for STANDARD, CDP1, CDP2 and CDP3 job categories:

JOBTYPE	ASR (%)	CPU (%)	PWM (%)
STANDARD	22.5	250.2	400.0
CDP1	7.4	167.1	600.0
CDP2	43.1	327.1	800.0
CDP3	27.4	15.1	500.0

if there is no EPU, FPU or GPU processor, CPU displays the following for STANDARD job categories:

JOBTYPE	ASR (%)	CPU (%)	PWM (%)
STANDARD	22.5	250.2	400.0

- STANDARD jobs are jobs which run only on standard processors (IPUs).
- CDP1 entitled jobs are jobs which run on EPU processors (Extended Processor Units), and also on standard processors (IPUs) if needed.
- CDP2 entitled jobs are jobs which run on FPU processors (Extended Processor Units type 2), and also on standard processors (IPUs) if needed..
- CDP3 entitled jobs are jobs which run on GPU processors (Extended Processor Units type 3), and also on standard processors (IPUs) if needed.
- The Absolute Service Rate (ASR) is the percentage of resource consumption of all jobs in the category concerned relative to the maximum capacity of the system.



- The CPU consumption of all jobs in the category concerned, relative to the maximum CPU power of the system (including both IPU's and EPU's, FPU's or GPU's). This value is not divided by the number of CPU's and therefore can exceed 100. The value is averaged over a relatively long period (presently 50 seconds).
- The maximum CPU power (PWM), including both IPU's and EPU's, FPU's or GPU's that the jobs in the category can use. This value is not divided by the number of CPU's.

This option can be used to ascertain the system load for each job category, for example, to check whether more standard jobs can be started.

RON	Run Occurrence Number
Xnnnn
SYSTEM	The system where the command is executed.

Rules

- Options ATTR, CN, PERF, RON, RSC, and CPU are available in FULL level only.
- When Option = Xnnnn, CPU and DIM must be *.
- Instantaneous information about Rons is displayed as it is collected; the total amount of ASR and CPU for all the dimensions can be more than 100. ASR, CPU and I/O concern the last regulation period; XL is the current one.

Examples

1. This command displays information about the execution requests for each of the dimensions.
DDIM DIM=* OPTION=MPL
2. This command displays ATTR, CN, MPL, PERF, RSC, CPU, and RON for all the existing dimensions and jobs.
DDIM DIM=* OPTION=*



- 3. This command displays information about system CPU consumption for standard and other domain if the corresponding CPU resource is present.

DDIM DIM=* OPTION=CPU

If there are no EPUs, FPUs or GPUs processors, the information displayed is similar to the following (the STANDARD line corresponds to the IPUs processors):

JOBTYPE	ASR (%)	CPU (%)	PWM (%)
STANDARD	11.2	50.0	100.0

If there are EPUs, FPUs or GPUs processor, the information displayed is similar to the following example (for 4 IPUs + 2 EPUs + 4 FPUs + 1 GPU processors):

JOBTYPE	ASR (%)	CPU (%)	PWM (%)	
STANDARD	22.5	250.2	400.0	(IPUs)
CDP1	7.4	167.1	600.0	(IPUs + EPUs)
CDP2	43.1	327.1	800.0	(IPUs + FPUs)
CDP3	27.4	15.1	500.0	(IPUs + GPUs)

The whole CPU power is 400 (IPUs) + 200 (EPUs) + 400 (FPUs) + 100 (GPUs) = 1100.

The table below, that refers to the example above, lists the CPU power used and the fraction of the whole CPU power that can be used by each type of job.

Job Type	CPU Power Used	Fraction of Whole CPU Power That Can Be Used
Standard jobs	250/400	400/1100
CDP1 entitled jobs	167.1/600	600/1100
CDP2 entitled jobs	327.1/800	800/1100
CDP3 entitled jobs	15.1/500	500/1100



7.6.18 DISPLAY_DJP (DDJP)

Purpose

Displays information on the Distributed Job Processing (DJP) activity of the system.

Syntax

```
{ DISPLAY_DJP }
{ DDJP }

[ OPTION = { ALL }
           { QUEUE }
           { HOLD }
           { EX }
           { WTFL }
           { WDTM }
           { WFR }
           { WAIT }
           { TFR }
           { Tddd } ]

[ SITE = { ALL }
         { sitenm } ]

[ YEAR = dec4 ]

[ JOBNAME = bool ]

[ DETAILED = bool ]

[ UFTWAIT = bool ]
```



Parameter

OPTION	<p>The information to be displayed, as follows:</p> <p>ALL all known requests with their characteristics, QUEUE requests that are in a queue, HOLD requests being held, EX requests being processed, WTFL requests waiting for file availability, WDTM requests waiting for date and time condition, WTFR requests waiting for finish of another transfer request, WAIT requests that are waiting and eligible to be processed, TFR the current status of the system processors in charge of transfer activities (files, jobsets and outputs), Tnnn status of processor in charge of transfer whose identification is Tnnn.</p> <p>The default value is ALL, except when SITE is specified, in which case, TFR is taken as the default value.</p>
SITE	<p>Restricts the display to the requests that are addressed to the named site (QUEUE, HOLD, EX, WTFL, WDTM, WTFR, WAIT) or to the processors which are dedicated to this site (TFR).</p>
YEAR	<p>Displays all the days of the week not worked, plus all the holidays for the specified year. There are default lists for non-working days that can be modified using the MODIFY_DJP command.</p>
JOBNAME	<p>When set to 1, this parameter gives either the subfile name or the file name. It is useful to locate specific requests among all requests, when there are a large number of requests. It must be used with the OPTION = ALL.</p>
DETAILED	<p>When this parameter is set to 1, the command displays the public holidays in detailed mode for the year specified in the YEAR parameter.</p>
UFTWAIT	<p>When this parameter is set to 1, the command displays the waiting time, in hours and/or minutes, between two data transfers.</p>



Rules

SITE is ignored when OPTION is Tnnn.

YEAR and SITE are incompatible.

YEAR and OPTION are incompatible.

UFTWAIT and (OPTION, SITE, YEAR, JOBNAME) are mutually exclusive.

Examples

```
DDJP                                (Display all DJP requests.)
DDJP QUEUE HST1                    (Requests in the queue for site HST1.)
DDJP WDTM                          (Requests waiting for date and time condition.)
DDJP TFR                            (Status of all transfer processors.)
DDJP T3                             (Status of transfer processor T3.)
DDJP TFR HST2                      (Status of transfer processors dedicated to site HST2.)
DDJP SITE=HST2                    (same as above)
DDJP ALL JOBNAME = 1
                                   (Displays the subfile or the file name for all the DJP
                                   job requests.)
```

Command (1) displays the information (2):

```
DDJP UFTWAIT                        (1)
UFT WAITING TIME : 20 minutes      (2)
```

Command (1) displays the information (2):

```
DDJP YEAR=2000 DETAILED            (1)

NOT WORKING DAYS FOR:BC06 ARE: SATURDAY,SUNDAY }
FOR 2000,HOLIDAYS ARE:                }
      JULIAN DAY NUMBER                JULIAN DAY } (2)
      1                                JAN 1      }
      3                                JAN 3      }
      36                               FEB 5      }
      252                              SEP 8      }
```



7.6.19 DISPLAY_DUMP (DDP)

Purpose

Displays information concerning the specified dump, or the percentage of free space and the number of dumps in SYS.SPDUMP. For further information, refer to Appendix B *Index of Commands by "Object"*.

Syntax

```
{ DISPLAY_DUMP }  
{  
{ DDP }  
}  
  
[ DUMP_ID = char14 ]  
  
[ SPACE = bool ]
```

Parameters

DUMP_ID the dump identification as displayed by the LSDP command.

The following message is displayed if the dump exists:

```
DP02 <dumpid> PGID=<jjjj> <lname> <date>  
TERM_MSG=xxxxxxxx
```

where "jjjj" is the process-group identifier; "lname" is the name of the load module (first 16 characters only); "date" is the date and time of dump creation; and "xxxxxxxx" is the termination message of the step.

SPACE if 1, the percentage of free space and the number of dumps in SYS.SPDUMP are displayed in the following message format:

```
DP03 nn% OF SYS.SPDUMP SPACE FREE  
NBDUMP=dec4
```




Constraints

DUMP_ID and SPACE are mutually exclusive.

Examples

DDP X2345_1_1 {display information on the dump X2345_1_1}

DDP SPACE {display percentage of free space and number of
 dumps in SYS.SPDUMP}



7.6.20 DISPLAY_DYNSAVE_SVR (DDYNSVR)

Purpose

This command displays information on the current state of the dynamic saves in progress.

Syntax

```
{ DISPLAY_DYNSAVE_SVR }  
{ DDYNSVR }
```

Parameters

None.

Rules

This function is only available when the server is active.

If the server is inactive, the following message is displayed: "PLEASE RETRY LATER (TERMINATE RUNNING OR NO SERVER)".

Example

DDYNSVR (Produces the display shown below:)

```
BYA2: ddynsvr  
  PLUG  SERVER OF DYNAMIC SAVE      :          X2762  
BYA2: ddynsvr  
  PLUG  SERVER OF DYNAMIC SAVE      :          X1616  
  PLUG  SAVE XRON                    :          X1815  
  PLUG  NUMBER OF FILES TO SAVE     :              3  
  PLUG  REMAINING FILES TO SAVE     :              3  
  PLUG  WORKFILE BLOCK-4K USED     :          10  
  SAVE IN SAVE PHASE
```

BYA2:



7.6.21 DISPLAY_ERROR_LOG (DERLOG)

Purpose

Displays information held in the SYS.ERLOG system file. The information to be displayed can be selected according to the type of events that are being analyzed.

Syntax

```
{DISPLAY_ERROR_LOG}
{DERLOG          }

[                {STATE} ]
[                {LAST  } ]
[OPTION = {      } ]
[                {SET   } ]
[                {PRINT} ]

[                {CHEXC  } ]
[                {SYSDOWN} ]
[                {SYSUP  } ]
[                {ITR    } ]
[SELECT = {      } ]
[                {SYSERR } ]
[                {SYSEXC } ]
[                {MISC   } ]
[                {OPR    } ]

[ {DEVICE}      ]
[ {          } = dvm ]
[ {DV          } ]

[ATN = bool]
[WARN = bool]
[ABN = bool]
[KEY = char8]

[ {DISPLAY_LENGTH}      ]
[ {          } = dec1 ]
[ {DLG          }      ]

[FROM = mm.dd. [yy] yy/hh.mm]
```



Parameters

OPTION	<p>the action to be performed, as follows:</p> <p>STATE display the state of the SYS.ERLOG file. LAST display the last message again. SET specifies default values for DISPLAY_LENGTH and FROM to apply to future operations. SET with no other options resets the search values (DISPLAY_LENGTH = 1 and FROM = current date). PRINT print messages, as selected by the other parameters.</p> <p>When omitted, PRINT is assumed.</p>
SELECT	<p>the type of messages searched for, as follows:</p> <p>CHEXC channel exceptions (device controller in stop) SYSDOWN system terminations and crashes SYSUP system loadings ITR Initialization Test Results SYSERR system errors SYSEXC system exceptions MISC miscellaneous, not any of the other cases OPR operator message written by SNDRLOG</p>
DEVICE	<p>select messages relating to I/O events on the named device.</p>
ATN/WARN/ABN	<p>further select the type of events on DEVICE (see command START_DEVICE_TRACE for an explanation)</p>
KEY	<p>(with SELECT=OPR) selects the key (first characters) of the operator messages to be printed.</p>
DISPLAY_LENGTH	<p>the number of selected records to be displayed.</p>
FROM	<p>the date and time from which to start search in reverse chronological order.</p>



Rules

- SELECT, DEVICE, KEY, DISPLAY_LENGTH and FROM are ignored when not relevant for the value of OPTION.
- When OPTION is PRINT, either SELECT or DEVICE must be used.
- ATN, WARN and ABN are ignored if DEVICE is not specified.
- KEY is significant with SELECT = OPR
- When the year is expressed with the digits in the FROM expression, it must verify the interval: 1961 =< yyyy =< 2060.

Examples

```
DERLOG SET DLG=5 FROM=10.25.88/15.30
DERLOG SET DLG=5 FROM=12.31.2041/20.10
DERLOG PRINT CHEXC
{5 records of type CHEXC are displayed (if any) starting from the indicated date}

DERLOG OPR KEY = XXXX
{operator messages written by SNDERLOG and beginning with XXXX are
displayed}
```

NOTE:

For more details on the information returned by this command, see the EL01, EL09, and EL10 messages in the *Console Messages Directory*.



7.6.22 DISPLAY_FUNCTION (DFUNC)

Purpose

Display information about functions and domains. This information for a domain is the function name (if any), and the number of running Load Modules that are using the domain resources.

Syntax

```
{ DISPLAY_FUNCTION      }
{ DFUNC                }

      [ DOMAIN = { name-8} ]
      [          = { *    } ]

      [ { SYSTEM      }      ]
      [ {              } = name-4 ]
      [ { SYS         }      ]
```

Parameters

DOMAIN	Name of the domain. If the default value "*" is specified, the command is applied to all domains.
SYSTEM	The system where the command will be executed.

Rules

The number of displayed domains is related to the hardware resource presence.



Examples

DFUNC * {display information about all domains}

DOMAIN	RUN JOBS	FUNCTIONS
STANDARD	84	SYSTEM
STANDARD	2	DW7
CDP1	12	ORACLE
CDP1	0	OPEN7-F1
CDP2	3	OPEN7
CDP3	23	IQS

On ARTEMIS2, configuration A:

DFUNC * {display information about all domains}

DOMAIN	RUN JOBS	FUNCTIONS
STANDARD	25	SYSTEM
STANDARD	0	DW7
CDP1	5	ORACLE
CDP1	0	IQS
CDP2	2	TDS_WEB
OPENP	4	OPEN7

On ARTEMIS2, configuration B:

DFUNC * {display information about all domains}

DOMAIN	RUN JOBS	FUNCTIONS
STANDARD	22	SYSTEM
STANDARD	1	IQS
CDP1	0	ORACLE
CDP1	0	DW7
CDP1	1	Y2KTEST
CDP1	2	TDS_WEB
OPENCDP	3	OPEN7_F1
OPENP	4	OPEN7

NOTE:

The CONNECT_FUNCTION, DISCONNECT_FUNCTION and DISPLAY_FUNCTION commands are part of the MAIN GCL domain.



7.6.23 DISPLAY_GCL_CACHE (DGC)

Purpose

This command generates a display giving the following information for cache levels:

- Number of SYS.SPOOLi files used,
- Names of pre-initialized domains from the SYS.HBINLIB file for GCL cache,
- Cache validation for GCL commands,
- Cache usage for GCL commands,
- Cache size used for GCL commands,
- Number of GCL commands found in cache.

Syntax

```
-----  
{ DISPLAY_GCL_CACHE }  
{ DGC }
```

Parameters

None.

Example

DGC

-> VALID SPOOLS: 0, 1

PREINITIALIZED DOMAINS:

H_BREAK
H_NOCTX
IOF
LIBMAINT_SL
MAIN

CACHE FOR COMMANDS:

VALIDATED AT ISL
USED FOR ALL COMMANDS
MAXIMUM SIZE IS 128K (2 segments)
1295/1344 REFERENCED COMMANDS FOUND IN CACHE



7.6.24 DISPLAY_HARDWARE_CONFIGURATION (DHW)

Purpose

This command requests a display of the state of hardware units on the operator console.

Syntax

```
{ DISPLAY_HARDWARE_CONFIGURATION }
{ DHW
  [
    { *
    { hardware unit
  ]
  [ UNIT = { hardware unit type } ]
  [
    { device class
    { device path
  ]

  [
    { CSS
  ]
  [ SELECT = { CTL
    { DEVICE DV
  ]

  [ PATH = { bool
  ] ]
```

Parameters

- UNIT Specifies the resources whose states are to be displayed. The following values are possible:
- * represents units specified in the SELECT parameter.
 - A hardware unit (4 characters) which can be:
 - a CSS resource (IPnm, EPnm, FPnm, GPnm, IOnm, MUhh, PXnm)
 - a controller (xCnm)
 - a peripheral device (dvnm).
 - The hardware-unit-type parameter (2 characters) represents a resource family (MU, MS, IP, EP, FP, GP, etc.).
 - The device-class representing all devices that have the given characteristics (MS/B10, MT/T9, TC/M3, MC/M5, UC/M53, etc.).



You can specify all characteristics or only some of them:

CT/36T: all class 36T cartridge tape devices not in a library

CT/LIB: all cartridge tape devices in the library

CT/C: all devices with the compaction capability, whether in a library or not.

– The device-path (9 characters) representing a path to a peripheral device (xCNm.dvnm).

SELECT

This parameter is used with the UNIT=* parameter to specify units whose states are to be displayed. Three options are possible:

- The CSS keyword specifies all CSS units.
- The CTL keyword specifies all controllers.
- The DEVICE or DV keyword specifies all peripheral devices.

PATH

The boolean value parameter 1 can be used to display the status of all path to or from the unit specified in the DHW command. A path is limited to two consecutive resources in the configuration hierarchy.

When an **IOP name** is specified, a display line appears for each path from the IOP with the following information: path status, PC number, and external name of the associated controller (xCNm).

When a **controller** is specified, the first display line indicates the external name (xCNm), the channel number and the status of the controller. This is followed by one display line per device linked to the controller giving the external name of the associated path (xCNm.dvnm), the channel number (pclc), and path status.

When a **device** is specified, one display line appears for each path leading to the device; each display line indicates the external name of the path (xCNm.dvnm), the channel number (pclc), and the path status.



Rules

- When the SELECT=CTL option is specified, display lines appear only for disk controllers (MC) and unit record controllers (UC).
- When the SELECT=DV option is specified, a display line appears indicating the status of all the peripheral devices and their controllers in the configuration.
- The commands DHW MU, DHW MUxx, or DHW * CSS do not distinguish between the different memory units. Consequently, the dedicated I/O cache memory unit (if any) is not distinguishable from other memory units. If, however, a dedicated I/O cache is not initialized, it is displayed as "NOT ACCESSIBLE"(see Examples).

Examples

DHW * SELECT=CSS	{Display the state of all Central Subsystem Units.}
DHW IP	{Display the state of all Instruction Processor Units.}
DHW EP	{Display the state of all Extension Processor Units type 1.}
DHW FP	{Display the state of all Extension Processor Units type 2.}
DHW GP	{Display the state of all Extension Processor Units type 3.}
DHW MS/FSA	{Display the states of all the FSA disks.}
DHW MS01 PATH	{Display all the paths leading to MS01.}
DHW IO01 PATH	{Display all the paths that start from IOP 01.}
DHW IP05	{Display the state of the IPU 05}
DHW MU	{Displays the state of all memory units, as follows:
	08.38 MU03 IN
	08.38 MU02 IN
	08.38 MU01 OUT (NOT ACCESSIBLE)
	08.38 MU00 IN
	MU01 is a dedicated I/O cache memory unit which has not been initialized.}



Biprocessor in bi-system (on the S1 system)

S: DHW IP

IP01 OUT (NOT ACCESSIBLE)
IP51 IN

Display all paths starting from IOP 06

S: DHW IO06 PATH

IO06 IN
PX60 IN
 TC02 C200 IN
 CC11 C300 IN
PX61 IN
 MCC4 C400 IN
PX62 IN
 UC20 CB00 IN
 MCCA CA00 IN
 MCC8 C800 IN

Multiprocessor running a mono-system

S: DHW * SELECT = CSS

MU07 OUT (NOT ACCESSIBLE)
MU06 OUT (NOT ACCESSIBLE)
MU05 OUT (NOT ACCESSIBLE)
MU04 IN
MU03 OUT (NOT ACCESSIBLE)
MU02 OUT (NOT ACCESSIBLE)
MU01 OUT (NOT ACCESSIBLE)
MU00 IN
IO07 IN
PX70 IN
PX71 IN
PX72 IN
IO06 IN
PX60 IN
PX61 IN
IO05 IN
PX52 IN
PX51 IN
PX50 IN
IO04 IN
PX47 IN
PX46 IN
PX45 IN
PX44 IN



```
PX43 IN
PX42 IN
PX41 IN
PX40 IN
IO03 OUT (NOT INSTALLED)
PX30 OUT (NOT INSTALLED)
PX31 OUT (NOT INSTALLED)
PX32 OUT (NOT INSTALLED)
IO01 IN
PX10 IN
PX11 IN
PX12 IN
IO00 IN
PX00 IN
PX01 IN
PX02 IN
IP03 IN
IP02 IN
IP01 IN
IP00 IN
EP04 IN
EP05 IN
```

Display of all the paths leading to MS01

```
S: DHW MS01 PATH

MC01.MS01 0001 IN (READY)
MC02.MS01 0401 OUT (BY OPERATOR)
MC21.MS01 2001 IN (SPARE)
MC23.MS01 2601 IN (READY)
```

Display of all FPU CPU

```
S: DHW FP

FP04 OUT (BY OPERATOR)
FP05 IN
```



7.6.25 DISPLAY_IO_CACHE (DIOC)

Purpose

This command displays current information and statistics on the I/O Cache function. See the *Large Memory Cache User's Guide* for full details.

Syntax

```
{DISPLAY_IO_CACHE}  
{DIOC }
```

Parameters

None.

Rules

None.



Example

```

S: DIOC
  BEGINNING DATE: 91/03/11  14.57
  CURRENT DATE  : 91/03/11  15.9
  INITIAL SIZE  : 8 [MB]
  CURRENT SIZE  : 8 [MB]
  STATUS       : ACTIVE
  MODE         : CACHE
  DEFAULT      : FORCE
  GLOBAL STATISTICS SINCE 91/03/11  14.57:
    :   NB_IOS  : NB_READS : NB_WRITES : NB_HITS  : %HITS
    :-----:-----:-----:-----:-----:-----
    :    30717:    15596:    10773:         0:    35
  CURRENT ACTIVITY (LAST 5 PERIODS) :
  PERIOD DURATION: 120 [SEC]
  MIN %WI         : 0
  MAX %WI         : 0
    :   NB_IOS  : NB_READS : NB_WRITES : NB_HITS  : %HITS
    :-----:-----:-----:-----:-----:-----
  1:    1659:     875:     660:         0:    39
  2:    6171:    3294:    2304:         0:    37
  3:    6425:    3146:    2139:         0:    33
  4:    6954:    3270:    2031:         0:    29
  5:    5325:    2844:    2053:         0:    38
    :-----:-----:-----:-----:-----:-----
    :    26534:    13429:    9187:         0:    34

```

Key to the headings:

NB_IOS:	Total number of I/O requests diverted to the cache.
NB_READS	Number of read requests.
NB_HITS	Number of physical reads that have been gained.
NB_WRITES	Number of physical writes which have been gained.
%HITS	The hit ratio, calculated as follows: $((NB_HITS - \text{nb. of read-aheads})/NB_IOS) \times 100$
MIN_%WI	Minimum occupancy of the WRITE-INTO images in the Cache during the last five periods. % is relative to the whole cache size.
MAX_%WI	Maximum occupancy of the WRITE-INTO images in the Cache during the last five periods. % is relative to the whole cache size.



7.6.26 DISPLAY_IOF (DIOF)

Purpose

Displays certain IOF parameters and statistics.

Syntax

```
{ DISPLAY_IOF }
{ DIOF          }

[          { name12 } ]
[          { Xnnnn  } ]
[ USER = { ALL     } ]
[          { *      } ]

[          { name12 } ]
[ DETAILS = {       } ]
[          { Xnnnn  } ]

[ { RECN_TIME } ]
[ {           } = bool ]
[ { RT        } ]

[ { IDLE_TIME } ]
[ {           } = bool ]
[ { IDT       } ]
```

Parameters

USER	The identification of an IOF user for whom global information is to be displayed. This may be expressed as the user name, the Run Occurrence Number (Xnnnn) of a session, or ALL (or *) if all users are intended.
DETAILS	The identification of an IOF user for whom detailed information is to be displayed. This may be expressed as the user name or as the Run Occurrence Number (Xnnnn) of an IOF session.
RECN_TIME	If 1, the maximum reconnection time is displayed (i.e. the lapse of time after which reconnection of an accidentally disconnected user is not allowed).
IDLE_TIME	If 1, the maximum idle time is displayed (i.e. the lapse of time after which idle IOF users are disconnected).



Rules

None.

Examples

```
DIOF                                {display global information}
DIOF USER=X1234                    {display global information for session X1234}
DIOF *                              {display global information for all sessions}
DIOF DETAILS=JONES                 {display detailed information for user Jones}
DIOF RT                             {display maximum reconnection time}
DIOF IDT                            {display maximum idle time}
```

NOTE:

Information displayed is as follows:

```
S: DIOF
```

```
hh.mm DGI LOGON=n1 PUT=n2 SEND=n3 GET=n4 RECV=n5
      SCD=n6 SSW=n7
```

where:

- n1 = number of logon
- n2 = number of logical outputs on IOF terminals
- n3 = number of physical outputs on IOF terminals
- n4 = number of logical inputs on IOF terminals
- n5 = number of physical inputs on IOF terminals
- n6 = number of system commands
- n7 = number of steps switches

For DIOF USER=... (user global information):

```
      usernm [lm-name]
```

where:

- usernm = user name
- lm-name = name of the program currently being executed.

```
S: DIOF USER=X1265
```



7.6.27 DISPLAY_JAS (DJAS)

Purpose

This command displays the jobs that are currently connected to the JAS (Journalization Advanced Service). It gives the following information for the jobs of the designated JAS:

- Run Occurrence Number,
- Job Name,
- User Name (submitter),
- Job Class,
- Load Module being executed.

DJAS is particularly useful before executing a TAKEOVER_MEMBER command with the WEAK option. It displays the TDS, Batch and IOF applications to be stopped.

Syntax

```
{ DISPLAY_JAS }  
{ DJAS          }  
  
[ JAS_NAME = { GREEN | BLUE | SYS } ]
```

Parameters

JAS_NAME Journalization Advanced Service name. It may be GREEN, BLUE or SYS. The default is all.

Rules

None.

**Example**

DJAS JAS_NAME=BLUE Displays all jobs connected to BLUE JAS, as follows:

```
hh.mm   JAS:BLUE  X151.1  TP7JCLAC  OPERATOR  P  TDS1
                X168.2  BAT2     USER2    P  H_BAT2
```

where:

P is the Job Class

BAT2 is a batch job using BLUE JAS.

TDS1 is a HA-type TDS using BLUE JAS.

H_BAT2 is the Load Module.



7.6.28 DISPLAY_JOB (DJ)

Purpose

Displays the current status of selected jobs.

Syntax

```
{DISPLAY_JOB}
{DJ

    [          {jstate      } ]
    [          {jtype       } ]
    [JOBID = { (jstate jtype) } ]
    [          {Xnnnn       } ]
    [          {name8       } ]

    [{SELECT}   {jclass} [ {usernm} [ {projnm} ] ] ]
    [ {         } = {         } [ / {         } [ / {         } ] ] ]
    [{SEL      }   { *       } [ { *       } [ { *       } ] ] ]

    [{STATION}   { *       } ]
    [ {         } = {ALL    } ]
    [{ST        }   {name8   } ]

    [JOBNAME = name8]

    [          {200  } ]
    [LIMIT = {20  } ]           (See parameter description
    [          {dec4} ]           for default value)

    [{SYSTEM}   ]
    [ {         } = name4 ]
    [ {SYS      }   ]

    [{SEQUENCE}   {PRIORITY|DECR_ITIME|INCR_ITIME} ]
    [ {         } = {         |         |         } ]
    [ {SEQ      }   {         |DIT      |IIT      } ]

    [{DETAILED}   ]
    [ {         } = {bool|_0} ]
    [ {DTLD     }   ]
```



Parameters

JOBID	<p>the jobs to be displayed designated by their Run Occurrence Number (Xnnnn), name (name8), current state (jstate), job type (jtype), or current state and job type.</p> <p>jstate must be one of the following: ALL, SCH, EX, SUSP, OUT, IDLE, IN, HOLD</p> <p>jtype must be one of the following: ALL, BATCH, IOF, SERVICE, NSERVICE</p> <p>If NSERVICE is specified, all jobs except the service jobs are displayed.</p> <p>When JOBID is unspecified, ALL is assumed.</p>
SELECT	<p>restricts the directive to the jobs that belong to the specified combination of class/user-name/project. * is all jobs. When omitted, all jobs for which USER is the terminal user are assumed.</p>
STATION	<p>restricts the list of jobs displayed to those that are attached to the named station (* or ALL for any station). When omitted, the current working station of the submitter is assumed.</p>
JOBNAME	<p>the job or jobs to be displayed by name (name-8).</p>
LIMIT	<p>maximum number of jobs to be displayed. Beyond that limit, only the number of jobs per class is displayed. The default and maximum values are: When DETAILED=1, default 20, maximum 100 When DETAILED=0, default 200, maximum 850.</p>
SYSTEM	<p>indicates the system where the command is to be executed.</p>
SEQUENCE	<p>specifies the order in which jobs are to be displayed for each job state. The sort criterion can be the scheduling priority (keyword PRIORITY) or the introduction time (keywords DECR_ITIME and INCR_ITIME). The display order for each job state is determined by:</p> <ul style="list-style-type: none">– the scheduling priority (PRIORITY), from highest to lowest. The introduction time is used for second level sorting within each priority level. This is the default display criterion.



– or the introduction time alone, (PRIORITY not specified):

DECR_ETIME = from most recent to oldest

INCR_ETIME = from oldest to most recent.

DETAILED

(abbreviation DTLD) specifies whether detailed job information is to be displayed. The information returned depends on the job state:

HOLD

Hold count

Introduction time

EX/ SUSP

Introduction time

Switches

Full load module name or step state

System internal identifier associated with the step

CPU and elapsed times for the current step

Number of processes

Backing store

Used and fixed instantaneous memory sets.

It is not always possible to provide all of the above information for jobs in state EX or SUSP, in particular the ARM dimension (e.g. IOF, BATCH) and the step-specific information (load module name, step CPU and elapsed time, number of processes, backing store, memory sets). This is the case at step initialization (if the step is waiting for a resource statically assigned by a JCL statement or for resources in a dimension), and at step termination.

OUT

The introduction and termination times.

These times may be unknown if the job is in the OUT state, and if they were kept after SYS.OUT recovery at system RESTART.

IN, SCH, IDLE

The introduction time.

When DETAILED=1, LIMIT may not exceed 100 and the default value is 20.

Rules

When JOBID is in the form Xnnnn, SELECT and STATION and LIMIT are ignored.

JOBID and JOBNAME are mutually exclusive.



Examples

S : DJ	{Display the status of all jobs for which USER is the command submitter and that are attached to his current working station.}
S : DJ EX	{Display the status of all executing jobs for which USER is the command submitter and that are attached to his current working station.}
S : DJ SCH	{Display the status of all scheduled jobs for which USER is the command submitter and that are attached to his current working station.}
S : DJ X222	{Display the status of the job with RON X222.}
S : DJ EX*	{Display the status of all executing jobs that are attached to the current working station of the command submitter.}
S : DJ ALL */JONES	{Display the status of all jobs for which the USER is JONES and that are attached to the current working station of the command submitter.}
S : DJ EX */*/PROJ	{Display the status of all executing jobs submitted under project PROJ.}
S : DJ PAYROLL	{Display the status of all jobs named PAYROLL for which USER is the command submitter and that are attached to his current working station.}
S : DJ BATCH C	{Display the status of all batch jobs in class C attached to the current working station of the command submitter.}
S : DJ EX ST=ST1	{Display the status of all executing jobs for which USER is the command submitter and that are attached to station ST1.}
S : DJ (EX NSERVICE) *	{Display the status of all executing jobs that are not service jobs and that are attached to the current working station of the command submitter.}
S : DJ SEQUENCE=INCR_ ITIME DTLD	{Display detailed information about jobs for which USER is the command submitter and that are attached to his current workstation. Within each state, display the oldest jobs first.}



7.6.29 DISPLAY_LOAD (DLD)

Purpose

Display selected information on the current system load.

Syntax

```
{ DISPLAY_LOAD }
{ DLD }

[ {BRIEF} ] ]
[ {LOAD} ] ]
[ {CLASS} [[SELECT=]{* | jclass | jclass-jclass}] ] ]
[ {CLDTLD|CD} [[SELECT=]{* | jclass | jclass-jclass}] ] ]
[OPTION={CONNECT|CN} [[SELECT=]{* | jclass | jclass-jclass}] ] ]
[ {ALLOUT} [[SELECT=]{* | ALL | station} ] ] ]
[ {OUTPUT} [[SELECT=]SPACE | {* | ALL | station} ] ] ]
[ {MEM} [[SELECT=]ron ] ] ]
```




Parameters

OPTION	SELECT	Information displayed	Examples
BRIEF	-	Brief information on the job load	DLD DLD BRIEF *
LOAD	-	Job load by classes	DLD LOAD
CLASS	aa[-aa] * nil	Current load of the specified class or range of classes (all classes if * or no class specified)	DLD CLASS X DLD CLASS AA-ZZ DLD CLASS * DLD CLASS
CLDTLD (abbr. CD)	aa[-aa] * nil	Detailed information on specified job class or range of job classes (all classes if * or no class specified)	DLD CLDTLD A DLD CD BB-MM DLD CD * DLD CD
CONNECT	Aa[-aa] * nil	Current connection or specified class or range of classes (all classes if * or no class specified)	DLD CONNECT X DLD CN AA-ZZ DLD CONNECT * DLD CONNECT
ALLOUT	Name8 ALL or * nil	All jobs with outputs in READY, WAIT, HOLD or OUT states of the named station (all if ALL or *)	DLD ALLOUT DLD ALLOUT ST01 DLD ALLOUT *
OUTPUT	Name8 ALL or * nil	The number of outputs in the queues of the named station (all if ALL or *)	DLD OUTPUT DLD OUTPUT ALL DLD OUTPUT *
OUTPUT	SPACE	The percentage of free space in the SYS.OUT file	DLD OUTPUT SPACE
MEM	xnnnn	Memory used by job Xnnnn	DLD MEM X2222

(-) means that SELECT is not allowed.

Example

```
DLD CD P
{display detailed information about the job class P}
```



7.6.30 DISPLAY_MEMBER (DMB)

Purpose

Displays the States and Services of a MEMBER. This command is a part of the High Availability product (HA). For more information, see the *High Availability Administrator's Guide*.

Syntax

```
{ DISPLAY_MEMBER }
{ DMB                }

    { MEMBER }
    {          } = name4
    { MB      }

[ STATE = { 0 | bool } ]

[ { SERVICES } ]
[ {          } = { 0 | bool } ]
[ { SRV      } ]
```

Parameters

MEMBER	Name of the MEMBER to be displayed.
STATE	If entered, only the state of the MEMBER is displayed.
SERVICES	When this parameter is entered, the list of the SERVICES implemented and running on this MEMBER, their type and their state is displayed.



Rules

The CMSR of the MEMBER on which this command is entered must be running.

The MEMBER must be in the COMPLEX description.

Example

DMB	MB=MB1		Displays all information about the MEMBER named MB1.
DMB	MB=MB1	STATE	Displays only information about the current status of the MEMBER named MB1.
DMB	MB=MB1	SRV	Displays only information about the SERVICES implemented on the MEMBER named MB1.



7.6.31 DISPLAY_MFT_FACILITIES (DMFTF)

Purpose

Displays the status of the MFT facility (active or not active) and (to SYSADMIN only) indicates if the SITE.MFT file does not exist on your system. For status messages that can be generated, see "Command Output" below.

Syntax

```
{ DISPLAY_MFT_FACILITIES }  
{ DMFTF }
```

```
[ OUTAREA = name31 ]
```

Parameters

OUTAREA Name of the existing GCL global variable that is to receive the result of this command (the message number only). The global variable must be declared with:

TYPE = CHAR
LENGTH = 6 (at least) for a user in SYSADMIN project
LENGTH = 2 (at least) for a user in any other project

Command Output

```
01 CAT MFT SUPPORT FACILITY IS NOT ACTIVE  
02 CAT MFT SUPPORT FACILITY IS ACTIVE  
07 CAT MFT SUPPORT FACILITY INVALIDATION REQUESTED  
09 SITE.MFT DOES NOT EXIST
```

Messages 07 and 09 are only displayed for SYSADMIN.

Message 07 describes the temporary state occurring when a request has been made to deactivate MFT (via the command VALIDATE_MFT_FACILITIES) but certain files under the control of MFT have not yet been closed. When these files are closed, MFT moves to the status described in message 01, which means that no files or tapes remain under the control of MFT and no file or tape information is added or updated in the SITE.MFT file.



Notes to User

- The System Administrator (SYSADMIN project) and other users (with projects other than SYSADMIN) can use this command.
- To send the status of MFT to a GCL global variable using OUTAREA, make sure that you first declare the GCL global variable.
- If you specify a global variable using OUTAREA, only the *number* of the appropriate message is sent to that variable.

Examples

```
DMFTF ;
      {In GCL (IOF), you ask to display the status of MFT,
      and the following message is displayed:
      02  CAT MFT SUPPORT FACILITY IS ACTIVE }
```

```
GB MYVAR CHAR 2 ;
DMFTF OUTAREA=MYVAR ;
      {You are a user not in the SYSADMIN project. You declare
      the GCL global variable MYVAR; then you execute the
      DISPLAY_MFT_FACILITIES directive, asking that the
      result (the status of MFT) be sent to MYVAR. MYVAR now
      contains the value 02, which means that MFT facility is
      activated for cataloged files.}
```

```
EXDIR 'DMFTF';
      {You execute the DISPLAY_MFT_FACILITIES directive using
      the JCL statement EXDIR}
```



7.6.32 DISPLAY_MIRROR_MODE (DMIR)

Purpose

Displays the current operating mode for the mirrored disks. It can be either AUTOMATIC or MANUAL.

Syntax

```
{ DISPLAY_MIRROR_MODE }  
{ DMIR }
```

Parameters

None.

Rules

Refer to *Mirror Disks User's Guide*.

Examples

DMIR	Displays the message: "MR16 MANUAL MIRROR MODE" or "MR16 AUTOMATIC MIRROR MODE"
------	---



7.6.33 DISPLAY_MOT (DMOT)

Purpose

Displays the current value of the Message Of Today (MOT) which has been set by command MODIFY_MOT.

Syntax

```
{ DISPLAY_MOT }  
{ DMOT          }  
  
[ { SYSTEM     }      ]  
[ {           } = name4 ]  
[ { SYS       }      ]
```

Parameters

SYSTEM This parameter indicates the system where the command will be executed.

Rules

None.

Example

DMOT (Display the Message Of Today.)



7.6.34 DISPLAY_OUTPUT (DO)

Purpose

Displays the current status of outputs. Outputs may be selected according to the job to which they belong (OUTPUT parameter) or to their current state in the system (STATE parameter). The display shows for each output its name, priority, class, device (PR = printer), state, station, size and number of copies.

Syntax

```
{ DISPLAY_OUTPUT }
{ DO                }

[                    { (output14 [ output14 ] ...) } ]
[ OUTPUT = { output14-output14 } ]
[                    { * | ALL } ]

[ SELCLASS = a [-a] ]

[ { QUEUE } ]
[ STATE = { WAIT } ]
[ { HOLD } ]
[ { OUT } ]

[ { STATION } { * } ]
[ { } = { ALL } ]
[ { ST } { name8 } ]

[ { SELECT } { user/project } ]
[ { } = { */project } ]
[ { } { */* } ]
[ { SEL } { * } ]

[ { JOBNAME } ]
[ { } = name8 ]
[ { JNM } ]

[ SIZE = { bool | 0 } ]

[ { SYSTEM } ]
[ { } = name4 ]
[ { SYS } ]
```




Parameters

OUTPUT

Output(s) to be displayed. The following formats are allowed:

1. a list of the names of up to 16 outputs:
 - Xnnnn : nnnn } to refer to a specific output
 - } (Xnnnn:name means the first output so named)
 - Xnnnn : name } so named)
 - Xnnnn } to refer to all outputs of a job.
2. a range of outputs delimited by a lower and an upper boundary. This may take the following form only:
 - Xnnnn-Xnnnn to refer to all outputs of a range of jobs. The first ron must be lower than the second.
3. to refer to all outputs of all jobs, specify:
 - * or ALL (default when JOBNAME is specified)

If neither OUTPUT nor STATE are specified, the scope of the command is restricted to only the outputs of the OPERATOR job (or of the IOF job corresponding to the IOF user submitting the command).

SELCLASS

(meaningful only for output names in the form Xnnnn) restricts the scope of the command to those outputs that belong to a certain class or range of classes. This may be expressed as:

class for a single class
class-class for a range of classes

where class is a letter in the range A to Z.

When omitted, all classes are assumed (A-Z).

STATE

the current state of the outputs to be listed. This may be expressed as OUT (for currently being output), HOLD (for held), WAIT (for waiting to be output), or QUEUE (for any of the preceding).



STATION	(with STATE) further qualifies the outputs to be listed by naming the station to which they are attached (* and ALL stand for any station). When omitted, the default station is assumed. (with OUTPUT) restricts the scope of the command to the specified station. When omitted, or when * or ALL is specified, there is no station restriction.
SELECT	restricts the scope of the command to outputs belonging to a specified user and/or project. Not meaningful when OUTPUT specifies a list of outputs (format 1) or when STATE is specified. By default, only the outputs belonging to the command submitter are selected.
JOBNAME	restricts the scope of the command to the outputs of jobs with the name specified. If JOBNAME is specified, the default value for OUTPUT is *.
SIZE	when 1 (meaningful only when STATE is specified), the number of lines and pages of the queued outputs is displayed. Default is 0.
SYSTEM	indicates the system on which the command is to be executed.

Rules

- SELCLASS is ignored for names of specific outputs (Xnnnn:nnnn or Xnnnn:name).
- OUTPUT and STATE are mutually exclusive.
- If neither OUTPUT or STATE is specified, the command only concerns the outputs of the OPERATOR job (or the IOF job corresponding to the IOF user submitting the command).
- If SELECT is used, OUTPUT and/or JOBNAME must be used.
- STATE and JOBNAME are incompatible.
- If JOBNAME is used, the only allowable values for OUTPUT are "*", "ALL", or "Xnnnn-Xnnnn" or unspecified.



Examples

DO {display only the outputs of the OPERATOR or IOF job. See constraints above.}

DO STATE=QUEUE {display all outputs, belonging to the command submitter, in the queue of the current or specified station}

DO STATE=HOLD STATION=ST1 {display all held outputs, belonging to the command submitter, that are attached to station ST1}

DO X2345 {display all outputs of X2345}

DO X2345 Z {only those in class Z}

DO X2345 P-Z {only those of classes P to Z}

DO X2345:2 {display specified output}

DO (X2345:1 X2345:RES) {display two specified outputs}

DO STATE=QUEUE SELCLASS=W SIZE=1 {display the queued, class W outputs with their number of lines and pages}

DO STATE=WAIT {display all queued outputs in the WAIT state}

DO X123-X2617 SELECT=U1/P1 {display the outputs, belonging to user U1 of project P1, of the jobs whose RON is in the range X123 to X2617}

DO JOBNAME=JOB1 {display all the outputs, belonging to the command submitter, of all jobs named JOB1}



7.6.35 DISPLAY_POWER_TIMER (DPWR) (DPWRM)

Purpose

This command displays the values of both GCOS 7 power timers:

The Power Off Timer, indicating at what time the system will shut down, then powered off;

The Power On Timer, indicating at what time the system will power on.

Syntax

```
{ DISPLAY_POWER_TIMER }  
{ DPWR }  
{ DPWRM }
```

Rule

This command is not available on DPS 7000/2X0 or 3X0.

Example

DPWR (Displays values of power-off and power-on timers.)



7.6.36 DISPLAY_REGULATION_PARAMETER (DRP)

Purpose

This command displays the main memory size in Mbytes and the dedicated I/O cache size in Mbytes. It also displays the current level (Full or Basic) and state (Normal or Degraded) of the Automatic Resource Manager (ARM).

See the *ARM User's Guide* for full details.

Syntax

```
-----
{ DISPLAY_REGULATION_PARAMETER }
{ DRP                             }

[ { SYSTEM }                       ]
[ {           } = name4            ]
[ { SYS      }                     ]
```

Parameters

SYSTEM The system on which the command is executed.

Rules

None.

Example

DRP {displays the following information:

```
MEM = 128MB
IOMEM=256MB
LEVEL = FULL                  STATE = NORMAL
```

where MEM is the main memory size of the system and IOMEM is the dedicated I/O cache size. The physical memory size is MEM + IOMEM}



7.6.37 DISPLAY_REQUEST (DR)

Purpose

Display all pending requests either from the system or from jobs in execution.

Syntax

```
{ DISPLAY_REQUEST }
{ DR }

[NUMBER = { bool | 0 } ]

[ { SYSTEM } ]
[ { } = name4 ]
[ { SYS } ]
```

Parameters

NUMBER	When 1, only the number of pending requests is displayed; when 0 or not specified, all pending requests are displayed.
SYSTEM	This parameter indicates the system where the command will be executed.

Rules

None.

Examples

DR	{display all pending requests}
DR NUMBER	{display the number of pending requests}



7.6.38 DISPLAY_RESTART_OPTIONS (DRSO)

Purpose

Displays some of the system initialization parameters.

Syntax

```
{ DISPLAY_RESTART_OPTIONS }  
{ DRSO }  
  [ ISLOPT = {bool | 0}]
```

Parameters

ISLOPT

When 0 (default value), displays:

- the automatic repeat option,
- the automatic restart option,
- the silent restart option,
- the initialization device supporting the system disk.

The meaning of this options is fully described in the example below.

When 1, displays:

- the initialization options of the current GCOS 7 session,
- additional sorted lists of current resident and shared disks (if any),
- the current TIMEDEV value,
- the four options mentioned above (when 0).

The initialization options are BKST, DATE, DISK, DUMP, DUMP_CD, HA, INIT, LIB, MAM, NSYSSTUP, PAGESIZE, PVMF, REFORMAT, REPEAT, RESIDENT, RESTART, RESTORE, SHARED, SPOOL, SYSQM, TIMEDEV, TRC and TVMF. These options are completely described in Chapter 3 *Starting and Terminating a GCOS 7 Session*.

Note that the initialization options values displayed are those entered at IL01 question. Modifications on resident or shared disks, TIMEDEV and automatic restart or repeat options are reflected in the information displayed after the initialization options.



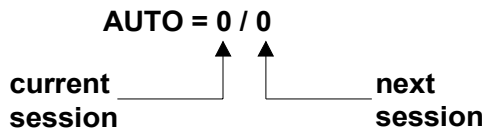
Rules

None.

Example

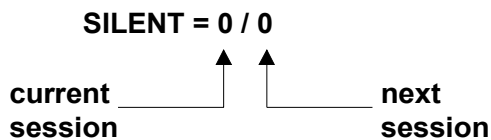
```
DRSO                                {display some of the initialization parameters}
16.13    AUTO=0/0 SILENT=0/0 REPEAT=0
          SYSTEM DUMP AVAILABLE
          ANTESP AUTO SWITCH IS OFF INIT=MS17
```

The boolean values in the display indicate the initialization options used for the session; all possible values are given below:



current-auto if this parameter is 0, the current ISL has been done in manual mode; if 1, the current ISL has been done in automatic mode.

next-auto If this parameter is 0, the next ISL will be done in manual mode; if 1, the next ISL will be done in automatic mode.



current-silent If this parameter is 0, the current ISL has not been performed silent; if 1, the current ISL has been performed silent.

next-silent If this parameter is 0, the next ISL will not be performed silent; if 1, the next ISL will be performed silent.



REPEAT	If this parameter is 0, the next system shutdown or crash requires a manual intervention from an operator to restart the system; if 1, the next system shutdown or crash is followed by a restart without any operator intervention.
ANTESP	or ASPA (Ante Service Processor). On DPS 7000/2x0/3x0, sets a number of configuration parameters. See Section 2.3.3 <i>ASPA Configuration Parameters on the DPS 7000/2x0 and /3x0</i> .

Example

DRSO ISLOPT=1

```

INITIALIZATIONS OPTIONS 17:44:02 FEB 26, 1997
BKST          = /, BFR055
DATE          = MM.DD.YY/HH.MM
DISK          = ISL_OPTIONS..RELOAD_OPTIONS:BFS753
DUMP          = CONS
DUMP_CD       =
HA            = ON
INIT          = NAUTO
LIB           = /, BFR054
MAM           = YES
NSYSSTUP     =
PEGESIZE     = 11*6
PVMF         = /
REFORMAT     =
REPEAT       = OFF
RESIDENT     = BFS753, BFR054, BFR055
RESTART      = CLEAN
RESTORE      = SYS.SYSTEM:BFS753
SHARED       =
SHARED2      =
SHARED3      =
SHARED4      =
SHARED5      =
SHARED6      =
SPOOL        = 0, 2, 8
SYSQM        = WARM
TIMEDEV      = 0
TRC          = ON
TVMF         = /

```



CURRENT LIST OF RESIDENT DISKS

BFR054
BFR055
BFS753

CURRENT LIST OF SHARED DISKS

BFU023

CURRENT TIMEDEV = +01:00

17.44 AUTO = 1/0 SILENT = 0/0 REPEAT = 0
 SYSTEM DUMP UNAVAILABLE
 ANTESP AUTO SWITCH IS OFF INIT = MS53



7.6.39 DISPLAY_RMS (DRMS)

Purpose

This command displays information on the Remote Maintenance Service.

Syntax

```
{ DISPLAY_RMS }  
{ DRMS }
```

Parameters

None.

Rules

This function is only available on DPS 7000/An and DPS 7000/500/700 systems.

Example

DRMS Various information concerning the current state of
RMS is displayed on the operator's screen.



7.6.40 DISPLAY_SECURITY_OPTIONS (DSO)

Purpose

This command displays the security options of a site declared by the SECOPT and SADMOP statements of CONFIG, and recorded in the SITE.CATALOG.

Syntax

```
{ DISPLAY_SECURITY_OPTIONS }  
{ DSO }
```

Parameters

None.

Rules

None.

Example

DSO (Produces the display shown below:)

```
hh.mm SECURITY OPTIONS FOR: sitename ON: date ARE:  
(1) PROJECT/USER CHECKS={YES|NO}  
(1) PROJECT/BILLING CHECKS={YES|NO}  
(1) PROTECTED SYSTEM (ACCESS RIGHTS)={YES|NO}  
  
(2) hh.mm SA7LOGON={YES|NO} SA7ADMIN={YES|NO} SA7NOCSL={YES|NO}  
(3) NETSEC ={YES|NO} CHPW =YES|NO ERASE ={YES|NO}  
  
hh.mm SYSADMIN OPTIONS ARE:  
(4) CHKJSADM={YES|NO} GCLKPROJ={YES|NO} GCLKSADM={YES|NO}  
(5) GCSADMOP={YES|NO} MIRROR={YES|NO}
```

Lines (1), (2), and (3) display the parameters of the SECOPT option of CONFIG.
Lines (4) and (5) display the parameters of the SADMOP option of CONFIG.



Description of Message Entries

PROTECTED SYSTEM (ACCESS RIGHTS):

This option defines the access rights recorded in the SITE.CATALOG.

=YES File access is checked with the rights defined in the SITE.CATALOG.

=NO File access is unrestricted for all users.

CHKJSADM (SADMOPT parameter):

This option defines the restriction of job submission under SYSADMIN.

=YES: GCOS 7 checks that users running jobs for SYSADMIN are cataloged under the SYSADMIN project.

=NO: Job submission under SYSADMIN is unrestricted.

CHKPW (SECOPT parameter):

This option determines if the password is to be checked; *applicable where the project of the user submitting the request has been defined with either UFT, DJP, or SPOOL in the catalog.*

=NO: default: password is not checked.

=YES: the site receiving the request checks the password of the user submitting the job (*the password can be provided when the command is submitted*).

GCLKPROJ (SADMOPT parameter):

This option defines the user access to GCL procedures specifying LOCK in the PROC statement.

=YES: access is available to any user under the same project as the user compiling these procedures.

=NO: access is limited only to the user compiling these procedures.



GCLKSADM (SADMOPT parameter):

This option defines the SYSADMIN access to GCL procedures specifying LOCK in the PROC statement.

- =YES: users under the SYSADMIN project have the same rights of access as the user compiling these procedures.
- =NO: users under the SYSADMIN project have no additional access rights.

GCSADMOP (SADMOPT parameter):

This option defines the SYSADMIN access to Main Operator commands through EXDIR.

- =YES: users under the SYSADMIN project have the same rights as the Main Operator.
- =NO: users under the SYSADMIN project are normal IOF users.

MIRROR (SADMOPT parameter):

Defines if MIRRORED disks are used.

- =YES: duplicate WRITES on 2 disks.
- =NO: no MIRRORED disks.

NETSEC (SECOPT parameter):

This option defines how connections are to be checked.

- =YES: connections always go through the front-end processor which performs security checks using LACS software.
- =NO: local connections are checked by GCOS 7.

SA7ADMIN (SECOPT parameter):

This option defines how the user administration of the site is performed.

- =YES: administration is performed by SECUR'ACCESS.
- =NO: administration is performed by SYSADMIN through catalog management (MNCAT).

**SA7LOGON** (SECOPT parameter):

This option defines how the user's access rights are to be checked.

- =YES: user rights to access TDS/IOF applications are checked by SECUR'ACCESS at connection time.
- =NO: access rights are checked by GCOS 7.

SA7NOCSL (SECOPT parameter):

This option defines if SECUR'ACCESS checks the identity of the user logging on to the local system console.

- =NO: the user's identity is checked by SECUR'ACCESS; **meaningful only if SA7LOGON=YES.**
- =YES: the user's identity is checked by GCOS 7.

For further information, refer to the *GCOS 7 System Installation Configuration and Updating Guide*.

Example

S: DSO;

```

14.16 SECURITY OPTIONS FOR : BPA6 ON:   OCT 10, 1995 ARE:
      PROJECT/USER CHECKS              = YES
      PROJECT/BILLING CHECKS            = NO
      PROTECTED SYSTEM (ACCESS RIGHTS) = YES

14.16  SA7LOGON = NO      SA7ADMIN = NO      SA7NOCSL = NO
14.16  NETSEC   = NO      CHKPW      = NO      ERASE     = NO

      SYSADMIN OPTIONS ARE:
      CHKJSADM = NO      GCLKPROJ = NO      GCLKSADM = NO
      GCSADMOP = YES      MIRROR   = YES

```



7.6.41 DISPLAY_SERVICE (DSRV)

Purpose

The operator uses this command to obtain the state of the different instances of a SERVICE on the MEMBERS of a COMPLEX. This command is a part of the High Availability product (HA). For more information, see the *High Availability Administrator's Guide*.

Syntax

```
{ DISPLAY_SERVICE }
{ DSRV           }

    { SERVICE }
    {           } = name8
    { SRV      }

[ {   { 0 } } ]
[ STATE = {   } ]
[ { bool } ]

[ { MAIN_SERVICES } { 0 } ]
[ {           } = {   } ]
[ { MSRV          } { bool } ]

[ { USED_SERVICES } { 0 } ]
[ {           } = {   } ]
[ { USRV          } { bool } ]

[ { DETAILED } { 0 } ]
[ {           } = {   } ]
[ { DTLD     } { bool } ]
```

Parameters

SERVICE	Name of the SERVICE.
STATE	When entered, gives only the two states of this SERVICE (required and effective).
USED_SERVICES	When entered gives only the list of the USED_SERVICESs of this SERVICE.
MAIN_SERVICES	When entered gives only the list of the MAIN SERVICES of this SERVICE.



DETAILED When entered, gives more detailed information on the states of this SERVICE.

Rules

The CMSR of the MEMBER on which this command is entered must be running.

The SERVICE must be in the COMPLEX description.

Examples

DSRV SRV=TDS1

Displays all information about the SERVICE named TDS1.

DSRV SRV=TDS1 USRV

Displays only information about the USED_SERVICES of the SERVICE named TDS1.

DSRV SRV=TDS1 STATE

Displays only information about the state of the SERVICE named TDS1.



7.6.42 DISPLAY_SSI (DSSI)

Purpose

Displays the status of the SSI (Storage Server Interface). It indicates whether the SSI processes on OPEN 7 are active or not. SSI is a component of the CLX product (Cartridge tape Library uniX server).

Syntax

```
{ DISPLAY_SSI }  
{ DSSI }
```

Parameters

None

Example 1

DSSI is entered while SSI is inactive

```
S: DSSI  
11.15 X549 IN CLX_SSI USER=OPERATOR CLASS=P SPR=6 STATION=BCA9  
11.15 X549 STARTED CLX_SSI OPERATOR P  
JB08 X549.1 STEP H_CLX_CMDSSI XPR=8 PGID=22  
TU70 ACSEL AND SSI NO MORE RUNNING  
11.15 X549.1 COMPLETED CLX_SSI OPERATOR P
```

Example 2

DSSI is entered while SSI is active

```
S: DSSI  
11.17 X551 IN CLX_SSI USER=OPERATOR CLASS=P SPR=6 STATION=BCA9  
11.17 X551 STARTED CLX_SSI OPERATOR P  
JB08 X551.1 STEP H_CLX_CMDSSI XPR=8 PGID=22  
TU70 ACSEL AND SSI ARE RUNNING  
11.17 X551.1 COMPLETED CLX_SSI NA_OP P
```



7.6.43 DISPLAY_STORAGE_FACILITIES (DSF)

Purpose

Displays the status (active or not active) and the level (basic or full) of the VOLSET and QUOTA facilities.

Syntax

```
{ DISPLAY_STORAGE_FACILITIES }  
{  
{ DSF
```

```
[ OUTAREA = name31 ]
```

Parameters

OUTAREA the name of the GCL global variable that will receive the message numbers of the messages output by the DSF command. The global variable must be declared with:

```
TYPE = CHAR  
LENGTH = 6 (at least) for a SYSADMIN user  
LENGTH = 4 (at least) for any other user
```

When DSF is executed, the global variable referenced by OUTAREA is set as follows:

- the first two characters give the status of the QUOTA facility (01 or 02)
- the next two characters give the status of the VOLSET facility (03, 04, or 05)
- the fifth and sixth characters are available only if the command submitter is a SYSADMIN user; they give complementary information (06, 07, 08, 09, 10, or 11)

If OUTAREA is omitted, the result is displayed on the current output media defined for the active domain.

Rules

The GCL variable name specified in OUTAREA must be different from the keyword names of GCL procedures.



Command Output

```
01 QUOTA FACILITY IS NOT ACTIVE
02 QUOTA FACILITY IS ACTIVE
03 VOLSET FACILITY IS NOT ACTIVE
04 BASIC VOLSET FACILITY IS ACTIVE
05 FULL VOLSET FACILITY IS ACTIVE
06 VOLSET FACILITY VALID BUT NOT ACTIVE, SITE VOLSET NOT CREATED
07 VOLSET FACILITY IS NOT VALID
08 SITE.QUOTA FILE DOES NOT EXIST
09 THE SYSTEM HAS ITS OWN VOLSET
10 THE SYSTEM HAS ITS OWN SHARABLE VOLSET
11 SITE VOLSET IS SHARED AND HAS BEEN DEFINED ON THE OTHER SYSTEM
```

Messages 06 to 11 are not issued to non-SYSADMIN users.

Examples

```
DSF      {for a non-sysadmin user, this displays:
          02 QUOTA FACILITY IS ACTIVE
          03 VOLSET FACILITY IS NOT ACTIVE}

          {for a sysadmin user, this displays:
          02 QUOTA FACILITY IS ACTIVE
          03 VOLSET FACILITY IS NOT ACTIVE
          06 VOLSET FACILITY IS VALID BUT NOT ACTIVE; SITE VOLSET
            NOT CREATED}

GLOBAL MY_VAR CHAR 6
DSF OUTAREA=MY_VAR
          {for a non-sysadmin user, the variable MY_VAR=0203
          for a sysadmin user, the variable MY_VAR=020306}
```



7.6.44 DISPLAY_SUBMITTER (DS)

Purpose

Display the current status of selected jobs, with the selection based on their respective submitter.

Syntax

```
{DISPLAY_SUBMITTER}
{DS
}

[
    {jstate
}
[
    {jtype
}
[ JOBID = { (jstate jtype) } ]
[
    {Xnnn
}
[
    {name8
}

[ {SELECT} {jclass} [ {usernm} [ {projnm} ] ] ]
[ { } = { } [ / { } [ / { } ] ] ]
[ {SEL } { * } [ { * } [ { * } ] ] ]

[ {STATION } { * } ]
[ { } = { ALL } ]
[ {ST } { name8 } ]

[JOBNAME = name8]

[
    {200 } ]
[LIMIT = {20 } ] (See parameter description for default value)
[
    {dec4} ]

[ {SEQUENCE} { PRIORITY | DECR_ITIME | INCR_ITIME } ]
[ { } = { | | } ]
[ {SEQ } { |DIT |IIT } ]

[ {DETAILED} ]
[ { } = { bool | 0 } ]
[ {DTLD } ]
```



Parameters

JOBID	<p>the jobs to be displayed designated by their Run Occurrence Number (Xnnnn), name (name-8), current state (jstate), type (jtype), or current state and job type.</p> <p>jstate must be one of the following: ALL, SCH, EX, SUSP, OUT, IDLE, IN, HOLD</p> <p>jtype be one of the following: ALL, BATCH, IOF, SERVICE, NSERVICE (NSERVICE means all jhobs except service jobs.)</p> <p>When JOBID is omitted, ALL is assumed.</p>
SELECT	<p>restricts the directive to the jobs that belong to the specified combination of class/submitter-name/project. * is all jobs. When omitted, all jobs submitted by the terminal user are displayed.</p>
STATION	<p>restricts the list of jobs displayed to those that are attached to the named station (* or ALL stand for any station). When omitted, the current working station of the command submitter is assumed.</p>
JOBNAME	<p>the job to be designated by name (name-8).</p>
LIMIT	<p>maximum number of jobs to be displayed. Beyond that limit, only the number of jobs per class is displayed. The default and maximum values are: When DETAILED=1, default 20, maximum 100 When DETAILED=0, default 200, maximum 850.</p>
SYSTEM	<p>indicates the system where the command is to be executed.</p>



SEQUENCE	<p>specifies the order in which jobs are to be displayed for each job state. The sort criterion can be the scheduling priority (keyword PRIORITY) or the introduction time (keywords DECR_ITIME and INCR_ITIME). The display order for each job state is determined by:</p> <ul style="list-style-type: none">- the scheduling priority (PRIORITY), from highest to lowest. The introduction time is used for second level sorting within each priority level. This is the default display criterion.- or the introduction time alone, (PRIORITY not specified):<ul style="list-style-type: none">DECR_ITIME = from most recent to oldestINCR_ITIME = from oldest to most recent.
DETAILED	<p>(abbreviation DTLD) specifies whether detailed job information is to be displayed. The information returned depends on the job state:</p>
HOLD	<ul style="list-style-type: none">- Hold count- Introduction time
EX/ SUSP	<ul style="list-style-type: none">- Introduction time- Switches- Full load module name or step state- System internal identifier associated with the step- CPU and elapsed times for the current step- Number of processes, Backing store- Used and fixed instantaneous memory sets. <p>It is not always possible to provide all of the above information for jobs in state EX or SUSP, in particular the ARM dimension (e.g. IOF, BATCH) and the step-specific information (load module name, step CPU and elapsed time, number of processes, backing store, memory sets). This is the case at step initialization (if the step is waiting for a resource statically assigned by a JCL statement or for resources in a dimension), and at step termination.</p>
OUT	<p>The introduction and termination times. These times may be unknown if the job is in the OUT state, and if they were kept after SYS.OUT recovery at system RESTART.</p>



IN, SCH, IDLE The introduction time.

When DETAILED=1, LIMIT may not exceed 100 and the default value is 20.

Rules

- when JOBID is in the form Xnnnn, SELECT, STATION and LIMIT are ignored.
- JOBID and JOBNAME are mutually exclusive.

Examples

S:DS	{Display the status of all jobs submitted by the terminal user that are attached to his current working station.}
S:DS EX	{Display the status of all executing jobs submitted by the terminal user that are attached to his current working station.}
S:DS SCH	{Display the status of all scheduled jobs submitted by the terminal user that are attached to his current working station.}
S:DS X222	{Display the status of the job with RON X222.}
S:DS EX*	{Display the status of all executing jobs that are attached to the current working station of the command submitter.}
S:DS ALL */JONES	{Display the status of all jobs submitted by JONES that are attached to the current working station of the command submitter.}
S:DS EX */*/PROJ	{Display the status of all executing jobs submitted under project PROJ.}
S:DS PAYROLL	{Display the status of all jobs named PAYROLL submitted by the terminal user that are attached to his current working station.}
S:DS BATCH C	{Display the status of all batch jobs in class C attached to the current working station of the command submitter.}



S:DS EX ST=ST1 {Display the status of all executing jobs submitted by
the terminal user that are attached to station ST1.}

S:DS (EX NSERVICE) *

 {Display the status of all executing jobs that are not
service jobs and that are attached to the current
working station of the command submitter.}

S:DS (HOLD BATCH) DETAILED
 {Display detailed information about held batch jobs
for which USER is the command submitter and that
are attached to his current workstation. }



7.6.45 DISPLAY_SYSTEM_TRACE (DSYSTR)

Purpose

This command displays, on the main console, trace object identifiers and their status. If the selected object is a single buffer or domain, the display details its attributes; otherwise, a list of the selected object identifiers is produced. The command can be issued at any time, but system trace files exist only during collector execution.

NOTE:

Use this command only under the direction of your Bull support engineer.

```
{ DISPLAY_SYSTEM_TRACE }
{ DSYSTR                }

[ { DOMAIN } { ALL } ]
[ { DOM    } = {      } ]
[ { TDOM   } { name63 } ]

[ { BUFFER } { ALL } ]
[ {       } = {      } ]
[ { BFR    } { name6  } ]

[ FILES = bool ]
```

Parameters

DOMAIN: the name of a system domain, or if ALL (the default) is specified, all domains. Maximum length is 63 characters.

BUFFER: the name of a system trace buffer, or if ALL is specified, all system trace buffers. Maximum length is 6 characters.

FILES: if 1, system trace collector file names are displayed; if 0, the file names are not displayed.



Rules

The parameters DOMAIN, BUFFERS, and FILES are mutually exclusive.

Examples

```
DSYSTR                {Display active domain names.}
DSYSTR DOMAIN=JOBM    {Display attributes of JOBM trace domain}
DSYSTR BUFFER=ALL     {Display the names of all trace buffers}
DSYSTR BUFFER=BFRSYS  {Display the characteristics of trace buffer BFRSYS}
DSYSTR FILES          {Display system trace collector file names}
```



7.6.46 DISPLAY_TERMINAL_OUTPUT (DTO)

Purpose

Displays the characteristics and status of outputs to be printed by the Generalized Terminal Writer. Refer to the *Generalized Terminal Writer User Guide* for a complete description of the Generalized Terminal Writer.

Syntax

```
{DISPLAY_TERMINAL_OUTPUT}
{DTO

    [{OUTPUT = Rnnnn }]
    [{
    }]
    [{USER = {name12}}]
    [{
        {ALL }
    }]
    [{
    }]
    [{RON = Xnnnn }]
    [{TWDEST = name12}]

    [SELCLASS = a[-a]]

    [
        {HOLD}]
    [
        {FRZN}]
    [STATE = {WAIT}]
    [
        {OUT }]
    [
        {MNT }]
    [
        {CNCT}]

    [DETAILED = {bool | 0}]
```

Parameters

- OUTPUT** the name of the output to be displayed. A Generalized Terminal Writer output name has the form Rnnnn and is provided as the result of the DPRINT command (with parameter TWDEST). When none of OUTPUT, USER, RON and TWDEST is specified, all outputs belonging to the terminal user are displayed.
- USER** when specified, all Generalized Terminal Writer outputs belonging to the user specified by USER are displayed. When none of OUTPUT, USER, RON, and TWDEST is specified, all outputs belonging to the terminal user are displayed



RON	when specified, all Generalized Terminal Writer outputs submitted by the job with this Run Occurrence Number are displayed. When none of OUTPUT, USER, RON and TWDEST is specified, all outputs belonging to the terminal user are displayed.
TWDEST	when specified, all Generalized Terminal Writer outputs to be printed on this terminal or pool of terminals are displayed. When none of OUTPUT, USER, RON, and TWDEST is specified, all outputs belonging to the terminal user are displayed.
SELCLASS	restricts the scope of the directive to those outputs that belong to a certain class (SELCLASS = a) or range of classes (SELCLASS = a-a).
STATE	restricts the scope of the directive to those outputs that have the specified status: OUT (being printed), WAIT (waiting for printing), HOLD (held), FRZN (frozen), CNCT (connecting to printer), or MNT (wait for paper to loaded).
DETAILED	when 1, detailed information for each output is provided; when 0 (the default value) a one line status message is given for each selected output.

Rules

OUTPUT, USER, RON and TWDEST are mutually exclusive.

Examples:

DTO	{display all the terminal user Terminal Writer outputs}
DTO R12345 DETAILED	{display detailed information for the output R12345}
DTO STATE=WAIT	{display the terminal user's output waiting for printing}
DTO TWDEST=POOL3 STATE=WAIT	{display all outputs waiting for printing on pool POOL3}



7.6.47 DISPLAY_TERMINAL_WRITER (DTW)

Purpose

Displays miscellaneous information relating to the Generalized Terminal Writer. Refer to the *Generalized Terminal Writer User Guide* for a complete description.

Syntax

```
{ DISPLAY_TERMINAL_WRITER }
{ DTW

    [ { TERM } ] [ { name12 } ]
    [ {          } ] = [ { { * | ALL } } ]
    [ { TN   } ] [ { POOL   } ]

    [ ID = name8 ]

    [ NODE = name8 ]

    [ DRIVER = name4 ]

    [ { TWRITER }          ]
    [ {          } = bool ]
    [ { TW      }          ]

    [ DETAILED = { bool | 0 } ]

    [ FORM = { * | name6 } ]
```

Parameters

TERM	the name of the terminal for which information is to be displayed. If * or ALL is specified, all terminals are assumed. If POOL is specified, information concerning the terminal pools is displayed.
ID	when specified, the information to be displayed is that relative to the terminal whose identification in the generation is given by ID parameter.
NODE	(only when ID is specified) qualifies the node for which ID is specified.
DRIVER	display information for the named driver.



TWRITER	when 1, general information on the Generalized Terminal Writer is displayed.
DETAILED	when 1, information provided is detailed; when 0 (the default value), brief information is provided.
FORM	the name of the form for which information is to be displayed. If the asterisk (*) is specified, the information on all the forms is displayed.

Rules

- TERM, ID, DRIVER and TWRITER are mutually exclusive.
- NODE may be used only in conjunction with ID.
- When TWRITER is 1, DETAILED is ignored.

Examples

DTW POOL	{display the pools}
DTW DRIVER=TW02	{display driver TW02}
DTW *	{display all terminals}
DTW TWRITER	{display global information on Terminal Writer}



7.6.48 DISPLAY_TIME (DTM)

Purpose

Displays the date and time of day.

Syntax

```
-----  
{ DISPLAY_TIME }  
{ DTM }
```

Parameters

None.

Rules

None.

Example

DTM {display the date and time}



7.6.49 DISPLAY_USER_REQ (DUR)

Purpose

Displays the current status of file transfer or jobset requests.

Syntax

```
{ DISPLAY_USER_REQ }
{ DUR
  [
    [ REQUEST = { ALL } ]
    [ { name8 [name8] ... } ]

    [ { SUBMITTER } ]
    [ { } = usernm [.projnm] ]
    [ { SELECT } ]

    [ USER = usernm [.projnm] ]

    [ JOBNAME = bool ]
    [ DETAILED = {0|bool} ]
```

Parameters

REQUEST a list of up to 16 names of requests whose statuses are to be displayed. Each name may take one of the following forms:

- Jnnnnnnn a Jobset Occurrence Number (JON),
- Fnnnnnnn a File transfer Occurrence Number (FON),
- name-8 the symbolic name of a request as specified by the NAME parameter of the ENTER_JOB_REQ or ENTER_FILETRANS_REQ commands.

JONs and FONs are provided as results of the ENTER_JOB_REQ and ENTER_FILETRANS_REQ commands. They are unique identifiers for the corresponding requests.



SUBMITTER	(meaningful only when REQUEST is ALL) restricts the display to the requests submitted by the named user, possibly further restricted to the named project. When SELECT is not specified, the terminal user name is assumed.
USER	This is the same as the preceding parameter except that the USER address requests submitted on behalf of the secondary submitter (parameter USER= of EJR).
JOBNAME	This parameter is useful to locate specific job requests when there are a large number of requests. It shows either the subfile name or the file name for job requests.
DETAILED	is meaningful only when REQUEST=ALL, if = 1: displays all information as for REQUEST=name8 (Request-name), if = 0: (default value): displays only the list of requests.

Rules

- SELECT and DETAILED are ignored when REQUEST is not ALL.
- DETAILED is meaningful when REQUEST is ALL.
- SUBMITTER and USER are mutually exclusive.

Examples

DUR or DUR ALL	{display the list of all terminal user's requests}
DUR ALL JONES	{display the list of all JONES' requests}
DUR SELECT=JONES . PREP	{display the list of all JONES' requests under project PREP}
DUR (F22 J44 RQP)	{display three requests}
DUR ALL DETAILED	{displays the list of all terminal user's requests with detailed information}



7.6.50 DISPLAY_XL_CLASS (DXLC)

Purpose

Displays the current values associated with an XL class. See the *ARM User's Guide* for full details.

Syntax

```
{ DISPLAY_XL_CLASS }
{ DXLC                }

[ { XL_CLASS }    { dec2 } ]
[ {           } = {     } ]
[ { XLC         } { *   } ]

[ { SYSTEM }      ]
[ {           } = name4 ]
[ { SYS      }      ]
```

Parameters

XLC	The identity of the Execution Level Class to be displayed (0 to 10). If it is not specified then all the classes are displayed.
SYSTEM	This parameter indicates the system where the command will be executed.



Example

S: DXLC XLC=3;

00.35	XLC	ATTR	<HPRTYRANGE>		EPRTY	MAXCPU
	3	FS	12	13	12	90

- ATTR** There are three possible control modes: Fixed non-sliced (F), Fixed sliced (FS), or Optimized (O).
- HPRTYRANGE** Range of hardware priorities that a process is allowed belonging to this execution level class, ranging from 0 to 15.
- EPRTY** This is the entry priority of a process. It is applicable only when ATTR=O and also it is the starting value of HWPRTY (See MDXLC).
- MAXCPU** The maximum CPU consumption rate a process may achieve before it is penalized, expressed as a percentage.



7.6.51 DPRINT (DPR)

Purpose

Requests the printing of a file or of subfiles on a printer; requests the printing of a file or subfile by the Generalized Terminal Writer (refer to the *Generalized Terminal Writer User Guide*).

Syntax

```
{ DPRINT }
{
{ DPR
}
```

```
[ SUBFILES = (star31 [star31] ...) ]

[ LIB = lib78 ]

[ CLASS = a ]

[ PRIORITY = { 0|1|2|3|4|5|6|7 } ]

[ WHEN = { IMMED | END } ]

[ DEST = [name8.] name8 ]

[ TWDEST = { name12 | * } ]

[ NAME = name8 ]

[ BANINF = (char12 [char12 [char12 [char12]]] ) ]

[ COPIES = dec3 ]

[ NUMBER = bool ]

[ INFILE = file78 ]

[ PAGE = (dec5 [-dec5] [dec5 [-dec5]]... [dec5 [-$]]) ]

[ OUTPUT = ( output14 [ output14] ...) ]

[ HOLD = bool ]

[ BANNER = { NO|YES|JOB|OUTPUT|COPY|MEMBER } ]

[ SILENT = bool ]
```



```
[ { DEVCLASS } ]
[ {           } = device-class ]
[ { DVC       } ]

[ { MEDIA } ]
[ {         } = char6 ]
[ { MD     } ]

[ DELETE = bool ]

[ SKIP = bool ]

[ DATAFORM = { SARF|SSF|DOF|ASA } ]

[ PRNAME = { bool | 0 } ]
```

Parameters

SUBFILES	a list of up to 10 star-names denoting the names of the subfiles of LIB to be printed or punched. If you omit the SUBFILES parameter, you must specify a sequential file using the INFILE parameter.
LIB	the library holding the subfiles. When omitted, the default output SL library (#SLIB) is assumed. When DPRINT is launched from the main console or if TWDEST is specified, a cataloged library must be cataloged in either an auto.attach catalog or in the SITE.CATALOG. When TWDEST is specified, the library cannot be TEMPRY.
CLASS	the output class (a letter in the range A to Z). The default is defined by the installation.
PRIORITY	a number in the range 0 to 7. The lower the number, the higher the urgency.
WHEN	IMMED immediate enqueueing of the request, END enqueueing deferred to the end of the current session.
DEST	destination of the output expressed as [host-name.] station-name. See the <i>IOF Terminal User's Manual</i> , part 1, for details.



TWDEST	(Generalized Terminal Writer only) the name of the Generalized Terminal Writer terminal or pool of terminals where the output is to be printed. When * is specified, the default value defined when the Generalized Terminal Writer is generated, or modified by the MODIFY_PROFILE directive (parameter TWDEST), is assumed.
NAME	to appear in the result of the DISPLAY_OUTPUT command.
BANINF	up to 4 12-character texts to appear on the output banner (from bottom to top).
COPIES	number of copies in the range 1 to 254.
NUMBER	if 1, generate line numbers on the output.
INFILE	the file containing the text to be printed.
PAGE	a list of up to 10 pages or ranges of pages of the specified file to be printed. The ending value of the last range may be \$, which means that the upper limit is the end of the file.
OUTPUT	a list of up to 10 outputs to select specific reports of a job. OUTPUT may take the following forms: Xnnnn: nnnn or Xnnnn: name to select a specific output Xnnnn to select all outputs of a job. OUTPUT applies only if the specified file was produced via the Output Writers.
HOLD	if 1, output is held. The command RELEASE_OUTPUT (or RELEASE_TERMINAL_OUTPUT, if TWDEST is specified) can subsequently be used to cancel the effect of this parameter.



BANNER	<p>if NO, banners are not printed at the beginning of each output.</p> <p>if YES, resets effect of NO.</p> <p>if JOB, asks for job level banners (default if WHEN=END).</p> <p>if OUTPUT, asks for output level banners.</p> <p>if COPY, asks for banners at output copy level. This is the default unless WHEN=END.</p> <p>if MEMBER, asks writer to insert banners between each member.</p>
SILENT	<p>(Generalized Terminal Writer only) if 1, no message is issued to the user when printing starts or is completed.</p>
DEVCLASS	<p>the class of the device to be used.</p>
MEDIA	<p>when TWDEST is not specified, the printer belt to be used. If not specified, the default is defined by the installation. When TWDEST is specified, the paper to be used on the terminal; if omitted, the default is the form currently mounted on the terminal specified by TWDEST.</p>
DELETE	<p>if 1, delete subfiles after printing.</p>
SKIP	<p>if 1 or omitted, any paper move instructions are obeyed when the file is printed. If 0, when the file is printed, every skip function is reduced to "skip to next line" (move to the next line).</p>
DATAFORM	<p>data format, as follows:</p> <ul style="list-style-type: none">SARF Standard Access Record FormatSSF Standard System FormatDOF Device Oriented FormatASA American Standard
PRNAME	<p>(meaningful only when SUBFILES is specified) if 1, the name(s) of the subfile(s) and their libraries are printed on the first line of the output. If 0 (default), these names are not printed.</p>



Rules

- INFILE and SUBFILES are mutually exclusive. One of the two must be specified.
- LIB and DELETE may only be used when SUBFILES is used.
- NAME, WHEN, DEST, BANNER, PAGE, OUTPUT, SKIP, and DEVCLASS are exclusive of TWDEST.
- SILENT may be used only when TWDEST is specified.
- BANINF and BANNER=0 are mutually exclusive
- TWDEST and DATAFORM=DOF are mutually exclusive
- PAGE and OUTPUT may be used only when INFILE is specified.
- The parameters specified in \$JOB with the EJ command or an anticipated MO (MODIFY_OUTPUT) command override those specified in a batch job (OVL SYSOUT or WRITER), or in a IOF directive (DPR).

Examples

```
DPR SF1 {print member of current SL output library}
```

```
DPR SF *MYPROJ.MYLIB
      BANINF=(,PR2,PRX)
      NAME=PR2
      COPIES=2
      {print two copies of all members beginning with SF, give a name
      to the output and specify banner lines 3 and 4}
```

```
DPR (S* T*) MYPROJ.MYLIB
CLASS=Z
      {print all members beginning with S or T in class Z}
```

```
.
DPR INFILE=MYPROJ.MYF
      {print a sequential file}
```

```
DPR MYSF MYPROJ.MYLIB TWDEST=*
      {print subfile MYSF of the specified library on the default
      Terminal Writer terminal attached to the user}
```

```
DPR SUBFILES=SBFILE1 LIB=LB1 PRNAME=1
      {print the names SBFILE1 and LB1 on the first line of the
      output}
```



7.7 "E" Commands

7.7.1 ENTER_FILETRANS_REQ (EFTR)

Purpose

Requests a file transfer.

Syntax

```
{ ENTER_FILETRANS_REQ }
{ EFTR }

INFILE = rfile78
OUTFILE = rfile78

[ APPEND = { bool | 0 } ]
[ BINARY = { bool | 0 } ]
[ COMPACT = { 1 | bool } ]
[ START = dec5 ]
[ RESTART = bool ]
[ USER = name12 ]
[ PROJECT = name12 ]
[ BILLING = name12 ]
[ PASSWORD = char12 ]
[ RHOLD = bool ]
[ RCLASS = a ]
[ RPRIORITY = { 0|1|2|3|4|5|6|7 } ]
[ NAME = name8 ]

[ { IMMED } ]
[ { [mm.dd.[yy]yy]. hh.mm } ]
[ WHEN = { } ]
[ { +ddd{W|D|H|M} } ]
[ { when-clause } ]
```



```

[           { W | WEEK }           ]
[           { D | DAY   }           ]
[           { H | HOUR  }           ]
[ EVERY = ( [ dec5 ] { M | MIN } [dec5] ) ]
[           { MONTH    }           ]
[           { WKD      }           ]
[           { NWKD     }           ]

[ WAIT = ( name8 [name8] ... ) ]

[           { CANCEL | NO } ]
[           { IMMED      } ]
[ RRESTART = {           } ]
[           { delay      } ]
[           { delayE     } ]

[ { CREATE } = {NO|NEW|KEEP|REPLACE} ]
[ { CR      } ]

[ CATNOW = bool ]

[ EXPDATE = [yy]yy.mm.dd ]

[ DELETE = bool ]

```

```

-----

[ TYPE = {SL|CU|LM|SM|BIN} ]

[ INDEF = (file-define-parameters) ]

[ OUTDEF = (file-define-parameters) ]

[           { NORMAL    } ]
[           { MONITOR   } ]
[ SHARE = {           } ]
[           { ONEWRITE  } ]
[           { DIR       } ]

[           { READ      } ]
[           { SPREAD    } ]
[ ACCESS = { ALLREAD   } ]
[           { WRITE     } ]
[           { SPWRITE   } ]

[ OUTALC = file-allocation-parameters ]

[ DENSITY = {6250|1600|800|556|200} ]

[ { TRACE } ]
[ {       } = { bool | 0 } ]
[ { TR    } ]

[ BRIEF = { bool | 0 } ]

```



Parameters

INFILE	the origin file or remote file.
OUTFILE	the destination file or remote file.
APPEND	if 1, append in OUTFILE; otherwise OUTFILE is overwritten.
BINARY	if 1, binary format transferred (for SSF files: record headers and control records are transferred).
COMPACT	0 means do not compact; 1 (default value) means compress records for transmission.
START	number of record to start from in INFILE after an incident in a previous file transfer request execution (first record is 1).
RESTART	if 1, restart transfer after interruption of a previous file transfer request execution. The restart point is automatically determined by the file transfer processor.
USER	specifies, if needed, the name of a user, other than the submitter, for whom the connection to the remote site is being requested.
PROJECT/BILLING	when submitting a request on behalf of another billing or project.
PASSWORD	specifies the password of the USER described above.
RHOLD	if 1, request is initially held; the command RELEASE_USER_REQ can be subsequently used to release it.
RCLASS	the class of the request, expressed as a letter in the range A to Z.
RPRIORITY	the priority of the request, expressed as a number in the range 0 to 7. The lower the number, the higher the urgency.
NAME	the symbolic name to be given to the request for further references in dedicated commands. When omitted, the request can only be referred to by its File transfer Occurrence Number (FON in the form Fnnnnnnn) returned as a result of this command.



WHEN	<p>When the request is to be taken into account, expressed as:</p> <p>IMMED now; [mm.dd.[yy]yy]. hh.mm a date; +nnnnn{W D H M} a number of weeks, days, hours or minutes.</p> <p>When omitted, IMMED is assumed.</p> <p>For more information on the when-clause, refer to Appendix E <i>WHEN, EVERY and RRESTART</i>.</p>
EVERY	<p>The frequency and number of times the request is to be submitted, in the form: (frequency, number-of-times) It is two values separated by a comma.</p> <p>For more information, refer to Appendix E <i>WHEN, EVERY and RRESTART</i>.</p>
WAIT	<p>a list up to 8 request names (FON or JON). The current request is not executed until the WAIT requests have completed execution. In the case of the JON, the current request waits until completion of the issued JOB if executed locally, and until completion of the JCL file transfer if executed on a remote site.</p>
RRESTART	<p>Specifies various possibilities for handling requests that were received while the system was off.</p> <p>For further information, consult Appendix E <i>WHEN, EVERY and RRESTART</i>.</p>
CREATE	<p>Creates the outfile, there are 3 possibilities:</p> <p>CREATE = NO: No creation, except for older releases under the OUTDEF parameter.</p> <p>CREATE = REPLACE: The existing output file is deallocated and a new file is created as specified. A warning message is issued.</p> <p>CREATE = KEEP: The existing file is kept. A warning is issued and any creation and allocation parameters are ignored.</p> <p>CREATE = NEW: An error is returned and file transfer aborts.</p>
CATNOW	<p>indicates whether a file level catalog entry is to be created in the catalog; meaningful if CREATE is used.</p>



EXPDATE	file expiration date expressed as: nnn (day in the year), [yy]yy/ddd (year and day), or [yy]yy/mm/dd (year, month and day in the month). Default is no expiration date. Meaningful if CREATE is used.
DELETE	The subfile of the local site will be deleted at the normal end of the transfer. Confirmation of deletion will be asked if the CONFIRM parameter is available in the user profile.
TYPE	In the case of an infile from a foreign machine, and the existence of the CREATE option, this parameter gives the type for the library to be created on the DPS 7000.
INDEF/OUTDEF	further processing options for the input and output files expressed as file-define parameter groups, as detailed in the <i>IOF Terminal User's Reference Manual</i> .
SHARE/ACCESS	sharing and access mode for local file(s) being handled. When omitted, default value for SHARE is NORMAL; default value for ACCESS is READ for INFILE and WRITE for OUTFILE.
OUTALC	It is the allocation group for the out file and supplies supplementary information for the file to be created. It is only used with the CREATE option. For further information, refer to the <i>UFT User Guide</i> .
DENSITY	This is the tape density.
TRACE	If 1, the RFA trace is displayed on the screen.
BRIEF	If 1, only the messages issued at the beginning and at the end of the transfer are displayed. The default is 0.

Rules

- Either INFILE or OUTFILE must refer to a local file
- APPEND is exclusive of START and RESTART
- START and RESTART are mutually exclusive
- BRIEF and TRACE are mutually exclusive
- It is possible to mix file transfer and job requests, but it is not allowed to mix requests (JONs/FONs) and mnames. WAIT=(F18,F25,J38) or WAIT=(A,B,C,D) is allowed, whereas WAIT=(F18,A,J38,B) is not allowed.



- Chaining between requests is limited to 1 level.
 EFTR <...> NAME=REQ1
 EFTR <...> NAME=REQ2 WAIT=REQ1 is allowed
 EFTR <...> WAIT=REQ2 is rejected because REQ2 is already waiting for REQ1.
- OUTALC, CATNOW and EXPDATE are not allowed without CREATE or if CREATE = NO.

Examples

```

EFTR PROJ.LIB1..FLOC
  $SITE:LIB2..FREM
  NAME=FTRANS2 WAIT=FTRANS1 BINARY
                                {transfer SSF subfile to remote file, on completion of
                                request FTRANS1}

EFTR $LYON:LIB..FROM
  LOCLIB..FLOC RESTART WHEN=12.00
                                {at 12.00 restart file transfer which previously failed}

EFTR MYFILE1 $LYON:MYFILE3 NAME=REQ1
EFTR MYFILE2 $LYON:MYFILE4 NAME=REQ2 WAIT=(REQ1)
                                {first transfer MYFILE1 to MYFILE3 on $LYON site;
                                when that is finished, transfer MYFILE2 to
                                MYFILE4}

EFTR $LYON:LIB..FROM
  LOCLIB..FLOC APPEND BINARY
  COMPACT=0 NAME=FTRANS
                                {start the transfer immediately, appending in an
                                existing file}

EFTR MYFILE $LYON:MYFILE2 WAIT=(F23,J67,F45)
                                {Request the transfer of the MYFILE file to $LYON
                                site; the transfer starts when preceding file transfer
                                with FON=F23 and FON=F45, and job with JON=J67
                                are terminated}

EFTR MYFILE $LYON: EVERY=(2H,3);
                                {Transfer MYFILE to site $LYON, 3 times every 2
                                hours.}

EFTR LIB1.MB1 $SITE:LIB2..MB2 BINARY DELETE
                                (Transfer MB1 subfile to remote site and delete it from
                                LIB1 library.)
  
```



7.7.2 ENTER_JOB_REQ (EJR) (EJ)

Purpose

Requests the execution of one or more jobs or submits a jobset request.

Syntax

```
{ ENTER_JOB_REQ }
{ EJR           }
{ EJ           }

[ MEMBERS = star62 ]
[ PROC = name31 ]
[ LIB = lib78 ]
[ CLASS = aa ]
[ PRIORITY = { 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 } ]
[ HOLDOUT = bool ]
[ DEST = [name8.]name8 ]
[ HOLD = bool ]

[ SWITCHES = { ( dec2 [ dec2 ] ... ) | 32 } ]

{ VALUES }
[ {           } = ( value-expressions ) ]
{ VL         }

[ USER = name12 ]
[ PROJECT = name12 ]
[ BILLING = name12 ]
[ REPEAT = bool ]
[ HOST = name8 ]
[ RHOLD = bool ]
[ RCLASS = a ]
[ RPRIORITY = { 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 } ]
[ NAME = name8 ]

[ WHEN = {
           { IMMED }
           { [mm.dd.[yy]yy]. hh.mm } ]
           { +nnnn{W|D|H|M} }
           { when-clause }
        ]
```




```

                                { W | WEEK }
                                { D | DAY }
                                { H | HOUR }
[ EVERY = ( [ dec5 ] { M | MIN } [ dec5 ] ) ]
                                { MONTH }
                                { WKD }
                                { NWKD }

```

```

                                { CANCEL | NO }
                                { IMMED }
[ RRESTART = { } ]
                                { delay }
                                { delayE }

```

```

[ WAIT = ( name8 [ name8 ] ... ) ]
[ JOBLANG = { JCL | GCL } ]
[ DEBUG_GCL = { bool | 0 } ]
[ TRACE_GCL = { bool | 0 } ]
[ PRTFILE = file78 ]

```

```

                                { VALUES2 }
[ { } = ( value-expressions ) ]
                                { VL2 }

```

```

                                { VALUES3 }
[ { } = ( value-expressions ) ]
                                { VL3 }

```

```

                                { VALUES4 }
[ { } = ( value-expressions ) ]
                                { VL4 }

```

```

                                { INFILE }
[ { } = rfile78 ]
                                { INF }

```

```

[ STARTUP = bool ]

```

```

                                { HOLDCOUNT }
[ { } = dec2 ]
                                { HOLDCNT }

```



```
[ DELETE = { NO | YES | FORCE } ]
[ FROM = name12 ]
[ TO = name12 ]
[ BANINF = ( char12 [ char12 [ char12 [ char12 ] ] ] ) ]
[ BANNER = bool ]
[ LIST = { SOURCE | ALL | NO } ]
[ JOR = { NORMAL | NO | ABORT } ]
[ SEV = { 1 | 2 | 3 | 4 | 5 } ]
[ ENVT = name12 ]
[ JOBNAME = name8 ]

{ TRACE }
[ { } = { bool | 0 } ]
{ TR }

[ BRIEF = { bool | 0 } ]
[ EXPVAL = { bool | 0 } ]
```

Parameters

MEMBERS	a star-name denoting the member(s) containing the job(s) to be submitted.
PROC	the name of the GCL procedure to be submitted as an absentee job.
LIB	the library containing the members or the procedure to be submitted. It must not be a temporary library. If the parameter is omitted, the default SL output library (#SLIB), for MEMBERS, or the current binary input search path (#BINLIB), for PROC, is assumed, unless it refers to temporary libraries. If #SLIB or #BINLIB is temporary or unassigned, a system default library is assumed.
CLASS	the class of the submitted jobs, expressed as one or two letters in the range A to RZ.
PRIORITY	the priority of the submitted jobs, expressed as a digit in the range 0 to 7. The lower the priority, the higher the urgency.
HOLDOUT	if 1, jobs' outputs are to be held. The directive RELEASE_OUTPUT can subsequently be used to release them.



DEST	destination of output expressed as [host-name.] station-name. See <i>IOF Terminal User's Reference Manual, Part 1</i> .
HOLD	if 1, jobs are held. The directive RELEASE_JOB can be used in order to make them eligible for execution.
SWITCHES	a list of the switches (expressed as numbers in the range 0 to 31) to be set in the submitted jobs. Value 32 denotes that current values for the switches are passed.
VALUES	a list of one or more values, enclosed in parentheses, to be passed to the submitted job (MEMBERS), or denoting the parameters of the submitted absentee job (PROC). If the values exceed 254 characters, "hidden" parameters VALUES2, VALUES3, and VALUES4 may be used.
USER/PROJECT/BILLING	if submitting jobs on behalf of another user, project, or billing.
REPEAT	if 1, repeat the job if it is cancelled or if there is a system failure or abort.
HOST	the name of the host where the jobs are to execute.
RHOLD	if 1, hold the jobset request. The directive RELEASE_USER_REQ can subsequently be used to release the request.
RCLASS	the jobset request class, expressed as a letter in the range A to Z.
RPRIORITY	the priority of the jobset request, expressed as a number in the range 0 to 7. The lower the number, the higher the urgency.
NAME	the symbolic name to be given to the request for further references in dedicated directives. <i>Names in the form Jnnnnnnn and Fnnnnnnn are not allowed.</i> When omitted, the request may only be referred to by its Jobset Occurrence Number (JON in the form Jnnnnnnn) returned by this directive.



WHEN when the request is to be taken into account, expressed as:

IMMED: now (default)
[mm.dd.[yy]yy].hh.mm: a date
+nnnnn{W|D|H|M}: a number of
weeks, days,
hours, minutes
when-clause: a range of
expressions

The WHEN parameter offers a rich selection of options. For a detailed discussion, to Appendix E . When omitted, IMMED is assumed.

EVERY the interval and number of times the request is to be executed, expressed as:

(interval [number-of-times])

If number-of-times is omitted, an infinite number of times is assumed. Interval can be expressed as a number of weeks, days, hours, or minutes; month (MONTH), working days (WKD), or non-working days (NWKD). The minimum interval is defined by the CONFIG statement DJPCTL (default = 15 minutes, lowest is 1 minute). The maximum interval is 45 days, 12 hours, and 15 minutes (65535 minutes). For a more detailed discussion, refer to the Help texts and Appendix E *When, EVERY and RRESTART*.

RRESTART specifies how an EFTR request should be handled, if the request is not executed because of a system failure or normal shutdown. Valid options are:

IMMED request enqueued
immediately.
CANCEL/NO request is cancelled.
delay delay added to normal
enqueueing time.
delayE elapsed delay, the
maximum delay
allowed for placing
request in queue.

For more detailed information, refer to the Help texts and Appendix E *When, EVERY and RRESTART*.



WAIT	<p>a list of up to 8 request names (FON, JON, or symbolic). The current request will not be executed until the WAIT requests have completed execution. In the case of the JON, the current request waits until completion of the issued job if executed locally, and until completion of the JCL file transfer if executed on a remote site.</p> <p>In the case of a local job request waiting for completion of another job, if the first job terminates with a severity greater than or equal to 3, the waiting request is cancelled. If the severity is less than 3, the request is released.</p>				
JOBLANG	<p>command language used in the job description:</p> <table><tr><td>GCL</td><td>GCOS 7 Command Language</td></tr><tr><td>JCL</td><td>GCOS 7 Job Control Language</td></tr></table> <p>The default value for the parameter JOBLANG is the value set in the system variable #JOBLANG of the user profile. The variable #JOBLANG has no default value. When JOBLANG is specified in EJR and in the statement \$JOB, the value specified in \$JOB overrides the value specified in EJR.</p>	GCL	GCOS 7 Command Language	JCL	GCOS 7 Job Control Language
GCL	GCOS 7 Command Language				
JCL	GCOS 7 Job Control Language				
DEBUG_GCL	<p>if 1, debug GCL procedures. Each time a line is executed, it is displayed with evaluated variables prefixed by procedure name and line number. In batch, a missing procedure name is replaced by the RON. The default value is 0. This parameter may only be used when the job description language is GCL. DEBUG_GCL=1 is equivalent to the directive MODIFY_PROFILE DEBUG.</p>				
TRACE_GCL	<p>if 1, trace all CALL statements executed on the current output device. The default value is 0. This parameter may only be used when the job description language is GCL. TRACE_GCL=1 is equivalent to the directive MODIFY_PROFILE TRACE.</p>				
PRTFILE	<p>the report file, used when the job description language is GCL. If omitted (and if PRTFILE is not specified in the \$OPTIONS statement), the report is directed to the standard system file SYS.OUT.</p> <p>The value specified for PRTFILE in EJR overrides the value specified in \$OPTIONS.</p>				



VALUES2/VALUES3/VALUES4	may be used when the values exceed 254 characters (see parameter VALUES). VALUES2, VALUES3, and VALUES4, though "hidden parameters", appear on the menu directly after VALUES when "EJR+" is entered.
INFILE	the file or remote file containing the jobs to be submitted.
STARTUP	if 0, do not execute the STARTUP sequences in the submitted jobs.
HOLDCOUNT	the submitted jobs are held. They will be released only when the specified number of RELEASE_JOB directives have been issued.
DELETE	if YES, delete the subfile after submission. If FORCE, the subfile is deleted, even if submission is unsuccessful.
FROM/TO	(with INFILE only) select the first and last jobs to be submitted in INFILE. Default values are the first job for FROM, and the last one for TO.
BANINF	up to four 12-character texts to appear on the output banner. The banner is built from bottom to top, with each succeeding line pushing up the previous one(s).
BANNER	if 0, no output banner is generated.
LIST	permits the user to select the type of information to be printed when listing the JCL on the Job Occurrence Report.
SOURCE	the source JCL or GCL, inserted Stream Reader statements, records inserted using a \$SWINPUT statement with the CONSOLE parameter, and error messages are listed.
ALL	the source JCL or GCL, records inserted using a \$SWINPUT statement, and all expansions of JCL sequences which are entered using INVOKE, and project start-up are listed, along with error messages.



NO	only Stream Reader statements, comments, and error messages are listed. If Access Rights have been implemented and the user does not have the READ right over the file(s) containing the information to be printed, then LIST is ignored.
JOR	
NORMAL	a Job Occurrence Report will be produced at the end of job execution.
NO	no Job Occurrence Report will be produced.
ABORT	a Job Occurrence Report will be produced when job execution aborts. For a GCL procedure submitted as an absentee job, the default value is ABORT.
SEV	with PROC: defines the session severity level at which a GCL procedure submitted as an absentee job will be aborted at the end of the current step. The default value is 4. with JOBLANG=GCL: defines the severity level at which the batch job will be aborted at the end of the current step if DEFAULT is the current value for ON_ERROR. SEV is ignored when JOBLANG=JCL.
ENVT	(with PROC only) defines the alternate environment from which a GCL procedure submitted as an absentee job is accessible by the user's project. When omitted, the default environment of the user's project is assumed.
JOBNAME	(with PROC only) the symbolic name to be given to the absentee job. When omitted, the name ABSENTEE is assumed.
TRACE	if 1, the RFA (Remote File Access) trace is displayed on the screen. The default value is 0.



BRIEF	if 1, only the messages issued at the beginning and end of the job execution are displayed when the parameter HOST is used. It is ignored if HOST is not used. The default value is 0.
EXPVAL	if 1, expansion of the values in the JCL of the job is required in the Job Occurrence Report. The default value is 0.

Rules

- BANNER=0 and BANINF are mutually exclusive.
- A jobset is created if:
 - INFILE is used and denotes a remote file
 - one of the following parameters is used: HOST, RHOLD, WHEN, EVERY, WAIT.
- Either MEMBERS, PROC, or INFILE must be specified.
- INFILE is exclusive of LIB, MEMBERS, PROC, and DELETE.
- HOLD and HOLDCOUNT are mutually exclusive.
- STARTUP has meaning only when MEMBERS is specified.
- FROM and TO can only be used when INFILE is used.
- ENVT and JOBNAME are meaningful only when PROC is used.
- SEV is meaningful only when PROC or JOBLANG=GCL is used.
- File transfer and job requests may be mixed, but the mixture of JONs/FONs and symbolic names is not allowed. WAIT=(F18,F25,J38) or WAIT=(A,B,C,D) is allowed; WAIT=(F18,A,J38,B) is not allowed.
- Chaining between requests is limited to 1 level, for example:
EJ MYJOB1 NAME=REQ1
EJ MYJOB2 NAME=REQ2 WAIT=REQ1 is allowed
EJ MYJOB3 WAIT=REQ2 will be rejected because REQ2 already waits on REQ1
- If JOBLANG=GCL, EXPVAL and HOST must not be specified.



Examples

```
EJR PROC=LSV VALUES=MYV:MS/D500
                                {submit absentee job to list a volume}

EJR MYJOBS CLASS=D
                                {submit jobs in output library}

EJR JOBS LIB=PROJ.LIB
                                {submit jobs within specified library}

EJR JOBS WAIT=(F18 F19)
                                {submit after completion of two file transfer requests}

EJR MYJOB1 NAME=REQ1 WAIT=(F12,J67)
                                {submit after completion of file transfer F12 and job
                                J67}

EJR MYJOB2 WAIT=(A,B,C,D)
                                {submit after completion of jobs or file transfers A, B,
                                C, and D}

EJR PROC=MYPROC BINLIB=MYLIB.BIN ENVT=PREP
                                {submit absentee job using alternate environment
                                PREP}

EJR PROC=MYPROC BINLIB=MYLIB.BIN JOBNAME=MYPROC
                                {submit absentee job with symbolic name MYPROC}

EJR JOBS LIB=PROJ.LIB HOST=LYON NAME=PAYROLL WAIT=(FP1 FP2)
                                {execution on remote host, wait for two file transfer
                                requests}

EJR INFILE=$LYON:P1.REMLIB..REMSF WHEN=+20H
                                {submit jobs from remote file in 20 hours time}

EJR PROC=MYPROC VALUES=(A,B,C)
                                {submit absentee job with specified values}

LET A1 A; LET B1 B; LET C1 C
EJR PROC=MYPROC VALUES=#CAT(%A1,' ','%B1',' ','%C1)
                                {same as above with calculated values}

EJR MYJOB EVERY=(2H 3)
                                {submit job MYJOB; execution every two hours and
                                repeated three times}
```



7.7.3 ENTER_LIBTRANS_REQ (ELTR)

Purpose

A complete library or part of a library can be transferred with this command. The asterisk (*) used as a "wild card" is allowed.

Syntax

```
{ ENTER_LIBTRANS_REQ }  
{ ELTR }  
  
INLIB = lib44  
  
SUBFILE = ( star31 [ star31 ]... )  
  
OUTLIB = char44  
  
[ SITE = name4 ]  
  
[ BINARY = { bool | 1 } ]  
  
[ COMPACT = { bool | 1 } ]  
  
[ STARTMB = char31 ]  
  
[ ENDMB = char31 ]  
  
[ CONCISE = { bool | 0 } ]  
  
[ USER = name12 ]  
  
[ PROJECT = name12 ]  
  
[ BILLING = name12 ]  
  
[ PASSWORD = char12 ]  
  
[ RHOLD = bool ]  
  
[ RCLASS = name1 ]  
  
[ RPRIORITY = { 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 } ]  
  
[ NAME = name8 ]
```



```

[           { IMMED } ]
[           { [mm.dd.[yy]yy]. hh.mm } ]
[ WHEN = { } ]
[           { +dddd { W | D | H | M } } ]
[           { when-clause } ]

[           { W | WEEK } ]
[           { D | DAY } ]
[           { H | HOUR } ]
[ EVERY = ( [ dec5 ] { M | MIN } [dec5] ) ]
[           { MONTH } ]
[           { WKD } ]
[           { NWKD } ]

[ WAIT = ( name8 [name8] ..) ]

[           { CANCEL | NO } ]
[           { IMMED } ]
[ RRESTART = { } ]
[           { delay } ]
[           { delayE } ]

[ { CREATE } { } ]
[ { } = { REPLACE | KEEP | NEW | NO } ]
[ { CR } { } ]

[ CATNOW = bool ]

[ EXPDATE = [yy]yy.mm.dd ]

[ DELETE = bool ]

-----

[ INDEF = (file-define-parameters) ]

[ OUTDEF = (file-define-parameters) ]

[ OUTALC = (file-allocation-parameters) ]

[ { TRACE } ]
[ { } = { bool | 0 } ]
[ { TR } ]

[ BRIEF = { bool | 1 } ]

```



Parameters

INLIB	name of the INLIB library. This is the library from which the information is read (transferred).
SUBFILE	name of the library members to be transferred or star convention; up to 10 subfiles can be listed.
OUTLIB	name of the output library. This is the library to which the information is to be written (transferred).
SITE	A remote site can be specified with this parameter.
BINARY	If 1, binary format transferred (for SSF files: record headers and control records are transferred).
COMPACT	0 means do not compact; 1 (default value) means compress records for transmission.
STARTMB	Parameter used when transferring part of a library to specify, inside the star convention (*), the name of the first member to be transferred.
ENDMB	Name of the last library member to transfer; it must be compatible with the star convention (*).
CONCISE	Normally, when transferring part or all of a library, a message is sent to each transfer member. The CONCISE option enables these messages to be skipped, and in this case there is only one message giving the number of transferred members plus the complete number of records transferred.
USER	If needed, specifies the name of a user other than the submitter for whom a connection is to be made towards the remote site. If USER is given, a PASSWORD is also required.
PROJECT	The project identification used by the server on the remote site for file access checking.
BILLING	The BILLING for the specified project. If the parameters PROJECT and BILLING are not specified, the submitter USER must be catalogued on the remote site with the same PROJECT/BILLING affected to him locally when the file transfer is submitted.



PASSWORD	Specifies the password of the user given by the USER parameter. Note, that now, a same user can be known on two sites with different passwords.
RHOLD	If 1, request is initially held; the command RELEASE_USER_REQ can be subsequently used to release it.
RCLASS	The class of the request, expressed as a letter in the range A to Z.
RRIORITY	The priority of the request, expressed as a number in the range 0 to 7. The lower the number, the higher the urgency.
NAME	The symbolic name to be given to the request for further references in dedicated commands. When omitted, the request can only be referred to by its File transfer Occurrence Number (FON in the form Fnnnnnnn) returned as a result of this command.
WHEN	when the request is to be taken into account, expressed as: IMMED now; [mm.dd.[yy]yy]. hh.mm a date; +nnnnn{W D H M} a number of weeks, days, hours or minutes. When omitted, IMMED is assumed. For more information on the when-clause, refer to Appendix E <i>When, EVERY and RRESTART</i> .
EVERY	the frequency and number of times the request is to be submitted, in the form: (frequency, number-of-times) It is two values separated by a comma. For more information, refer to Appendix E <i>When, EVERY and RRESTART</i> .



WAIT	A list up to 8 request names (FON or JON). The current request is not executed until the WAIT requests have completed execution. In the case of the JON, the current request waits until completion of the issued JOB if executed locally, and until completion of the JCL file transfer if executed on a remote site.
RRESTART	Specifies various possibilities for handling requests that were received while the system was off. For further information, consult Appendix E <i>When, EVERY and RRESTART</i> .
CREATE	Creates the outfile, there are 3 possibilities: CREATE = NO: No creation, except for older releases under the OUTDEF parameter. CREATE = REPLACE: The existing output file is deallocated and a new file is created as specified. A warning message is issued. CREATE = KEEP: The existing file is kept. A warning is issued and any creation and allocation parameters are ignored. CREATE = NEW: An error is returned and file transfer aborts.
CATNOW	Indicates whether a file level catalog entry is to be created in the catalog; meaningful if CREATE is used.
EXPDATE	File expiration date expressed as: nnn (day in the year), [yy]yy/ddd (year and day), or [yy]yy/mm/dd (year, month and day in the month). Default is no expiration date. Meaningful if CREATE is used.
DELETE	The subfile(s) of the local site will be deleted at the normal end of the transfer. Confirmation of deletion will be asked if the CONFIRM parameter is available in the user profile.



INDEF	<p>Since the input file is known, this parameter is only useful in certain special cases:</p> <ol style="list-style-type: none">1. When an input file does not have standard labels (tape or cassette only).2. When the logical input definition is ignored by the remote system having the input file (SPS7, SPS9, DPX, etc.) because these files are stored in stream format.3. When the input file is on the local DPS 7000 site in order to change processing parameters such as: NBBUF, BPB.
OUTDEF	<p>This parameter is used as follows:</p> <ol style="list-style-type: none">1. When it has not been preallocated (tape and cartridge only).2. When the output file is a preallocated file on a DPS8. (In this case, the parameters RECSIZE, RECFORM, BLOCKSIZE or CISIZE, and FILEFORM must be used. When the destination file is a GFRC file, the FILEORG is SEQ and the RECSIZE must not be greater than 1272. If the destination file is UFAS SEQ, the FILEORG to be given is RANDOM.3. Under the CREATE option, when the user would like to change the input file definition image used for preallocating the outfile. This is needed so that he can modify only the parameters that are not the same. <p>NOTE: The parameters referred to above are defined in the <i>JCL Reference Manual</i>.</p>
OUTALC	<p>It is the allocation group for the out file and supplies supplementary information for the file to be created. It is only used with the CREATE option. For further information, refer to the <i>UFT User Guide</i>.</p>
TRACE	<p>If 1, the RFA trace is displayed on the screen.</p>
BRIEF	<p>If 1, only the messages issued at the beginning and at the end of the transfer are displayed. The default is 1.</p>



Rules

- The input library (INLIB) must be local and the output library (OUTLIB) must be remote.
- INLIB may not be a temporary input library.
- If CONCISE=1 is used, then SUBFILE=* must be used.
- If STARTMB or ENDMB is used, then SUBFILE=* must be used.
- OUTALC, CATNOW and EXPDATE are not allowed without CREATE or if CREATE = NO.
- WAIT and RRESTART are mutually exclusive.

Examples

```
ELTR mylib a* yourlib LYON;
                                {Transfer all members beginning with a from mylib to
                                yourlib located on site LYON.}

ELTR mylib a* yourlib LYON startmb=ap endmb=as;
                                {This is the same as the preceding example except that
                                the transfer begins with member ap and finishes with
                                member as included.}

ELTR mylib (a,b,c,d) yourlib LYON;
                                {This is the same as the first example, except that only
                                the named members, "a, b, c, and d" are transferred.}

ELTR mylib a* yourlib LYON EVERY=(2H,3);
                                {Transfer all members beginning with a from mylib to
                                yourlib located on site LYON, 3 times every 2 hours.}

ELTR LIB1 MB1* LIB2 SITE DELETE
                                (Transfer all subfiles matching the star convention to a
                                remote site and delete them from the LIB1 library.)
```




7.7.4 EXEC_CLS_COMMAND (EXCLS)

Purpose

Enters a Cartridge Library Server (CLS) command at the GCOS 7 main console.

NOTE:

The syntax and the description of the CLS commands can be found in the Cartridge Tape Library Operator's Guide.

Syntax

```
{ EXEC_CLS_COMMAND }
{                               }
{ EXCLS                         }

  { COMMAND }
  {         } = char255
  { CMD     }

  [ { VALUES }
  [ {         } = char255 ]
  [ { VL     } ]
```

Parameters

COMMAND	The character string is processed as a CLS command submitted by the CLC to the CLS. It must represent a valid CLS command name.
VALUES	The character string is processed as the parameters to be passed to the submitted command.



Rules

- If the command is entered in line mode and more than one parameter is specified in VALUES, the sequence of parameters must be enclosed in single quotes.
- EXECUTE_CLS_COMMAND is rejected during the CLC initialization phase.

Examples

```
EXCLS ENTER           {Enter cartridges through the CAP.}
EXCLS DRAIN 000      {Terminates the operation initiated by ENTER}
EXCLS DISPLAY 'V 001780 DET'
                    {Displays detailed information concerning cartridge
                    001780.}
```



7.7.5 EXEC_CLX_VOLRPT (VOLRPT)

Purpose

Activates the volrpt (volume report) utility on the UNIX server. This utility gives the list of volumes and their physical location in the library. It also provides history and usage statistics on these volumes.

Syntax

```
{ EXEC_CLX_VOLRPT }
{ VOLRPT           }

SERVER = <DPX/20 server name>

RPTFILE = 1 | 0

OPTIONS = <options value>.
```

Parameters

SERVER	The name of the UNIX server on which the report is to run.
RPTFILE	If set to 1, the report is stored in the file /home/ACSSS/log/volrpt.log on the Unix server, if set to 0, the report is displayed on the console
OPTIONS	Refer to ACSLS Administrator's guide.



Rules

- The OPEN 7 network name must be declared in the */home/ACSSS/rhosts* file on the library server. It must contain the following line in which the DPS 7000 name must be in lower case:

```
<dps7000_name> clx
```

- Permission bits for the file */home/ACSSS/rhosts* must be `-rw-r--r-`
- These conditions must have been fulfilled at the installation of CLX.

Example

```
VOLRPT  RPTFILE=1  
        SERVER='wolf'
```

The volume report is stored in the UNIX server file */home/ACSSS/log/volrpt.log*.



7.7.6 EXECUTE_DIRECTIVE (EXDIR)

Purpose

Executes the commands (in GCL format) that are given in the COMMAND parameter. Main Operator commands may be submitted this way if the ABSENTEE Job is running under a project that has the "MAIN" attribute. When it is not the case, only IOF directives as described in the *IOF Terminal User's Reference Manual* may be executed. The image of the submitted commands and the error-messages (if any) is provided in the JOR.

Syntax

```
-----  
{ EXECUTE_DIRECTIVE }  
{ EXDIR               }
```

COMMAND = char255

Parameters

COMMAND the string of commands to be executed (commands are separated by a semi-colon).

Rules

- The GCL directives 'MWINLIB BIN' and 'BYE' may not be executed via EXDIR.
- A job cannot suspend itself through HJ via EXDIR unless the ENDSTEP parameter is used in the HJ command.

Example:

```
EXDIR 'MODIFY_LOAD MULTI=100 CLASS=P ; MDHW TWIN=0';  
          {set multiprogramming level for P class to 100 and  
          reinsert all available resources in the CSS}
```



7.8 "F" and "G" Commands

7.8.1 FORCE_JOB (FJ)

Purpose

Positions a job at the front of the queue for job scheduling, ignoring any restriction due to the job class or the fact that the job is in the HOLD state. The named job starts as soon as it can, regardless of whether its class is started or not. The job starts even if its class or the system becomes overloaded or even if it is in the HOLD state. A forced job can be held before it is started by using the HOLD_JOB command, in which case it is no longer considered as being forced.

Syntax

```
{ FORCE_JOB }  
{ FJ      }
```

RON = Xnnnn

```
[ { SYSTEM }      ]  
[ {           } = name4 ]  
[ { SYS     }      ]
```

Parameters

RON	the Run Occurrence Number (RON) of the job to be forced.
SYSTEM	This parameter indicates the system where the command will be executed.



Rules

- FORCE_JOB cannot apply to service jobs or to jobs that are in the OUT state.
- Only one forced job may be present in the IN state at any one time. If there is a forced job waiting to start and another FORCE_JOB command is issued for another job, the first job returns to its original position in the scheduling queue and is no longer considered as being forced.
- A forced job does not start if the maximum J number is exceeded.

Example

FJ X1234 {force job with RON X1234}



7.8.2 FORCE_OUTPUT (FO)

Purpose

Forces an output. The output is started as soon as an output device of the correct type becomes available. If a forced output was held, it is released.

Syntax

```
{ FORCE_OUTPUT }  
{ FO           }  
  
    OUTPUT = output14  
  
    [ STRONG = bool ]
```

Parameters

OUTPUT	the name of the output to be forced. It may take one or the two following forms: Xnnnn:nnnn Xnnnn:name
STRONG	if 1, the output is forced even if the output is in the READY state.

Rules

Only one output may be forced on a station at any one time. Any subsequent FORCE_OUTPUT command on a given station is rejected, until the forced output is held, or until it no longer exists in the queue of a station. The operator must either wait until the previous forced output is completed, or else cancel it, or hold it, or transfer it to another station before trying again.

Examples

FO X1234:2	{force the named output}
FO X1234:REPORT	{force the output of job X1234 named REPORT}
FO X45:3 STRONG	{force the specified output even if in the READY state}



7.8.3 FORCE_SERVER_TIMEOUT (FSVRTO)

Purpose

If a SERVER aborts without notifying CMSR, the SERVER will be forcefully terminated when CMSR has not heard from it for a certain period of time. This command enables the operator to reduce this waiting time.

The operator must use this command with caution, because it can terminate the SERVICE on the current MEMBER.

Syntax

```
{ FORCE_SERVER_TIMEOUT }
{ FSVRTO

      { SERVICE }
      {          } = name8
      { SRV      }
}
```

Parameters

SERVICE Name of the SERVICE

Rules

- The CMSR of the MEMBER on which this command is entered must be running.
- The SERVICE must be the COMPLEX description.
- The CMSR of the current MEMBER must be waiting for a notification of this SERVICE.

Examples

```
FSVRTO  SRV=TDS1  {Forces the timeout for the SERVICE named TDS1 on
                  the current MEMBER.}
```



7.8.4 FORCE_USER_REQUEST (FUR)

Purpose

Forces a request. The WHEN and/or WAIT conditions (if any) are ignored and the request is put in queue in the WAIT state with the highest priority.

Syntax

```
{ FORCE_USER_REQ }  
{ FUR           }  
  
    REQUEST=name8
```

Parameters

REQUEST name of the request to be forced. It may take one of the following three forms:

nnnnn: a Jobset Occurrence Number (JON),
Fnnnnnn: a File transfer Occurrence Number (FON),
name8: the symbolic name of a request as specified by the NAME parameter of the ENTER_JOB_REQ or ENTER_FILETRANS_REQ commands.

RONs and FONs are provided as results of the ENTER_JOB_REQ or ENTER_FILETRANS_REQ commands.



Rules

- If a second FORCE_USER_REQ command is issued while the first forced request is not yet executed, the second request replaces the first one which is again given its initial priority.
- FORCE_USER_REQ (J41 J48) does not work, because the command only takes one job at a time.

Examples

FUR F1234	{force a file transfer request}
FUR J456	{force a jobset request}
FUR MYREQ	{force a named request}



7.8.5 GET_CLX_FILE (GTCLXF)

Purpose

Transfers a UNIX file to a local GCOS 7 file. The UNIX file can be located on the DPX/20 server or on OPEN 7.

For example, this command can be used to transfer SSI or ACSLS log files to GCOS 7.

Syntax

```
{ GET_CLX_FILE }  
{ GTCLXF }
```

```
SERVER = <server name> ! OPEN7 = 1  
REMOTE_FILE = <dpx/20 or OPEN7 file name>  
GCOS_LIB = <GCOS 7-library-name>  
GCOS_SUBFILE = <GCOS 7-subfile-name>
```

Parameters

SERVER	The name of the UNIX server that contains the file to be transferred. It is mutually exclusive with the parameter OPEN 7.
OPEN7	Must be set to 1 if the file to be transferred is a local OPEN 7 file. If set to 1, the SERVER parameter must contain blank characters.
REMOTE_FILE	Name of the file to be transferred.
GCOS_LIB	Name of the GCOS 7 library into which the file is transferred.
GCOS_SUBFILE	Name of the GCOS 7 subfile into which the file is transferred.



Rules

- Only files readable by the user <clx> can be transferred: permission bits must be set for the user <clx>.
- UNIX server files can be transferred if the OPEN 7 network name is declared in the */home/ACSSS/rhosts* file on the library server.
- This condition must have been fulfilled at the installation of CLX.

Example

```
GTCLXF
  GCOS_LIB= lciu.cmd.sllib
  GCOS_SUBFILE=es_volrpt_wolf
  REMOTE_FILE='/home/ACSSS/log/volrpt.log'
  SERVER='wolf'
```

The content of the */home/ACSSS/log/volrpt.log* file of the library server <wolf> is transferred into the subfile *es_volrpt_wolf* in the GCOS 7 library *lcIU.cmd.sllib*.



8. Description of the Operator Commands (Part 2)

Chapter 8 is the continuation of the detailed description of the GCL commands listed alphabetically.

8.1 "H" and "I" Commands

8.1.1 HOLD_GCL_CACHE (HGC)

Purpose

This command stops the operation of the GCL cache function and makes swappable the segments used by the cache function. The segments in question are those which are RESIDENT when the cache function is operational.

Syntax

```
{ HOLD_GCL_CACHE }  
{ HGC }
```

Parameters

None.

Rules

None.



8.1.2 HOLD_JOB (HJ)

Purpose

Prevents execution of one or more jobs in the scheduling queue and puts executing jobs in the SUSPENDED state at the end of their current step if the parameter ENDSTEP is set to 1. A job may be held by its submitter, its USER and by main or station operators. The effect of the HOLD_JOB directive may be reversed by means of the RELEASE_JOB directive.

Syntax

```
{HOLD_JOB}
{HJ      }

[ RONS = (Xdddd [Xdddd] ...) ]

[ {SELECT} = {jclass} [ {usernm} [ {projnm} ] ] ]
[ {SEL   } = { * } [ { * } [ { * } ] ] ]

[ ENDSTEP = { 0 | 1 } ]
[          {      } ]

[ STATION = st-name ]

[ {SYSTEM} = name4 ]
[ {SYS   } =      ]
```

Parameters

RONs	a list of up to 16 Run Occurrence Numbers (RONs) identifying the jobs to be held or suspended.
SELECT	applies the command to the jobs that belong to the specified combination of class/user-name/project. (* is all jobs accessible by terminal-user).
ENDSTEP	if 1, executing jobs are suspended at the end of their current steps. If 0 (default value) or not specified, executing jobs are immediately suspended. This parameter is ignored for jobs in the scheduling queue.



STATION	name of station where job must be held.
SYSTEM	This parameter indicates the system where the command is executed.

Rules

- Service jobs may be neither held nor suspended.
- Jobs in the OUT state may not be held nor suspended.
- When used on a forced job, the job is considered as being held and no longer forced.
- This command does not apply to the user jobs that use the GAC facility (General Access Control).
- RONS and SELECT are mutually exclusive.
- When no RONS are specified, all jobs matching the SELECT combination, which are not executing, are held. If ENDSTEP is set to 1, all executing jobs matching the SELECT combination are also suspended.
- A job cannot suspend itself through HJ via EXDIR unless the ENDSTEP parameter is used in the HJ command.

Examples

HJ	{hold all jobs not executing}
HJ ENDSTEP	{hold all jobs not executing and suspend the executing ones at the end of their current steps}
HJ (X123 X456)	{hold or suspend the two named jobs}
HJ X1234 ENDSTEP	{suspend job X1234 at the end of its current step}
HJ SL=AA	{hold or suspend all AA class jobs}
HJ SL= */ */ MYPROJ	{hold or suspend all jobs from project MYPROJ}
HJ X123 ENDSTEP	{hold job X123 at the end of its current step if it is executing, or hold job X123 if it is in scheduling queue}



8.1.3 HOLD_OUTPUT (HO)

Purpose

Prevents the printing of outputs in queue, or suspends the printing of outputs being printed. The effect of HOLD_OUTPUT may be reversed by means of the RELEASE_OUTPUT command.



IMPORTANT:

This command applies only to the outputs on which the submitter has access rights

Syntax

```
{ HOLD_OUTPUT }
{ HO

[ (output14 [ output14 ] ...) } ]
[ OUTPUT = { output14-output14 } ]
[ { * | ALL } ]

[ SELCLASS = a [-a] ]

[ STRONG = { bool | 0 } ]

[ { STATION } { * } ]
[ { } = { ALL } ]
[ { ST } { name8 } ]

[ { SELECT } { user/project } ]
[ { } = { */project } ]
[ { } { */* } ]
[ { SEL } { * } ]

[ { JOBNAME } ]
[ { } = name8 ]
[ { JNM } ]

[ { SYSTEM } ]
[ { } = name4 ]
[ { SYS } ]
```




STATION	restricts the scope of the command to outputs attached to the station specified. When omitted or when * or ALL is specified, there is no station restriction.
SELECT	restricts the scope of the command to outputs belonging to a specified user and/or project. Not meaningful when OUTPUT specifies a list of outputs (format 1). By default, only the outputs belonging to the command submitter are selected.
JOBNAME	restricts the scope of the command to the outputs of jobs with the name specified. If JOBNAME is specified, the default value for OUTPUT is *.
SYSTEM	indicates the system on which the command is to be executed.

Rules

- SELCLASS is ignored for names of specific outputs (Xnnnn:nnnn or Xnnnn:name). When using SELCLASS=A-Z, only currently existing outputs are concerned.
- If several outputs of the same job are given the same name, HO Xnnnn:name only affects the first output with that name encountered.
- If JOBNAME is used, the only allowable values for OUTPUT are "*", "ALL", "Xnnnn-Xnnnn" or unspecified.

Examples

HO X234	{hold all outputs of job X234}
HO X234 P	{hold all outputs of job X234 that are of class P}
HO X234 C-Q	{hold all outputs of job X234 that are of classes C to Q}
HO X234 :1	{hold specified outputs of job X234}
HO X234 :RESULT	{hold specified outputs of job X234}
HO (X234:2 X234:PH6)	{hold specified outputs of job X234}
HO X234 STRONG	{hold all outputs of X234, even if being printed}
HO X234 A-Z	{hold all currently known outputs of job X234}



HO X123-X2167	{hold all outputs, belonging to the command submitter, of the jobs whose RON is in the range X123 to X2167}
HO * SELCLASS=Z	{hold all class Z outputs, belonging to the command submitter, of all jobs}
HO * SELECT=U1/P1	{hold all the outputs belonging to user U1 of project P1}
HO * STATION=BC10	{hold all outputs, belonging to the command submitter, that are attached to the station BC10}
HO X123-X2167 SELCLASS=A ST=BC10 SEL=U1/P1	{hold all class A outputs of jobs in the range X123 to X2167 that are attached to station BC10 and belong to user U1 of project P1}
HO JOBNAME=JOB1	{hold all outputs, belonging to the command submitter, of the jobs named JOB1}



8.1.4 HOLD_TERMINAL_OUTPUT (HTO)

Purpose

Prevents the printing of outputs by the Generalized Terminal Writer, or suspends the printing of outputs being printed by the Generalized Terminal Writer. The effect of HOLD_TERMINAL_OUTPUT may be reversed by means of the RELEASE_TERMINAL_OUTPUT command. Refer to the *Generalized Terminal Writer User Guide* for a complete description of the Generalized Terminal Writer.

Syntax

```
{ HOLD_TERMINAL_OUTPUT }
{ HTO                    }

      OUTPUT = (Rnnnn [Rnnnn] ... )

      [STRONG = { bool | 0 } ]
```

Parameters

OUTPUT	a list of the names of up to 16 outputs to be held. Each output name has the form Rnnnn and is provided as the result of the DPRINT command (with parameter TWDEST).
STRONG	if 1, hold even if outputs are being printed (in that case, the outputs become "frozen").

Rules

None.

Examples

```
HTO R1234           {hold output with name R1234}
HTO (R1234 R1235)  {hold two outputs}
HTO R1234 STRONG   {hold even if being printed}
```



8.1.5 HOLD_USER_REQ (HUR)

Purpose

Prevents file transfer or jobset requests submitted by the terminal user from being executed. The effect of the HOLD_USER_REQ command is reversed by means of the RELEASE_USER_REQ (RUR) command.

Syntax

```
{ HOLD_USER_REQ }  
{ HUR           }
```

```
REQUEST = (name8 [name8] ...)
```

Parameters

REQUEST a list of up to 16 names of requests to be cancelled.
Each name may take one of the following three forms:

- Jnnnnnnn a Jobset Occurrence No. (JON)
- Fnnnnnnn a File transfer Occurrence No. (FON),
- name8 the symbolic name of a request, as specified by the NAME parameter of the ENTER_JOB_REQ or ENTER_FILE -TRANS_REQ commands.

JONs and FONs are provided for the ENTER_JOB_REQ and ENTER_FILETRANS_REQ commands. They are unique identifiers for these requests.

Rules

None.

Examples

```
HUR F128 {hold a file transfer request}
```

```
HUR J44 {hold a jobset request}
```

```
HUR (F3 HQR J2) {hold three requests}
```



8.1.6 INIT_SYSTEM_TRACE (ISYSTR)

Purpose

Restores system trace domain attributes and connections to initial state values. Initial state values are defined as follows:

- Before the first system trace collector execution, they are the values defined at GCOS 7 initialization.
- During system trace collector execution, they are the values defined by system trace collector execution commands.

NOTE:

This command is to be used only under the direction of a Bull support engineer.

Syntax

```
{ INIT_SYSTEM_TRACE }
{ ISYSTR              }

[ { DOMAIN }      { ALL } ]
[ { DOM       } = {      } ]
[ { TDOM      }   { name64 } ]
```

Parameters

DOMAIN It is the name of a system trace domain, or if ALL (the default), all domains.

Rules

None.

Example

ISYSTR (Restore all trace domains to initial state status.)



8.2 "L" Commands

8.2.1 LIST_DUMP (LSDP)

Purpose

Lists all binary dumps for the selected RON (Run Occurrence Number), or for all dumps of SYS.SPDUMP. For further information, refer to Appendix B *Calling System Utilities*.

Syntax

```
{ LIST_DUMP }  
{  
{ LSDP }  
}
```

```
[ RON = { Xnnnn | ALL | * } ]
```

Parameters

RON the Run Occurrence Number of the job whose dumps are to be listed. If ALL (the default) or "*" is specified, all dumps of SYS.SPDUMP are listed.

The following message is displayed for each dump that fits the selection criteria:

```
DP01 <dumpid> PGID=<jjjj> <lmname> <date>
```

where "dumpid" is the dump identification (RON, dynamic step number, and number of the dump in the step); "jjjj" is the process-group identifier; "lmname" is the name of the load module (first 16 characters only); and "date" is the date and time of dump creation.

Constraints

None.

Examples

```
LSDP X2345 {list all dumps of job X2345}
```

```
LSDP * {list all dumps of SYS.SPDUMP}
```




8.2.2 LIST_FILTER (LSFLT)

Purpose

List a single filter, a set of filters, or all filters that belong to the operator.

Syntax

```
{ LIST_FILTER }
{ LSFLT      }

      { FILTERSET } { *      }
      { FLTST     } = { fltst }

[ { FILTER }      ]
[ {           } =flt ]
[ { FLT        }  ]

[ { SYSTEM }      ]
[ {           } = name4 ]
[ { SYS     }      ]
```

Parameters

FILTERSET	When used alone (without FILTER), specifies filter set to be listed (* means all operator defined filter sets). When used with FILTER, specifies the filter set to which the filter to be listed belongs.
FILTER	the name of the filter to be listed. When omitted, all filters that belong to the set of filters named by FILTERSET are listed.
SYSTEM	The system where the command is executed.



Rules

When FILTERSET is *, FILTER cannot be specified.

Examples

LSFLT A1 A101	{list filter A101 within filter set A1}
FSFLT A1	{list all filters of set A1}
LSFLT *	{list all filters and filter sets belonging to the operator}



8.3 "M" Commands

8.3.1 MAIL

Purpose

Controls the flow of messages from the user's mailbox to his terminal.

Syntax

MAIL

```
[          { DISPLAY    } ]
[          { ON         } ]
[ ACTION = { OFF        } ]
[          { CLEAR     } ]
[          {           } ]

[          { ALL        } ]
[ OPTION = { Xnnnn     } ]
[          { name8 [ /Xnnnn ] } ]

[          { 0 } ]
[ SYNC = { } ]
[          { 1 } ]
```

Parameters

ACTION Indicates what control is to be applied as follows:

- DISPLAY,D Print the contents of the mailbox.
- ON Incoming messages are printed as they arrive.
- OFF Incoming messages are held in the mailbox.
- CLEAR,C Delete messages in the mailbox as indicated by OPTION.



OPTION

If ACTION is CLEAR, indicates which messages are to be deleted as follows:

- ALL All messages are deleted (default value).
- Xnnnn: Provides the Run Occurrence Number of the originator of the messages to be deleted.
- name8: It is the name of the site from which all originated messages are to be deleted.
- name8/Xnnnn Provides the site and the Run Occurrence of the originator of the messages to be deleted.

SYNC

In the synchronous response mode, the operator waits for the response before entering anything else at the keyboard. In asynchronous mode, the operator has control right away without waiting for the response. The SYNC parameter is used to change the response mode:

SYNC = 1 gives Synchronous responses

SYNC = 0 gives Asynchronous responses

There is no default. Each new connection uses the mode previously set. At GCOS 7 initialization, the mode is set to synchronous.



Rules

OPTION may be specified only if ACTION is CLEAR or C.

Examples

MAIL	{print contents of mailbox}
MAIL ON	{accept mail}
MAIL OFF	{do not accept mail}
MAIL CLEAR	{clear the mailbox}
MAIL CLEAR X123	{delete messages originating from job X123}
MAIL CLEAR LYON/X123	{delete messages originating from remote site job X123}
MAIL SYNC	{ask for a synchronous display of commands responses}



8.3.2 MODIFY_CLC (MDCLC)

Purpose

Modifies the Cartridge Library Client (CLC) parameters.

Syntax

```
{ MODIFY_CLC }  
{ MDCLC      }  
  
TRACE = { bool | 0 }  
  
CLEAR = { bool | 0 }
```

Parameters

TRACE	If 1, the contents of the protocol messages exchanged between the CLC and the Cartridge Library Server (CLS) is displayed on the console.
CLEAR	If 1, the content of the file SYS.CLC is cleared. If 0, the content is not modified. The default value is 0. CLEAR=1 deletes all members of the library whose names begin with XMIT (explicit file transfer).

Rules

MODIFY_CLC is accepted even during the CLC initialization. The TRACE parameter can be used either for debugging purposes or during the installation phase.

Examples

MDCLC TRACE	{Enter trace mode.}
MDCLC	{Exit trace mode, i.e. return to normal mode.}
MDCLC NTRACE	{Exit trace mode, i.e. return to normal mode.}
MDCLC CLEAR	{Clear the content of the SYS.CLC library without modifying the current trace mode.}



8.3.3 MODIFY_CLX (MDCLX)

Purpose

Turns the CLX trace on or off. When the CLX trace is active, all messages exchanged between the CLX and the ACSLS are displayed on the operator terminal from which CLX was started.

The trace displays requests sent to the ACSLS by CLX, and also displays the acknowledgements and responses returned by the ACSLS.

Syntax

```
{ MODIFY_CLX }  
{ MDCLX      }  
  
TRACE = { 0 | 1 }
```

Parameters

TRACE If 1 (default), starts CLX trace mode
 If 0, stops CLX trace mode.

Rules

The command MDCLX TRACE can be used to verify that everything is functioning correctly during the installation phase, in order to determine the cause of a mount/dismount problem, or to understand a problem related to the exchange protocol between the CLX and the ACSLM.

Example

Trace of a FILSAVE utility. It begins with a request to mount the volume 001863 on the device CT17 and terminates with the request to dismount this volume.

```
S: MDCLX TRACE  
14.38 V2.1 TU14 CLX COMMAND IN PROGRESS  
14.39 X1865 IN FILSAVE USER=OPERATOR CLASS=P SPR=6 STATION=BCA9  
14.39 X1865 STARTED FILSAVE OPERATOR P  
* 14.39 CT17 MOUNT 001863 LIB FOR X1865  
LOCK DRIVE SEQ_NO= 76 LOCKID= 0 USERID=BCA9 DRIVEID=0,0,1,2 WAIT=0 COUNT= 1  
STATUS= 0  
LOCK DRIVE SEQ_NO= 76 REP_TYPE=FINAL STATUS= 0 LOCKID=12136 DRIVEID=0,0,1,2  
STATUS_2= 0 TYPE=17 IDENT=
```



```
LOCK VOLUME SEQ_NO= 77 LOCKID=12136 USERID=BCA9 VOLID=001863 WAIT=0 COUNT= 1
STATUS= 0
LOCK VOLUME SEQ_NO= 77 REP_TYPE=FINAL STATUS= 0 LOCKID=12136 VOLID=001863
STATUS_2= 0 TYPE=17 IDENT=
MOUNT SEQ_NO= 78 LOCKID=12136 VOLID=001863 DRIVEID=0,0,1,2 READONLY=0 STATUS= 0
--- SEQ_NO= 78 REP_TYPE=ACK REQID= 9002
JB08 X1865.1 STEP H_UTILITY XPR=8 PGID=22
MOUNT SEQ_NO= 78 REP_TYPE=FINAL STATUS= 0 VOLID=001863 DRIVEID=0,0,1,2
TU11 V2.1 CT17 IN LIBRARY : 001863 MOUNTED
* DW10 CT17 DISMOUNT 001863 S
DISMOUNT SEQ_NO= 79 LOCKID=12136 VOLID=001863 DRIVEID=0,0,1,2 FORCE=0 STATUS= 0
14.40 X1865.1 COMPLETED FILSAVE OPERATOR P
--- SEQ_NO= 79 REP_TYPE=ACK REQID= 9004
DISMOUNT SEQ_NO= 79 REP_TYPE=FINAL STATUS= 0 VOLID=001863 DRIVEID=0,0,1,2
TU12 V2.1 CT17 IN LIBRARY : 001863 DISMOUNTED
UNLOCK VOLUME SEQ_NO= 80 LOCKID=12136 VOLID=001863 COUNT= 1 STATUS= 0 UNLOCK
VOLUME SEQ_NO= 80 REP_TYPE=FINAL STATUS= 0 COUNT= 1 VOLID=001863 STATUS_2= 0
TYPE=17 IDENT=
UNLOCK DRIVE SEQ_NO= 81 LOCKID=12136 DRIVEID=0012 COUNT= 1 STATUS= 0
UNLOCK DRIVE SEQ_NO= 81 REP_TYPE=FINAL STATUS= 0 COUNT= 1 DRIVEID=0,0,1,2
STATUS_2= 0 TYPE=17 IDENT=
--- SEQ_NO= 81 REP_TYPE=ACK REQID= 9006
```

The GCOS 7 mount request causes the following ACSLS requests:

- lock drive,
- lock volume,
- mount.

The GCOS 7 dismount request causes the following ACSLS requests:

- dismount,
- unlock volume,
- unlock drive.

The request <lock drive> is sent with the parameter LOCKID set to 0, where 0 means no LOCKID, this ask ACSLS for a LOCKID in return. The returned LOCKID value is used by the CLX for the commands <lock volume>, <mount>, <dismount>, <unlock drive> and <unlock volume>.



8.3.4 MODIFY_CONFIGURATION (MDC) - STATION OPERATOR

Purpose

Modifies the default output priority, the default output media, and the quota of lines for an output class.

Syntax

```
{ MODIFY_CONFIGURATION }
{ MDC

    { OUTCLASS }
    {           } = (a [a] ...)
    { OC       }

    [{ PRIORITY }
    [{           } = { 0|1|2|3|4|5|6|7 } ]
    [{ PRTY     }

    [{ MEDIA    }
    [{           } = name6 ]
    [{ MD       }

    [ LINES = { * | char7 } ]
```

Parameters

OUTCLASS	a list of up to 10 output classes whose characteristics are to be modified. Each class is expressed as a letter (A through Z).
PRIORITY	the new default output priority for the output class, expressed as a digit in the range 0 through 7.
MEDIA	the new default output media for the output class.
LINES	Specifies the maximum number of lines for an output in the given class. If * or 9999999 is specified, there is no limit. If this parameter is omitted, the current value is kept.

The value of LINES must be greater than 0. Any job that creates outputs for the class concerned with more than the number of lines permitted by the LINES parameter is aborted with return code ERLMOV.



The limit set by LINES does not concern dumps.

Any buffering done by the OUTPUT-WRITER is not taken into account for comparison with the value of the LINES parameter.

Rules

At least one of PRIORITY, MEDIA, or LINES must be specified.

Examples

```
MDC (A, B) 3          {default priority for classes A and B is 3}
MDC Z MD=I12B45      {default media for class Z is I12B45}
MDC Z 3 I12B45       {set both the default priority and media}
MDC OC=B LINES=100000
                    {outputs for class B have a maximum size of 100,000
                    lines}
MDC OC=Z LINES=*     {there is no limit on the number of lines for outputs of
                    class Z}
```



8.3.5 MODIFY_CONFIGURATION (MDC) - MAIN OPERATOR

Purpose

Modifies elements of the configuration: output classes, pre-initialized load modules, resident system functions, dispatching priorities, memory dimensions.

Syntax

```
{ MODIFY_CONFIGURATION }
{ MDC

    [ { OUTCLASS }
    [ {           } = (a [a] ...) ]
    [ { OC       }
    ]

    [ { PRIORITY }
    [ {           } = { 0|1|2|3|4|5|6|7 } ]
    [ { PRTY     }
    ]

    [ { MEDIA }
    [ {         } = name6 ]
    [ { MD     }
    ]

    [ { STATION }
    [ {         } = name8 ]
    [ { ST      }
    ]

    [PLM = [lib-name..] lm-name [:volume :dvc]]
    [CLM = [lib-name..] lm-name [:volume :dvc]]

    [ LINES = { * | char7 } ]
```

Parameters

OUTCLASS	a list of up to 10 output classes whose characteristics are to be modified. Each class is expressed as a letter (A to Z).
PRIORITY	(with OUTCLASS) the new default output priority for the output class, expressed as a digit in the range 0 to 7.
MEDIA	(with OUTCLASS) the new default output media for the output class.



STATION	(with OUTCLASS) the name of the (unattended) station for which output class characteristics are to be changed. When omitted, this applies to the MAIN station.
PLM/CLM	the identification of a load module that is to be pre-initialized (PLM) or whose pre-initialization is to be cancelled (CLM). When the library-name is omitted, SYS.HLMLIB is assumed.
LINES	<p>Specifies the maximum number of lines for an output in the given class. If * or 9999999 is specified, there is no limit. If this parameter is omitted, the current value is kept.</p> <p>The value of LINES must be greater than 0. Any job that creates outputs for the class concerned with more than the number of lines permitted by the LINES parameter is aborted with return code ERLMOV.</p> <p>The limit set by LINES does not concern dumps.</p> <p>Any buffering done by the OUTPUT-WRITER is not taken into account for comparison with the value of the LINES parameter.</p>

Rules

- If STATION, PRIORITY, MEDIA, or LINES is used, then OUTCLASS must be used.
- If OUTCLASS is used, at least one of PRIORITY, MEDIA, or LINES must be used.

Examples

```
MDC OC=Z 3 12B45    {modify characteristics of output class Z}
MDC OC=(A,B) 3 ST=STN
                    {modify output classes A and B default priority for
                    station STN}
MDC OC=B LINES=100000
                    {outputs for class B have a maximum size of 100,000
                    lines}
MDC OC=Z LINES=*   {there is no limit on the number of lines for outputs of
                    class Z}
```



8.3.6 MODIFY_DEVICE_TRACE (MDDVTR)

Purpose

Modifies the number of events on a device (or devices) after which a warning message (AV16) is issued. This threshold can be defined for a device, a range of devices or devices of a certain class.

Syntax

```
{ MODIFY_DEVICE_TRACE }
{ MDDVTR                }

[ { DEVICE } { * } ]
[ {         } = { dvtyp [dvnm] } ]
[ { DV      } { dvc      } ]

[ { THRESHOLD } { 0 } ]
[ {           } = {     } ]
[ { EVENTS    } { dec3 } ]
```

Parameters

DEVICE the device(s) for which the threshold is defined, expressed as:

```
*      all devices,
dvnm   a specified device,
dvtyp  all devices of a type,
dvc    all devices of a class.
```

When omitted, all devices (*) are assumed.

THRESHOLD the value of the threshold. When 0 or omitted, the warning message is never issued. When not zero, the warning message is issued each time the number of logged events becomes a multiple of the threshold.



Rules

None.

Examples

MDDVTR	{threshold becomes 0 for all devices}
MDDVTR MS 100	{threshold is 100 for all disk units}
MDDVTR MS/B10	{threshold is 0 for all MSU0555 disk units}
MDDVTR MS03 20	{threshold is 20 for disk unit MS03}

NOTE:

System initialization does not change the threshold value or the number of logged events. To reset the number of logged events to zero, use the PRLOG utility and the MODIFY_ERROR_LOG (MDERLOG) command.



8.3.7 MODIFY_DIMENSION (MDDIM) - BASIC ARM

Purpose

Change the MPL of the dimension (BASIC Level only). See the *ARM User's Guide* for full details.

Syntax

```
{ MODIFY_DIMENSION }
{ MDDIM              }

      DIM = name8

      MPL = dec3

      [ { SYSTEM }           ]
      [ {           } = name4 ]
      [ { SYS      }           ]
```

Parameters

DIM	The name of the dimension.
MPL	The new multiprogramming level.
SYSTEM	This parameter indicates the system where the command is executed.

Rules

User must be in a MAIN operator project.

Example

```
MDDIM IOF7 MPL=(10)
                        {the multiprogramming level of IOF7 dimension is set
                        to 10}
```



8.3.8 MODIFY_DIMENSION (MDDIM) - FULL ARM

Purpose

Changes the attributes and/or updates the XL (Execution Level) range of the dimension. See the *ARM User's Guide* for full details.

Syntax

```
{ MODIFY_DIMENSION }
{ MDDIM              }

DIM = name8

[ MPL = (dec3 [,dec3] ) ]

[ XL = (dec3 [,dec3] ) ]

[ RELWGHT = dec3 ]

[ BWGHT = dec3 ]

[ { INDIVIDUAL }           ]
[ {                } = bool ]
[ { ICA          }           ]

[ HOLD = bool ]

[ { PRIORITY_ATTRIBUTE }           ]
[ { PRYATTR           } = bool ]
[ { PA                }           ]

[ { SYSTEM }           ]
[ {                } = name4 ]
[ { SYS      }           ]
```

Parameters

DIM	The name of the dimension, for example: POOL3, TD55, etc.
MPL	Multiprogramming level: a set of two decimal values, the first one being the minimum and the second being the maximum. If minimum = maximum, only one value is specified. MPL may range from 0 to 999. ARM periodically computes an optimized MPL value between those two limits.



XL	Definition of an XL range through a list of two values (XL=(n1,n2)). If minimum = maximum, only one value is specified. XL may range from 0 to 100. For the SYS dimension, XL max is equal to 100 and cannot be changed.
RELWGHT	Relative weight. Range from 1 to 100. Indicates the Relative Weight of the Dimension. The higher the weight, the more important the dimension. It is used when balancing resources between different dimensions.
BWGHT	Balancing weight. This indicates the percentage of system balance rules applied to this dimension during calculation of XL as opposed to the specific requirements of the dimension. Range from 0 to 100.
ICA	Individual Control Attribute. This is a protection level. If 1, specific isolation and protection methods must be used for every application of this dimension. If 0, there is no ICA.
HOLD	If 1, a step in the dimension may be held by ARM if necessary. If 0, a step cannot be held.
PA	Execution Priority attribute. This indicates if steps are discriminated against inside the dimension, according to their job priority (XPRTY).
SYSTEM	The system on which the command is to execute.

Rules

- The dimension name and at least one of the other parameters must be present.
- The ARM level must be FULL for all parameters, except DIM and MPL which can be used in BASIC level.
- If the dimension is SYS only the MPL parameter can be specified.

Examples

```
MDDIM POOL4 MPL=(2,20)
                    {the multiprogramming level range of dimension
                    POOL4 is modified}

MDDIM IOF7 XL=(55,70)
                    {modification of the attributes of dimension IOF7}
```



8.3.9 MODIFY_DJP (MDDJP)

Purpose

Specifies the amount of transfer activity allowed between the local site and a remote site, or modifies the maximum number of job readers on the current site.

Syntax

```
{ MODIFY_DJP }
{ MDDJP      }

SITE = sitenm

TFRNB = tfnb

[ STRONG = bool ]

[ WAITIME = dec2 ]

[ NWKD = { NO|MON|TUE|WED|THU|FRI|SAT|SUN } ]

[ YEAR = dec4 ]

[ HOLIDAYS = dec3 ]

[ HOLIDAYS_MODE = {RPL|ADD|DL} ]

[ UFTWAIT = HH.MM ]
```

Parameters

SITE	Name of the site with which transfer activity is to be modified.
TFRNB	The number of transfers that can take place simultaneously between the local site and the one specified by SITE. When 0, no transfer is possible. If SITE=current site, then TFRNB specifies the maximum number of job readers that can be started simultaneously on the site.



STRONG	This parameter is meaningful only when TFRNB is less than the current number of transfers in progress. If 0 or not specified, the system waits for the normal completion of transfers in excess of TFRNB. If 1, excess transfers are immediately aborted.
WAITIME	Time, in minutes, between two connections to the site given in the SITE = parameter. When a first connection fails, the default value is 5 minutes.
NWKD	Non Working Weekdays is used to designate 1 to 3 non working days of the week. NO means that all are working days.
YEAR	Designates the year.
HOLIDAYS	The Julian number (1 to 366) for each of the 1 to 60 days in the year that are public holidays and no work is to be done.
HOLIDAYS_MODE	Specifies how public holidays given in HOLIDAYS are used:
RPL	replace public holidays in the current list of specified year,
ADD	add public holidays to the current list of specified year,
DL	delete public holidays from the current list of specified year.
UFTWAIT	Waiting time in hours and/or minutes, between two data transfers (minimum = one minute, maximum = 72 hours).

Rules

- If WAITIME is used, then SITE (and SITE only) must be used.
- If TFRNB is used, SITE must be used.
- If HOLIDAYS is used, YEAR must be used.



Examples

MDDJP SIT3 10 {10 transfer activities with site SIT3.}

MDDJP SIT3 5 STRONG
 {Reduce transfer activity and abort excess transfers.}

MDDJP WAITIME = 1 SITE = SIT3
 {A connection to SIT3 is attempted every one minute.}

MDDJP NWKD = (FRI, SAT)
 {The two non working weekdays are Friday and
 Saturday. The default id Saturday and Sunday.}

MDDJP NWKD = NO {All the days of the week are working days.}

MDDJP YEAR = 1991 HOLIDAYS =
(1, 91, 127, 128, 129, 130, 227, 228, 305, 359)
 {For the year 1991, the public holidays are: January
 1st, April 1st (Easter), May 1st (Labor Day), May 8th
 (1945 Victory), May 9th, May 10th, August 15th,
 August 16th, November 11th, and December 25th.}

MDDJP YEAR = 2000 HOLIDAYS = (1, 366) HOLIDAYS_MODE =
ADD
 {For the year 2000, add the public holidays: January
 1st and December 31st.}

MDDJP YEAR = 2000 HOLIDAYS = (1, 365) HOLIDAYS_MODE = DL
 {For the year 2000, delete the public holidays: January
 1st and December 30th.}

MDDJP YEAR = 2000 HOLIDAYS_MODE = RPL
 {For the year 2000, there are no public holidays.}

MDDJP UFTWAIT = 00:20
 {The connection to a remote site will be aborted if
 during 20 minutes no data transfer occurs.}



8.3.10 MODIFY_ERROR_LOG (MDERLOG)

Purpose

Modifies some entries in the SYS.ERLOG file. This command is to be used only under Service Center supervision.

Syntax

```
{ MODIFY_ERROR_LOG }
{ MDERLOG           }

      [{ DEVICE }   { *           }]
      [{           } = { dvtyp [dvnm] }]
      [{ DV        }   { dvc       }]
```

Parameters

DEVICE the device or devices for which error counters are to be reset, expressed as:

```
*           all devices,
dvnm       a specified device,
dvtyp      all devices of a type,
dvc        all devices of a class.
```

When omitted, all devices (*) are assumed.

Rules

None.

Examples

```
MDERLOG                    {reset all counters}
MDERLOG MS                 {reset counters for all disk units}
```



8.3.11 MODIFY_FILTER (MDFLT)

Purpose

Modifies (that is, activates or deactivates) a filter or a set of filters.

Syntax

```
{ MODIFY_FILTER }
{ MDFLT          }
      OPTION=    { SUSP      }
                  {          }
                  { RESTART  }

      { FILTERSET }
      {           } = fltst
      { FLTST     }

[ { FILTER }      ]
[ {           } = flt ]
[ { FLT      }    ]

[ { SYSTEM }      ]
[ {         } = name4 ]
[ { SYS    }      ]
```

Parameters

OPTION	the action to be performed: <ul style="list-style-type: none">– SUSP suspend the action of a filter or filter set,– RESTART activate a previously suspended filter or filter set.
FILTERSET	when used alone (without FILTER), specifies the set of filters to be modified. When used in conjunction with FILTER, specifies the filter set to which the filter to be modified belongs.
FILTER	the filter to be modified.
SYSTEM	The system where the command is executed.



Rules

None.

Examples

MDFLT SUSP A1 A101 {suspend filter A101 within filter set A1}

MDFLT RESTART A1 {reactivate all filters of set A1}

MDFLT SUSP A1 {suspend all filters of set A1}



8.3.12 MODIFY_HARDWARE_CONFIGURATION (MDHW)

Purpose

This command inserts hardware resources into the GCOS 7 configuration (IN) or removes them (OUT). It is used by the operator for dynamic reconfiguration, that is, while the system is running. This command is not used for dynamic reconfiguration of the DPS 7000/A, DPS 7000/200/300 machines. See the appropriate *Operator's Guide* (hardware) and Chapter 10 *Reconfiguration* for more information.

Syntax

```
{ MODIFY_HARDWARE_CONFIGURATION }
{ MDHW                               }

[          { hardware-unit } ]
[ UNIT = {                   } ]
[          { device-path  } ]

[          { IN   } ]
[ ACTION = {       } ]
[          { OUT  } ]

[ SHARE   {           } ]
[        = { bool   } ]
[ SHR     {           } ]

[ TWINNING {           } ]
[         = { bool   } ]
[ TWIN     {           } ]

[ NTWIN ]

-----

[ FORCE = { bool } ]
```




Parameters

UNIT	<p>Specifies the resources to be inserted into or removed from the configuration. There are two possibilities:</p> <ul style="list-style-type: none">– A hardware unit (4 characters) which can be:<ul style="list-style-type: none">a CSS resource (IPnm, EPnn, FPnn, GPnn, IOnm, MUhh, PXnm),a controller (xCnm),a peripheral device (dvnm).– A device-path (9 characters) representing the path to a peripheral device (xCnm.dvnm).
ACTION	<p>if IN, the unit specified in the UNIT parameter is inserted into the GCOS 7 configuration.</p> <p>if OUT, the unit specified in the UNIT parameter is removed from the GCOS 7 configuration.</p> <p>NOTE: The last IPU, CPU board or IOP of a configuration cannot be removed.</p>
SHARE	<p>This parameter must appear when ACTION=IN is specified if the unit is a shareable disk. There are two possible values:</p> <ol style="list-style-type: none">1 When this parameter is used, the shareable disk specified in the UNIT parameter is inserted into the GCOS 7 configuration and added to the list of SHARED devices.<p>The disk must be OUT or SHARED on the other system.</p>0 When this parameter is used, the shareable disk specified in the UNIT parameter is inserted into the GCOS 7 configuration and is private to the system.<p>The disk must be OUT on the other system.</p> <p>If this parameter is omitted for a shareable device, SHARE=0 is assumed and warning message AW60 (03) explained in the <i>Console Messages Directory</i> is sent to the operator. See also the <i>Coupled Systems User's Guide</i>.</p>



TWINNING	<p>This parameter concerns bisystem sessions on a redundant multiprocessor installation.</p> <ol style="list-style-type: none">1 CSS resources necessary to initiate another system on the installation are removed from the running GCOS 7 configuration.0 CSS resources available after termination of the second system in a bisystem session are inserted into the current GCOS 7 configuration.
NTWIN	<p>This parameter means no twin and it is the same as TWINNING=0.</p>
FORCE	<p>This parameter can only be used when ACTION=IN is specified for CPU and disk resources; the only possible value is 1.</p> <p>A disk device is unconditionally reinserted into the GCOS 7 configuration.</p> <p>When FORCE is specified for a shareable disk, exclusive access to the disk by another system is cancelled before its insertion into the current GCOS 7 configuration. This option should be used with care as it may interrupt use of the disk by another system.</p> <p>Let a shareable MS01 disk be accessible by an:</p> <ul style="list-style-type: none">- S0 system through the MC01 controller- S1 system through the MC02 controller <p>In case of permanent and "accidental" seizure of MS01 by S1 caused by:</p> <ol style="list-style-type: none">1. a misuse of MDHW MS01 IN SHR=0/1 commands as for example:<ul style="list-style-type: none">- for S0 MDHW MS01 IN SHR=1- then for S1 MDHW MS01 IN SHR=02. a hardware problem on MC02 controller <p>the only way for S0 to regain access to MS01 is to key in the MDHW MS01 OUT and then MDHW MS01 IN FORCE commands on behalf of the S0 system.</p>



The FORCE option forces the MS01 reintroduction by unlocking the "parasitic" seizure.

NOTE: this FORCE option must be used carefully because it may disturb the S1 operation.

Rules

- When an MU unit is specified, only the OPTION=IN parameter can be used.
- TWINNING cannot be used with another parameter.
- Either the TWINNING parameter must be used or the UNIT and ACTION parameters must be specified together in any MDHW command.
- The SHARE and FORCE parameters are mutually exclusive.
- An IOC can only be isolated if all channels linked to it are unused. When one allocated device with only one path through the IOC remains, the command is:
 - either rejected if the device is a system or resident disk or if it is a coupled volume,
 - or deferred in all other cases.

The state of all channels remains unchanged.

- In the current release, Boolean value parameter 0 is the only value which can be used in a TWINNING command because the only dynamic transition possible is from bisystem to monosystem.
- With the TWINNING parameter, 0 value may only be used when the other system session has terminated and when the COUPLED switch on the SOP whose ENABLE lamp is lit is placed in the ON position. This command automatically reinserts CSS resources left available by the termination of the second system into the current GCOS 7 configuration.
- When a memory unit has failed, all the MUs of the terminated system are kept out of the GCOS 7 configuration. The operator must set the valid MUs (except dedicated I/O cache memory units, see the rule below) in the configuration using as many MDHW MUxx IN commands as the number of valid MUs to be inserted.
- A dedicated I/O cache memory unit (MU) cannot be dynamically reconfigured. This MU is not accessible to the system and remains in this state after the reconfiguration command. In other words, the MDHW MUxx IN command cannot be used for this MU.
- When related to a MIRROR disk, the command MDHW IN/OUT SHARE=1/0 must be used in manual operating mode (MDMIR MANUAL).



- Except when a specific action is required on one device, both disks of a pair of mirrored disks must be kept in the same state (both IN with the same shareability attribute, or both OUT). Refer to the *Mirror Disks User's Guide*.

Examples

MDHW IP51 IN	{inserts IPU 51 into the configuration.}
MDHW EP04 OUT	{removes EPU 04 from the configuration}
MDHW FP05 IN	{inserts FPU 05 into the configuration}
MDHW TWIN=0	{after the termination of one system in a bisystem session, inserts all CSS resources left free by this termination into the current configuration.}
MDHW MS15 IN SHR=0	{shareable disk MS15 is inserted into the configuration as a private disk.}
MDHW MC05 OUT	{the disk controller MC05 is removed from the configuration.}
MDHW TC05.MT23 IN	{gives access to tape unit MT23 via tape controller TC05 (path name=TC05.MT23).}
MDHW MU1* IN	{inserts all MU modules in the second Main Memory Unit into the configuration.}



8.3.13 MODIFY_INPUT_READER (MDIR)

Purpose

Starts or stops a permanent reader.

Syntax

```
{ MODIFY_INPUT_READER }  
{ MDIR }
```

```
STATE = { PERM }  
        { NPERM }
```

Parameters

STATE PERM starts a permanent reader, while NPERM stops it.

Rules

None.

Examples

```
MDIR PERM {start a permanent reader}
```

```
MDIR NPERM {stop a permanent reader}
```



8.3.14 MODIFY_IOF (MDIOF)

Purpose

Modifies the lapse of time during which reconnection of accidentally disconnected IOF users is possible and/or the lapse of time after which idle IOF users are disconnected.

Syntax

```
{ MODIFY_IOF }
{ MDIOF      }

  [{ RECN_TIME } { *   }]
  [{           } = { 0   }]
  [{ RT        } { dec2 }]

  [{ IDLE_TIME } { *   }]
  [{           } = {     }]
  [{ IDT       } { dec2 }]
```

Parameters

RECN_TIME	the lapse of time during which reconnection is allowed as follows: * no time limit for reconnection 0 reconnection is not allowed nn reconnection must occur within nn minutes of the accidental disconnection.
IDLE_TIME	the lapse of time after which idle IOF users are disconnected. * No idle time limit is defined.



Rules

At least one of RECN_TIME or IDLE_TIME must be specified.

Examples

MDIOF *	{set no limit to reconnection time}
MDIOF 0	{no reconnection is allowed}
MDIOF 15	{reconnection must occur within 15 minutes of the accidental disconnection}
MDIOF 15 6	{set reconnection time to 15 minutes and idle time to 6 minutes}
MDIOF IDT=6	{set idle time to 6 minutes}



8.3.15 MODIFY_JOB (MDJ)

Purpose

Modifies the characteristics of jobs. A job may only be modified by its submitter, by its USER and by main or station operators.

Syntax

```

{MODIFY_JOB}
{MDJ      }

[RONS = (ron [ron] ...)]

[{{SELECT}  {jclass} [ {usern} [ {projnm} ] ] ]
[{{      } = {      } [ / {      } [ / {      } ] ] ]
[{{SEL    }  { * } [ { * } [ { * } ] ] ]

[CLASS = aa]

[{{PRIORITY}  ({{spr} [{{xpr} ] ) ] ]
[{{      } = ({{      } [{{      } ] ) ] ]
[{{PRTY    }  ({{ * } [{{ * } ] ) ] ]

[ON = (dec2 [dec2] ...)]

[OFF = (dec2 [dec2] ...)]

[{{SYSTEM}      ]
[{{      } = name4]
[{{SYS      }      ]

[{{JOBSTATE}      ]
[{{      } = ALL|SCH|EX|SUSP|IN|HOLD]
[{{JSTATE  }      ]

[{{JOBNAME}      ]
[{{      } = name8]
[{{JNM      }      ]

```


**Parameters**

RONS	a list of up to 16 Run Occurrence Numbers of the jobs to be modified. When omitted, all jobs are assumed.
SELECT	restricts the scope of the command to the jobs that belong to the specified combination of jclass/user name/project. If SELECT=* all jobs are modified.
CLASS	the new class for the selected jobs. The class is one or two letters in the range A to P or AA to PZ. It cannot be changed once job execution has begun.
PRIORITY	the new priority for the selected jobs expressed as a pair of digits as follows: (scheduling-priority execution-priority), spr is an integer in the range 0-7 inclusive, xpr is an integer in the range 0-9 inclusive. (The lower the number, the higher the urgency). An asterisk leaves the corresponding value unchanged.
ON/OFF	a list of the switches to be set on or off. A switch is identified by a number in the inclusive range 0 to 31.
SYSTEM	indicates the system where the command is executed.
JOBSTATE	jobs to be modified by current state; jstate must be one of the following: ALL, SCH, EX, SUSP, IN, HOLD
JOBNAME	jobs to be modified by name (name-8)

Rules

When RONS is used, neither SELECT, JOBSTATE nor JOBNAME can be used.

When neither RONS, SELECT, JOBSTATE, nor JOBNAME are specified, SELECT =* is assumed.

ON and OFF may not be used simultaneously; they are mutually exclusive.

RONS and SELECT are mutually exclusive.

As with other parameters of the command, only an operator may modify jobs that he did not submit.



Examples

MDJ CLASS=BZ {set all user-submitted jobs to class BZ}

MDJ PRIORITY=(*, 9) {set all user-submitted jobs' execution priorities to 9}

MDJ X222 CC (5, 3) {set job X222 to class CC with priorities 5 and 3}

MDJ X222 ON=(7 17 21)
{set switches 7, 17 and 21 of job X222 to ON}

MDJ PRIORITY=(7, *) {set the scheduling priority of all user-submitted jobs to 7}

MDJ SEL=*/*/MYPROJ CLASS=AA
{set all jobs belonging to the MYPROJ project to class AA}

MJ JOBNAME=TEST2 ON=(1)
{set switch 1 to ON for all jobs named TEST2}



8.3.16 MODIFY_LOAD (MDLD)

Purpose

Modifies the characteristics (multiprogramming level, default scheduling, and execution priorities) for specified classes.

Syntax

```
{ MODIFY_LOAD }
{ MDLD          }

[ { CLASSLIST } { *          } ]
[ {           } = { aa        } ]
[ { CLASS     } { aa[-aa]   } ]

[ { MULTI_LEVEL }          ]
[ {           } = dec3    ]
[ { MULTI      }          ]

[ { PRIORITY } ( { spr } { xpr } ]
[ {         } = ( {     } , {     } ]
[ { PRTY     } ( { *   } { *   } ]

[ MDMULTI =bool ]

[ { SYSTEM }          ]
[ {         } = name4 ]
[ { SYS    }          ]
```

Parameters

CLASSLIST the classes or range of classes whose characteristics are to be modified, expressed as:

* all classes,
class for a single class,
class-class for a range of classes,



MULTI_LEVEL	<p>the multiprogramming level for each one of the selected classes. The range is from 0 to 255.</p> <p>Some classes dedicated to service jobs (e.g. "Y") have a lower limit for the multiprogramming level. If the value given to the MULTI_LEVEL parameter is outside that limit, the multiprogramming level is set to that limit.</p>
PRIORITY	<p>the new priority for the selected jobs expressed as a pair of digits as follows:</p> <p>(scheduling-priority execution-priority),</p> <p>spr is an integer in the range 0-7 inclusive,</p> <p>xpr is an integer in the range 0-9 inclusive.</p> <p>(The lower the number, the higher the urgency). An asterisk (*) leaves the corresponding value unchanged.</p>
MDMULTI	<p>if 1 enables MULTI-LEVEL to be modified later.</p> <p>if 0 MULTI-LEVEL cannot be modified later.</p>
SYSTEM	<p>This parameter indicates the system where the command is executed.</p>

Rules

At least one of CLASSLIST, MULTI_LEVEL, PRIORITY, or MDMULTI must be specified.

Examples

MDLD * 2	{set the multiprogramming level of each class to 2}
MDLD AA-PZ 2	{set the multiprogramming level of classes AA to PZ to 2}
MDLD AA 2 PRTY=(3,3)	{set the multiprogramming level of class AA to 2 and both the scheduling and execution priorities to 3}
MDLD PRTY=(3,3)	{set both the scheduling and execution priorities of all classes to 3}



8.3.17 MODIFY_MIRROR_MODE (MDMIR)

Purpose

This command changes the mirror operating mode from AUTOMATIC to MANUAL and vice versa.

It is available only if the system has been configured with MIRROR = YES.

Syntax

```
{ MODIFY_MIRROR_MODE }
{ MDMIR                }

MODE = { AUTO          }
        {               }
        { MANUAL       }
```

Parameters

MODE AUTO changes the operating mode from MANUAL to AUTOMATIC. It has no effect if the mode is already AUTOMATIC.

 MANUAL changes the operating mode from AUTOMATIC to MANUAL. It has no effect if the mode is already MANUAL.

Rules

- In MANUAL operating mode, the system waits for an operator action or reply to a MOUNT or a RECOVER request related to a MIRROR disk.
- The AUTOMATIC operating mode allows the system not to wait for the operator action or reply for such events: copy invalidation is performed and the processing continues with the MIRROR ALONE disk.

Refer to *Mirror Disks User's Guide*.

Examples

```
MDMIR MODE = AUTO    {Changes operating mode from MANUAL to
                     AUTOMATIC and displays the message:
                     MR 16 AUTOMATIC MIRROR MODE}
```



8.3.18 MODIFY_MOT (MDMOT)

Purpose

Defines the Message Of Today (MOT) or erases it. The Message Of Today is printed in the Job Occurrence Report (JOR) or all jobs and displayed to each IOF user when he logs on.

Syntax

```
{ MODIFY_MOT }
{ MDMOT      }

    [ { MESSAGE }           ]
    [ { MSSG   } = char200 ]
    [ { MSG    }           ]

    [ { SYSTEM }           ]
    [ {       } = name4   ]
    [ { SYS   }           ]
```

Parameters

MESSAGE	the text of the Message-Of-Today. When omitted the current text of the Message-Of-Today is erased.
SYSTEM	This parameter indicates the system where the command is executed.

Rules

None.

Examples

```
MDMOT 'Welcome to the XYZ Computer Center'
      {Defines the Message-Of-Today.}

MDMOT                                {Erases the Message-Of-Today.}
```



8.3.19 MODIFY_OPERATING_MODE (MDOM)

Purpose

Modifies the protocol of interaction between the system and the operator.

Syntax

```
{ MODIFY_OPERATING_MODE }
{ MDOM
    [
        [ MODE = { INT } ]
        [
            { CONS } ] ] ]
```

Parameters

MODE

the new protocol of interaction, as follows:

- INT In this mode, the S: prompt is displayed permanently, except when the system is executing a request keyed in by the operator. The advantage is that the dialog with the system is simplified. The disadvantage is that system messages must wait until a command is entered before they can be displayed on the console.
- CONS The S: prompt is only displayed when you press the <Break> or <Insert> key. The advantage is that messages are visualized as they appear.

On the system main console, the default mode is CONS.

Rules

None.

Examples

```
MDOM CONS      {console protocol: a break must be issued before each
                 input}

MDOM INT       {revert to interactive protocol}
```



8.3.20 MODIFY_OUTPUT (MDO, MO)

Purpose

Modifies the characteristics of outputs.



IMPORTANT:

This command applies only to the outputs on which the submitter has access rights

Syntax

```

{ MODIFY_OUTPUT }
{ MDO           }
{ MO           }

[           { (output14 [ output14 ] ...) } ]
[ OUTPUT = { output14-output14           } ]
[           { * | ALL                     } ]

[ SELCLASS = a [-a] ]

[ CLASS = a ]

[ PRIORITY = {0|1|2|3|4|5|6|7} ]

[ DEST = [ name8.] name8 ]

[ COPIES = nn ]

[ { STATION } { *      } ]
[ {          } = {       } ]
[ { ST      } { stnm  } ]

[ { SELECT } { user/project } ]
[ {          } = { */project  } ]
[ {          } { */*         } ]
[ { SEL    } { *           } ]

[ { JOBNAME }           ]
[ {          } = name8 ]
[ { JNM     }           ]

[ { SYSTEM }           ]
[ {          } = name4 ]
[ { SYS    }           ]

```




CLASS	the new class for the selected outputs: a letter in the range A to Z.
PRIORITY	the new priority for the selected outputs: a digit in the range 0 to 7 (the lower the priority number, the higher the urgency).
DEST	the new destination of the output(s) in the form: [host-name.] station-name
COPIES	the new number of copies (see <i>DPRINT</i> command).
STATION	restricts the scope of the command to outputs attached to the station specified. When omitted or when * is specified, there is no station restriction.
SELECT	restricts the scope of the command to outputs belonging to a specified user and/or project. Not meaningful when OUTPUT specifies a list of outputs (format 1). By default, only the outputs belonging to the command submitter are selected.
JOBNAME	restricts the scope of the command to the outputs of jobs with the name specified. If JOBNAME is specified, the default value for OUTPUT is *.
SYSTEM	indicates the system on which the command is to be executed.

Rules

- SELCLASS is ignored for names of specific outputs (Xnnnn:nnnnn or Xnnnn:name).
- At least one of CLASS, DEST, PRIORITY, or COPIES must be used.
- DEST parameter must begin with a letter.
- If several output of the same job are given the same name, MDO Xnnnn :name only affects the first output with that name encountered by the system.
- When SELCLASS is omitted, the modified CLASS, PRIORITY, and DEST (but not the COPIES) apply to all existing and future outputs of the job. When SELCLASS=A-Z, only currently known outputs are modified.
- While the output is being printed, no modification in the number of copies can be taken into account before the start of the next copy. For example, if MDO COPIES=3 is entered during the printing of the third copy, the printer prints a fourth copy before it stops.
- CLASS, DEST and PRIORITY are ignored if output is already being printed.



- MDO COPIES=nn takes into account the number of copies already printed. For example, if 6 copies were requested and 3 have been printed, MDO Xron COPIES=8 does not change the number of copies already printed. Five more are printed for a total of 8. However, MDO Xron COPIES=4 results in just one more copy for a total of 4.
- The parameters specified in \$JOB with the EJ command or an anticipated MO (MODIFY_OUTPUT) command override those specified in a batch job (OVL SYSOUT or WRITER) or in a IOF directive (DPR).

Examples

```
MDO X123 DEST=MAIN {print all outputs of X123 on one of the main site
                    printers}

MDO X123 C D 5      {set outputs of X123 of class C to class D and
                    priority 5}

MDO (X123:2 X123:RES) CLASS=C
                    {set two selected outputs of job X123 to class C}

MDO X123:1 CLASS=C DEST=HST1.STN2
                    {modify class of one output and print it on a remote
                    site}

MDO X123:RES COPIES=6
                    {provide six copies of named output}

MDO * DEST=MAIN    {reroute all outputs belonging to the command
                    submitter and attached to the current station, to the
                    MAIN station}

MDO * DEST=MAIN ST=STN
                    {reroute all outputs belonging to the command
                    submitter and attached to station STN, to station
                    MAIN}

MDO X123 PRIORITY=6
                    {modify the priority of all current and future outputs of
                    job X123}

MDO X123 A-Z PRIORITY=1
                    {modify only the priority of currently existing outputs}
```



MDO X123-X2167 CLASS=G
{modify the class of all outputs, belonging to the command submitter, of the jobs whose RON is in the range X123 to X2167}

MDO X34-X382 SELCLASS=A-E PRIORITY=2
{modify the priority of all outputs, belonging to the command submitter, of the jobs whose RON is in the range X34-X382 and whose output classes are in the range A to E}

MDO * P Q
{modify all class P outputs belonging to the command submitter, to class Q}

MDO * SELECT=U1/P1 CLASS=A COPIES=2
{modify the class and number of copies of all outputs belonging to user U1 of project P1}

MDO * JOBNAME=JOB1 PRIORITY=1
{modify the priority of all outputs, belonging to the command submitter, of all jobs named JOB1}



8.3.21 MODIFY_POWER_TIMER (MDPWRTM) (MDPOWER) (TPW) (MDPWR)

Purpose

This command modifies the values of both GCOS 7 power timers:

- The Power Off Timer, indicating at what time the system will shut down, then power-off.
- The Power On Timer, indicating at what time the system will power-on.

Syntax

```
{ MODIFY_POWER_TIMER }
{ MDPWRTH              }
{ MDPOWER              }
{ MDPWRTPW            }

[ OFF      { +interval } ]
[          = { date/time } ]
[ OFFDTM  { CANCEL   } ]

[ ON      { +interval } ]
[        = { date/time } ]
[ ONDTM  { CANCEL   } ]
```

Parameters

- OFF** Modifies the value given to Power Off timer. Can be given either in absolute form <date/time> or in relative form +<interval>. See examples below. If CANCEL is specified, the contents of the power off timer are cancelled.
- ON** Modifies the value given to Power On timer.

Rules

- The Power On Timer cannot be set before the Power Off Timer.
- The length of each element in the OFF and ON expressions is limited to 4 characters, and the total length of each expression is 14 characters.
- This command is not available on DPS 7000/2X0.



- Avoid setting the Power On Timer too short a time after the Power Off Timer: if the system receives the power-on command while the shutdown is still in progress, that command is ignored, and the system remains powered off.
- This command is only available if H_TPC is started (The most convenient way to do this is to add the command to do so in the STARTUP file).

Examples

MDPWR {Displays the available ranges for the Power On and Power Off Timers (ranges depend on hardware)}

MDPWR OFF=+02/03/00 {Shut down the system and power-off in two days and three hours.
Note: in the increment format only the dd/hh/mm components are accepted.}

MDPWR OFF=+2/00/ ON=+2/00/00 {Shut down the system and power-off in two hours, then power-on in two days}

MDPWR OFF=90/03/21/09/30 ON=CANCEL {Shut down the system and power-off on March 21st, 1990, at 9:30 AM; the Power On Timer is cancelled}



8.3.22 MODIFY_RESTART_OPTIONS (MDRSO)

Purpose

Modifies some of the system initialization options, namely those related to remote operations.

Syntax

```
{ MODIFY_RESTART_OPTIONS }  
{ MDRSO }
```

```
[REPEAT = bool]
```

```
[AUTO = bool]
```

```
[SILENT = bool]
```

```
[ { DEVICE } ]  
[ { } = MSxx ]  
[ { DV } ]
```

Parameters

REPEAT

When 1, the next system shutdown or crash is immediately followed by a restart, reload or re-initialization (depending on the system and the cause of the shutdown) without any operator intervention. This operation takes into account the disk unit identification stored in the service processor through the DEVICE parameter (refer to Section 3.3 *Operating Modes*). Consequently, when REPEAT is set to 1 the operator must check the device identification stored in the service processor.

When 0, the next system shutdown will require a manual intervention from an operator to restart the system. In the case of CRASH the system is restarted and reloaded automatically, taking into account the unit that was the system disk in the current session. This operation is necessary to process consistently the DUMP related to the CRASH.



AUTO	<p>When 1, the next system restart (whether with or without operator intervention) is automatic, i.e. does not ask the operator any questions, provided that default replies have been stored in the RELOAD_OPTIONS library member, as appropriate.</p> <p>When 0, the next system restart is run by the operator who is asked questions.</p>
SILENT	<p>When 1, the next system restart (with or without operator intervention) will result in no message being issued on the operator console; when 0, messages are issued.</p>
DEVICE	<p>The identification of the system disk from which the system is to be initialized. Unlike the other three parameters, this option is retained for all subsequent system initializations until modified by a new MODIFY_RESTART_OPTIONS command.</p>

Rules

If AUTO = 0 then SILENT is ignored.

Examples

```
MDRSO DV=MS03      {redefine the initialization device}
MDRSO AUTO SILENT  {next restart will ask no questions and issue no
                    messages}
MDRSO REPEAT = 0   {manual intervention is required for next restart}
```




8.3.23 MODIFY_STATION (MDST)

Purpose

Modifies the characteristics of a local (station operator) or remote (main operator) station.

Syntax

```
{ MODIFY_STATION }
{ MDST           }

[ ADDLDV = name4 ]

[ { DEVCLASS }   { PR [/attr] } ]
[ {           } = {           } ]
[ { DVC       }   { CD [/attr] } ]

[ { DEVICE }           ]
[ {           } = rfile78 ]
[ { DV      }           ]

[ SELCLASS = a [-a] ]

[ LCASE = bool ]

[ CLDRV = { name4 } ]
[           { * } ]

[ PRTDEST = name8 ]

[ PHCDEST = name8 ]

[ EBANNER = bool ]

[ FBANNER = {0|1|2} ]

[ BANCHAR = char2 ]

[ { RESCUE_ON }           ]
[ {           } = name8 ]
[ { RCON      }           ]

[ { RESCUE_OFF }           ]
[ {           } = name8 ]
[ { RCOFF     }           ]
```



```
[ { RESCUE_CLASS } ]
[ {                } = a [-a] ]
[ { RCCLASS       } ]

[ { STATION       } ]
[ {                } = stnm ]
[ { ST            } ]
```

Parameters

ADDLDV	the four-character name of a logical device to be defined and made known to GCOS 7.
DEVCLASS	(with ADDLDV) the device class of the logical device: printer, card reader or card punch.
DEVICE	(with ADDLDV) the DPS 6 path name (possibly including the site name) identifying the file or device to act as the named logical device. See the rfile78 parameter of the START_OUTPUT_WRITER command. Use of MDST DEVICE = rfile78, allows you to give a "DPS 7000 device-like" name to a remote device or file, for further use by a SOW or TOW command. (Such a name cannot be used to re-identify a DPS 7000 device.)
SELCLASS	(with ADDLDV) the default output class(es) attached to the logical device. This may be expressed as: class for a single class class-class for a range of classes where class is a letter (A to Z). When omitted, all classes are assumed.
LCASE	(with ADDLDV) when 1, the logical device is requested to keep lowercase when printing.
CLRLDV	the name of a logical device whose definition is to be erased. When * is used, all device definitions and options attached to the station are erased.
PRTDEST/PCHDEST	the name of a station to which printouts (PRTDEST) or punchouts (PCHDEST) are directed. This can prove useful if a station has no printer (punch) or if it is expected to be down for a long time.



EBANNER	whether an end banner is required (1) or not (0) for outputs.
FBANNER	The number of front banners (0, 1, 2) for outputs.
BANCHAR	a 2-character string. The first character to be used as a banner separator, and the second one to print the large characters on the banner.
RESCUE_ON/RESCUE_OFF	the name of another station to be used for rescuing (RESCUE_ON) or to stop rescuing (RESCUE_OFF) the station. When a station is rescued, output delivered and intended for that station is directed to the rescuing station, upon a special START_OUTPUT_WRITER command.

Rules

- If ADDLDV is used, then both DEVCLASS and DEVICE must be used.
- If DEVCLASS, DEVICE, SELCLASS or LCASE is used, then ADDLDV must be used.
- If RESCUED_CLASS is used, then either RESCUE_ON or RESCUE_OFF must be used.

Examples

```
MDST PR99 PR/H160 $YORK:>SPD>LPT00 A-I
    {define PR99 as the name of a printer at DPS 6 site
    YORK where it is known as >SPD>LPT00.
    When command SOW PR99 is issued, the writer
    processes classes A to I}

MDST CLRLDV=PR99    {erase the previous definition}

MDST RCON=ST3 ST=ST2
    {ask station ST2 to be rescued by station ST3 (this can
    only be issued by a main operator)}

MDST RCOFF=ST3 ST=ST2
    {cancel the effect of the preceding command}

MDST RCON=ST3 RCCLASS=A-Q
    {ask for station ST3 to rescue for output classes A to Q
    (used by a station operator)}
```



8.3.24 MODIFY_SYSTEM_TRACE (MDSYSTR) (MSYSTR)

Purpose

It activates or resets the following trace system selection filters for domains:

- the active trace event EVID range,
- the equality of a STRING to a trace event substring,
- the submitter XRON.

The MSYSTR command updates only the filters specified.

NOTE:

This command should be used under supervision of a Bull support engineer.

Syntax

```
{ MODIFY_SYSTEM_TRACE }
{ MDSYSTR                }
{ MSYSTR                  }

[ { DOMAIN } { ALL } ]
[ { DOM } = { } ]
[ { TDOM } { name63 } ]

[ EVID = ( dec4,dec4 ) ]

[ STRING = hex8,OFFSET = hex2 ]

[ XRON = { ron | Xnnnn } ]

[ RESET = bool ]
```

Parameters

DOMAIN	It is the name of a system trace domain, or if ALL (the default) is specified, all domains. Maximum length is 63 characters.
EVID	It is the active trace range, represented by a minimum value followed by a maximum value. The effective range is 1 - 4095. The maximum value must be greater than or equal to the minimum value.



STRING	It is the string value (max length = 8 char hex) to be compared to an event substring.
OFFSET	This is the substring displacement (max 2 char hex) in the trace event: $0A \leq \text{OFFSET} \leq 19$
XRON	This is the external RON number. Maximum length is 5 characters.
RESET	If 1, all trace domain filters are cleared; if 0, the filters are not cleared.

Rules

- The EVID filter applies to distributed requests that trace events with HDR=0 or HDR=1, and to variable length data trace events.
- The STRING filter is defined for fixed length distributed trace event requests.
- The XRON filter applies to all distributed trace requests. This filter must be cleared by operator command when the defined job terminates.
- RESET clears all trace domain filters.
- RESET and EVID are mutually exclusive.
- RESET and STRING are mutually exclusive.
- RESET and XRON are mutually exclusive.

Examples

MSYSTR XRON=X25 {Accept only trace requests issued from job X25 for all trace domains.}

MSYSTR DOMAIN=MYTDOM EVID=(128,256) XRON=X25
 {For domain MYTDOM, accept only trace requests issued by job X25, and filter trace event requests with the specified EVID range.}

MSYSTR DOMAIN=MYTDOM RESET
 {Reset all MYTDOM trace domain filters.}



8.3.25 MODIFY_TERMINAL_OUTPUT (MDTO)

Purpose

Modifies the characteristics of an output to be printed by the Generalized Terminal Writer, or makes a Generalized Terminal Writer output eligible to be printed by the system output writer. Refer to the *GTWriter User Guide* for a complete description.

Syntax

```
{ MODIFY_TERMINAL_OUTPUT }  
{ MDTO }  
  
    OUTPUT = Rnnnn  
  
    [CLASS = a]  
  
    [PRIORITY = dec1]  
  
    [TWDEST = name12 ]  
  
    [OWDEST = { name8 | MAIN } ]  
  
    [COPIES = dec2 ]  
  
    [NUMBER = bool ]  
  
-----  
  
    [MEDIA = name6 ]
```

Parameters

OUTPUT	the name of the output to be modified. A Generalized Terminal Writer output name has the form Rnnnn and is provided as the result of the DPRINT command (with parameter TWDEST).
CLASS	the new class for the output: a letter in the range A to Z.
PRIORITY	the new priority for the output: a digit in the range 0 to 9.
TWDEST	the name of the terminal or pool of terminals where the output is to be redirected.



OWDEST	the name of the station where the output is to be printed by the system output writer. When MAIN is specified, the system main pool of printers is assumed.
COPIES	number of copies.
MEDIA	This parameter defines both the printing form and the paper or belt to be mounted. When TWDEST is not specified, the first two characters refer to a paper belt and the last four to a paper format. When TWDEST is specified, it defines the paper to be used on the terminal; if omitted, the default is the form currently mounted on the terminal specified by TWDEST.
NUMBER	If 1, the line numbers are printed.

Rules

OWDEST and TWDEST are mutually exclusive.

Examples

```
MDTO R1234 TWDEST=POOL12
      {Redirect the output to another pool of terminals.}

MDTO R1234 OWDEST=MAIN
      {Redirect the output to the system main pool of
      printers.}

MDTO R1234 CLASS=T PRIORITY=2
      {Change the output class and priority.}
```



8.3.26 MODIFY_TERMINAL_WRITER (MDTW)

Purpose

Modifies the characteristics of a terminal or pool of terminals used by the Generalized Terminal Writer. Only the owner of the object terminals or users declared as Master Operators for the Generalized Terminal Writer may use this command. Refer to the *GTWriter User Guide* for a complete description.

Syntax

```
{ MODIFY_TERMINAL_WRITER }
{ MDTW                       }
{ MTW                         }

CURRENT_TWDEST = name12

[ TWDEST = { name12 | NO } ]

[ OWDEST = { name8 | MAIN } ]

[ OWNER = { name12 | NO } ]

[ MOUNT = bool ]

[ KEEP = bool ]

[ INVCHAR = { NONE | char1 } ]

{ PADDING }
[ {           } = dec2 ]
{ PAD      }

{ BLOCKING }
[ {           } = dec5 ]
{ BLOCK    }

[ ID = name8 ]

[ NODE = name4 ]

[ RETRYCT = dec3 ]

[ MEDIA = char6 ]
```




```
[ TEST = bool ]  
[ ACCOUNT = bool ]  
[ STREAM = bool ]  
[ DISCO = bool ]
```

Parameters

CURRENT_TWDEST	the name of the terminal or pool of terminals whose characteristics are to be changed.
TWDEST	the name of a terminal or pool of terminals where all outputs addressed to CURRENT_TWDEST are to be redirected. If NO is specified, redirection is discontinued and outputs are again handled by the original terminal or pool.
OWDEST	the name of a station where all outputs addressed to CURRENT_TWDEST are to be redirected and processed by the system output writer. When MAIN is used, the pool of system printers is assumed.
OWNER	the name of the owner of the terminal (that is, the person who is in charge of mounting the paper and performing similar operations). When NO is specified, the terminal has no owner; mount and recovery requests are sent to the owner of the output being printed.
MOUNT/KEEP/INVCHAR/PADDING/BLOCKING/RETRYCT	refer to the <i>Generalized Terminal Writer User's Guide</i> .
ID	the terminal mailbox name to be changed.
NODE	the node name to be changed.
MEDIA	the name of the paper (media) which is mounted on the terminal. This is used in response to a mount request or to specify that the named paper is premounted.
TEST	(with MEDIA only) requests the printing of the print test defined for the paper (media) specified by parameter MEDIA.
ACCOUNT	if 1, the system accounting must be activated for this terminal; if 0, the system accounting does not have to be activated for this terminal.



STREAM	if 1, reports are printed at the terminal without any control or formatting; if 0, the GTWriter manages the control and formatting of reports.
DISCO	if 1, the terminal is logged off by the GTWriter at the end of each report; if 0, the terminal is logged off by GTWriter when there is no further report to be printed at this terminal (unless the terminal has the KEEP attribute).

Rules

- TWDEST and OWDEST are mutually exclusive.
- TEST may be specified only when MEDIA is specified.

Examples

```
MDTW POOL2 TWDEST=POOL1
                                {redirect all outputs intended for POOL2 to POOL 1}
MDTW POOL2 TWDEST=NO
                                {revert outputs for POOL2 to original destinations}
MDTW POOL2 OWDEST=MAIN
                                {redirect outputs for POOL2 to a system printer}
```



8.3.27 MODIFY_TIME (MDTIME) (MDTM)

Purpose

Modifies the standard local time; useful to set winter and summer time.

Syntax

```
{ MODIFY_TIME }
{ MDTIME      }
{ MDTM       }

SHIFT = { [+|-] HH[.MM] }
        {          SYNC   }

[ { SYSTEM } ]
[ {         } = name4 ]
[ { SYS   } ]
```

Parameters

SHIFT Given in hours and minutes, this parameter specifies the shift between GMT (Greenwich Mean Time) and the standard local time. When used, HH (hours) must be 01 to 23, MM (minutes) 01 to 59. The default SHIFT is positive (+) making local time lead GMT. When negative, GMT leads local time.

With the SYNC parameter, the command takes the new Time Deviation from the Time Deviation Register (TDR) of the Service Processor and enters it in GCOS 7 as the Time Deviation. Meaningless on DPS 7000/2X0/3X0 systems.

SYSTEM This parameter indicates the system where the command is executed.

NOTE:

From GCOS 7-V7 TS7458, system time can be set backward, using the MODIFY_TIME command (even when journalization is active).



Examples

MODIFY_TIME SHIFT=+1 {local time leads GMT by 1 hour}
MODIFY_TIME SHIFT=1

MODIFY_TIME SHIFT=0 {local time is GMT}

MODIFY_TIME SHIFT= -2 {GMT leads local time by 2 hours}



8.3.28 MODIFY_USER_REQ (MDUR)

Purpose

Modifies file transfer or jobset requests.

Syntax

```
{MODIFY_USER_REQ}
{MDUR
}

REQUEST = {name8 [name8] ...}

[RCLASS = a]

[RRIORITY = 0|1|2|3|4|5|6|7]

[HOST = name8]

[
  { IMMED
}
[
  { [mm.dd.[yy]yy].hh.mm}
]
[WHEN = {
}
[
  { +dddd{W|D|H|M}
}
[
  { when-clause
}

[
  {W|WEEK}
]
[
  {D|DAY}
]
[
  {H|HOUR}
]
[EVERY = ([dec5] {M|MIN} [dec5])]
[
  {MONTH}
]
[
  {WKD}
]
[
  {NWKD}
]

[TIMEOFF = bool]

[WAIT = (name8 [name8] ...)]

[NOWAIT = bool]

[
  {CANCEL|NO}
]
[
  {IMMED}
]
[RRESTART = {
}
[
  {delay}
]
[
  {delayE}
]
```



Parameters

REQUEST	<p>a list of up to 16 names of requests to be modified. Each name may take one of the following three forms:</p> <ul style="list-style-type: none">– Jnnnnnnn a Jobset Occurrence Number (JON)– Fnnnnnnn a File transfer Occurrence Number (FON)– name8 the symbolic name of a request, as specified by the NAME parameter of the ENTER_JOB_REQ or ENTER_FILETRANS_REQ commands. <p>JONs and FONs are provided as results of the ENTER_JOB_REQ or ENTER_FILETRANS_REQ commands. They are unique identifiers for the corresponding requests.</p>
RCLASS	<p>the new class for the request, expressed as a letter in the range A to Z.</p>
RPRIORITY	<p>the new priority for the request, expressed as a number in the inclusive range 0 to 7. The lower the number, the higher the urgency.</p>
HOST	<p>(jobset requests only) the new host where the jobset is to be executed.</p>
WHEN/EVERY/WAIT	<p>new wait conditions for the request. See the parameters with the same names in ENTER_JOB_REQ and ENTER_FILETRANS_REQ commands. If EVERY = (0,0) is specified, the repeat condition is cancelled and other enqueueing parameters are left unchanged.</p>
TIMEOFF	<p>If 1, cancel the WHEN and EVERY conditions.</p>
NOWAIT	<p>If 1, cancel the WAIT condition.</p>
RRESTART	<p>Specifies various possibilities for handling requests that were received while the system was off.</p> <p>For further information, consult Appendix E.</p>



Rules

- WAIT and NOWAIT are mutually exclusive.
- TIMEOFF is exclusive of WHEN and EVERY.
- HOST is ignored for file transfer requests.
- When the year is expressed with 4 digits in the WHEN expression, it must verify the interval: 1961 = < yyyy < 2060.

Examples

```
MDUR J18 Z {Change class of J18 to Z.}
```

```
MDUR F36 RRIORITY=0  
 {Change priority of F36 to 0.}
```

```
MDUR MYJS HOST=HST1  
 {Change execution host of jobset MYJS.}
```

```
MDUR J37 TIMEOFF {Ignore time condition for J37.}
```

```
MDUR F5 WAIT=(F3 F4)  
 {Make F5 begin only when both F3 and F4 have been  
 completed.}
```

```
MDUR J1234 WHEN=03.01.9200.00 EVERY=52W  
 {Execute on March 1st 1992 and repeat every 52  
 weeks.}
```



8.3.29 MODIFY_VOLUME (MDV)

Purpose

Makes allocation of temporary files (im)possible on a resident volume or clears the locks set by the other system (of a set of coupled systems) after a failure of this other system.

Syntax

```
{ MODIFY_VOLUME }  
{ MDV           }
```

NAME = name6

```
ACTION = { CLEAR }  
         { TEMP  }  
         { NTEMP }
```

Parameters

NAME the name of the volume (Volume Serial Name) which is the object of the action.

ACTION the action to be performed on the volume expressed as:

CLEAR - to clear the locks set by the other system of a set of two coupled systems after a failure of that other system. Meaningful on a SHARED volume only (rejected otherwise).

TEMP - to allocate temporary files on named volume.

NTEMP - to make allocation of temporary files no longer possible on the named volume.



Rules

When related to a SHARED MIRROR volume, the MDV CLEAR command must be used after issuing the RSCMIR command.

Refer to *Mirror Disks User's Guide*.

Examples

```
MDV XYZ123 CLEAR      {clear locks on coupled volume XYZ123}
MDV WK12 TEMP         {to allocate temporary files on resident volume
                       WK12}
MDV WK12 NTEMP        {make allocation of temporary files on resident
                       volume WK12 no longer possible}
```



8.3.30 MODIFY_XL_CLASS (MDXLC)

Purpose

Modifies one or more parameters of an XL Class (XLC, or eXecution Level Class). See the *ARM User's Guide* for full details.

Syntax

```
{ MODIFY_XL_CLASS }
{ MDXLC           }

      { XL_CLASS }
      {           } = dec2
      { XLC       }

[      { F } ]
[ ATTR = { FS } ]
[      { O } ]

[ { HARDWARE_PRIORITY } ]
[ { HWPRTY             } = (dec2,dec2) ]
[ { HPRTYRANGE        } ]

[ { ENTRY_PRIORITY } ]
[ {                 } = dec2 ]
[ { EPRTY          } ]

[ MAXCPU = dec3 ]

[ { SYSTEM } ]
[ {       } = name4 ]
[ { SYS   } ]
```

Parameters

XLC	current XLC named in the command. New XLCs cannot be created.
ATTR	type of CPU processing. There are three possible control modes: Fixed non sliced (F), Fixed sliced (FS) or optimized (O). The Fixed sliced control mode uses the slice mechanism. The Optimized control mode uses dynamic changes of hardware priority.



HWPRTY/HPRTYRANGE	range (from 8 to 15) of hardware priorities that may have a process belonging to the CPU class. The two hardware priorities may be equal. The highest hardware priority of XL CLASS n must be inferior to the highest hardware priority of the XL CLASS (n-1). (see <i>Rules</i> below)
EPRTY	entry priority of a process. It is the priority a process is allocated when it is started. Must be within the priority range (see <i>HPRTYRANGE</i> above).
MAXCPU	maximum CPU consumption rate a process may use before it is penalized. Range is from 0 to 100.
SYSTEM	This parameter indicates the system where the command is executed.

Rules

- An XLC number and at least one of the other parameters must be specified.
- The lowest priority value for a given XLC (on the left inside the parenthesis) must be less than or equal to the lowest priority value for the next XLC. Remember that hardware priorities increase as their corresponding values decrease; priority 8 is greater than priority 15!

Example

```
MDXLC 1 HWPRTY=(8,13)
MDXLC 2 HWPRTY=(9,15) is accepted
MDXLC 3 HWPRTY=(9,9)

MDXLC 2 HWPRTY=(9,9)
MDXLC 3 HWPRTY=(8,9) is refused
```

(Keep your eyes on the left-hand column: 8-9-9 is OK, 9-8 is not.)

Example

```
MDXLC XLC=3 EPRTY=9 MAXCPU=70 HPRTYRANGE=(8,10)
      {Update some parameters of XLC 3}
```



8.3.31 MWENVT (Modify Working Environment)

Purpose

Modifies the current working environment or displays the names of the environments that may be accessed by the operator. Several different environments may be accessible to operators: system operator, network operator, device operator, etc. Their definition is dependent on the installation.

Syntax

MWENVT

[ENVT = name12]

Parameters

ENVT the name of the environment which is to become the new current working environment. When omitted, the command lists the names of the environments that are accessible to the operator.

Rules

The named environment must:

1. be the name of an existing environment,
2. be accessible to the project under which the operator is logged.

Examples

MWENVT {list the accessible environments}

MWENVT PPREP {switch to environment PPREP}



8.4 "N" and "R" Commands

8.4.1 NAME_VOLUME (NV)

Purpose

Names a non-standard disk, tape, or cartridge volume. This may be used to name the premounted non-standard volume or as a reply to a request to mount the volume, usually to perform a volume preparation.

Syntax

```
{ NAME_VOLUME }  
{ NV           }  
  
      { DEVICE }  
      {         } = dvnm  
      { DV      }  
  
VOLNAME = volnm
```

Parameters

DEVICE	the identification of the device on which the named volume is mounted or premounted.
VOLNAME	the name of the premounted volume.

Rules

None.

Examples

NV MT03 TAPE6	{Name volume TAPE6 on tape drive MT03.}
NV MS02 INPUT	{Name volume INPUT on disk drive MS02.}



8.4.2 RELEASE_GCL_CACHE (RGC)

Purpose

This command restarts the cache function after a HOLD_GCL_CACHE command has put the function on hold. The cache function is interrupted to allow the system to make swappable the segments used by the cache function. The segments in question are those which are RESIDENT when the cache function is operational.

The RELEASE_GCL_CACHE command is used once the above-mentioned segments have been made swappable.

Syntax

```
{ RELEASE_GCL_CACHE }  
{ RGC }
```

Parameters

None.

Rules

None.



8.4.3 RELEASE_JOB (RJ)

Purpose

Cancels the effect of the HOLD_JOB command, putting HELD jobs in the IN SCHEDULING state, and SUSPENDED jobs in the EXECUTING state. A job may only be released by its submitter, by its USER and by main or station operators.

Syntax

```
{RELEASE_JOB}
{RJ

    [RONS = (Xnnnn [Xnnnn] ...)]

    [{SELECT} {jclass} [ {usernm} [ {projnm}]]]
    [{      } = {      } [ / {      } [ / {      }]]]
    [{SEL   } { * } [ { * } [ { * }]]]

    [STRONG = {bool|0}]

    [{SYSTEM}      ]
    [{      } = name4]
    [{SYS   }      ]
```

Parameters

RONS	a list of up to 16 Run Occurrence Numbers of the jobs to be released.
SELECT	restricts the scope of the command to the jobs that belong to the specified combination of jclass/user-name/project. SELECT=* releases all held or suspended jobs accessible by terminal user.
STRONG	if 1, jobs are released regardless of their hold count values (see <i>ENTER_JOB_REQ</i> command for this feature).
SYSTEM	This parameter indicates the system where the command is executed.



Rules

- RONS and SELECT are mutually exclusive.
- When neither RON nor SELECT is specified, SELECT=* is assumed and all suspended jobs accessible by terminal user are released.

Examples

RJ	{release all held jobs submitted by the user}
RJ (X123 X456)	{release two jobs}
RJ X123 STRONG	{release job, regardless of hold count value}
RJ SL=AA	{release all AA class jobs}
RJ SL=*/JONES	{release all jobs submitted under the user-name JONES}



8.4.4 RELEASE_OUTPUT (RO)

Purpose

Cancels the effect of a previous HOLD_OUTPUT command by restarting the printing of suspended outputs or making held outputs eligible for printing. This command is also used to cancel the effect of the WHEN clause of the DPRINT or MODIFY_OUTPUT_PARAMETERS (MDOP) commands or of the OUTi parameter group.



IMPORTANT:

This command applies only to the outputs on which the submitter has access rights

Syntax

```
{ RELEASE_OUTPUT      }
{ RO                  }

[ { output14 [ output14 ] ... } ]
[ OUTPUT = { output14-output14 } ]
[ { * | ALL           } ]

[ SELCLASS = a [-a] ]

[ STRONG = { bool | 0 } ]

[ { STATION } { * } ]
[ {         } = { ALL } ]
[ { ST      } { name8 } ]

[ { SELECT } { user/project } ]
[ {         } = { */project } ]
[ {         } { */* } ]
[ { SEL    } { * } ]

[ { JOBNAME } ]
[ {         } = name8 ]
[ { JNM     } ]

[ { SYSTEM } ]
[ {         } = name4 ]
[ { SYS    } ]
```




STATION	restricts the scope of the command to outputs attached to the station specified. When omitted or when * or ALL is specified, there is no station restriction.
SELECT	restricts the scope of the command to outputs belonging to a specified user and/or project. Not meaningful when OUTPUT specifies a list of outputs (format 1). By default, only the outputs belonging to the command submitter are selected.
JOBNAME	restricts the scope of the command to the outputs of jobs with the name specified. If JOBNAME is specified, the default value for OUTPUT is *.
SYSTEM	indicates the system on which the command is to be executed.

Rules

- If several outputs of the same job are given the same name, RO Xnnnn :name only affects the first output with that name encountered by the system.
- SELCLASS is ignored for names of specific outputs (i.e. Xnnnn:nnnn or Xnnnn:name).

Examples

RO X23	{release all held outputs of job X23}
RO X23 Z	{release all held outputs of job X23 that are of class Z}
RO X23 A-I	{release all held outputs of job X23 that are of classes A to I}
RO X23 :2	{release specified outputs of job X23}
RO X23 :PUNCH	
RO (X23 :2 X23 :PR)	
RO X23 STRONG	{release all outputs of job X23, bypassing all WHEN clauses. Also enqueues the READY outputs}
RO X123-X2167	{release all outputs, belonging to the command submitter, of the jobs whose RON is in the range X123 to X2167}
RO * Z	{release all class Z outputs, belonging to the command submitter, of all jobs}



- RO * SELECT=U1/P1 {release all the outputs belonging to user U1 of project P1}
- RO * STATION=BC10 {release all outputs, belonging to the command submitter, that are attached to the station BC10}
- RO X123-X2167 A ST=BC10 SEL=U1/P1
{release all class A outputs of jobs in the range X123 to X2167 that are attached to station BC10 and belong to user U1 of project P1}
- RO JOBNAME=JOB1 {release all outputs, belonging to the command submitter, of the jobs named JOB1}



8.4.5 RELEASE_TERMINAL_OUTPUT (RTO)

Purpose

Cancels the effect of the HOLD_TERMINAL_OUTPUT command by making a held terminal output eligible for printing, or by restarting the printing of a suspended or frozen terminal output. Refer to the *GTWriter User Guide* for a complete description of the Generalized Terminal Writer.

Syntax

```
{ RELEASE_TERMINAL_OUTPUT }
{ RTO }

OUTPUT = ( Rnnnn [ Rnnnn ] ... )

[          {+dec5} ]
[          {-dec5} ]
[ FROM = { dec5 } ]
[          { CUR } ]
[          { BEG } ]

[STRONG = { bool | 0 } ]

[BANNER = { bool | 1 } ]
```

Parameters

OUTPUT	a list of up to 16 names of outputs to be released. A Generalized Terminal Writer output name has the form Rnnnn and is provided as the result of the DPRINT command (with TWDEST parameter).
FROM	(for a frozen output only) the page where printing is to restart expressed as a number of pages backwards (-nn), forwards (+nn), the current page (CUR, the default value), or the beginning of the output (BEG).
STRONG	if 1, the output is considered as being frozen; if 0 (default), it is considered as being held.
BANNER	(for a frozen output only) if 0, the banner at the beginning of the report is not printed; if 1 (default), it is left as it is.



Rules

- FROM may be used only if STRONG is 1.
- BANNER may be used only if STRONG is 1.
- FROM = -nn cannot restart printing more than 8 pages back.

Examples

RTO R1234 {release the held output with name R1234}

RTO R2345 STRONG FROM=BEG
{restart the frozen output R2345 from the beginning}



8.4.6 RESYNC_MIRROR_SHARED (RSCMIR)

Purpose

Re-synchronizes on system B all shared mirrored disks after a crash on system A, A and B being coupled through mirrored disks in dynamic sharing.

This command is available only if the system has been configured with MIRROR = YES.

Syntax

```
{ RESYNC_MIRROR_SHARED }  
{ RSCMIR }
```

Parameters

None.

Rules

RSCMIR is to be used in coupled systems environment with dynamic sharing of volumes after a crash on one of the two coupled systems.

RSCMIR is mandatory prior to MDV CLEAR on any MIRROR SHARED volume.

Refer to *Mirror Disks User's Guide*.

Example

```
RSCMIR displays the messages:  
  
MR02 RESYNCHRONIZATION OF volnm1 dvc  
      SUCCESSFUL  
MR02 RESYNCHRONIZATION of volnm2 dvc  
      SUCCESSFUL  
MR04 RESYNCHRONIZATION OF SHARED  
      VOLUMESCOMPLETED
```



8.4.7 RELEASE_USER_REQ (RUR)

Purpose

Cancels the effects of the HOLD_USER_REQ command by making file transfer or jobset requests eligible for execution.

Syntax

```
{ RELEASE_USER_REQ }  
{ RUR }
```

```
REQUEST = (name8 [name8] ...)
```

```
[START = dec5]
```

Parameters

REQUEST a list of up to 16 names of requests to be released. Each name may take one of the following three forms:

- Jnnnnnnn: a Jobset Occurrence Number (JON)
- Fnnnnnnn: a File transfer Occurrence number (FON)
- name8: the symbolic name of a request, as specified by the NAME parameter of the ENTER_JOB_REQ or ENTER_FILETRANS_REQ commands.

JONs and FONs are provided as results of the ENTER_JOB_REQ and ENTER_FILETRANS_REQ commands. They are unique identifiers for the corresponding requests.

START (meaningful only for file transfer requests) the rank of the logical record (starting with 1) from which the transfer is to be resumed.



Rules

START is ignored for jobset requests.

Examples

RUR J66 {release one jobset}

RUR (F4 J7 FREQ) {release three requests}

RUR F33 START=600 {release file transfer and restart at record 600}



8.4.8 REPLY (REP)

Purpose

Provides a reply to a deferred reply question.

Syntax

```
{ REPLY }
{ REP   }

      QID = dec2

      [ { MSSG }           ]
      [ {           } = char160 ]
      [ { MSG   }           ]

      [ { SYSTEM }         ]
      [ {           } = name4 ]
      [ { SYS   }           ]
```

Parameters

QID	the identifier of the question to which the reply applies. It is the two-digit number which is set in front of the text of the question.
MSSG	the text of the reply; when omitted, a blank reply is assumed.
SYSTEM	This parameter indicates the system where the command is executed.

Rules

QID must be the identifier of a deferred reply question that was addressed to the terminal user (not to another person) and that has not yet been replied to.

Examples

```
REPLY 12 YES           {reply to question 12}
REP 13 'A=3 , B=7 , k=8 '
                        {reply to question 13}
REP 15                 {reply blank to question 15}
```



8.5 "S" Commands

8.5.1 SEND

Purpose

Directs a message to a user's mailbox, to the log file, or to the terminal.

Syntax

SEND

```
{ MSSG }  
{      } = char105  
{ MSG  }
```

```
[      { name12 } ]  
[ DEST = { ME    } ]  
[      { LOG    } ]
```

Parameters

MSSG the text of the message to be forwarded.

DEST the destination of the message is as follows:

name: the name of the person to whose mailbox the message is directed. The person may be currently logged on to the system or not. If no such person is registered in the system catalog, ME is assumed.

ME: the message is printed on the terminal.

LOG: the message is registered in the IOF log file.



Rules

None.

Examples

```
SEND 'Hello Joe !' JSMITH  
      {Send message to user JSMITH.}
```

```
SEND OK      {Write on terminal.}
```

```
SEND 'POINT 3' LOG {Write to log file.}
```



8.5.2 SEND_ERROR_LOG (SNDERLOG)

Purpose

Enters a text in the SYS.ERLOG file.

Syntax

```
{ SEND_ERROR_LOG }  
{ SNDERLOG      }  
  
      { MSSG }  
      {      } = char255  
      { MSG  }
```

Parameters

MSSG the text of the message to be entered.

Rules

None.

Example

SNDERLOG 'all disk counters reset after repair'

NOTE:

The message can be retrieved later by a DISPLAY_ERROR_LOG command.



8.5.3 START_ARS (SARS)

Purpose

Initializes and enables the ASM 7 File Migrator (ARS).

Syntax

```
{ START_ARS }  
{           } ;  
{ SARS     }
```

Parameters

None.

Example

```
SARS;
```

Activates the ARS facility.



8.5.4 START_CLC (SCLC)

Purpose

Starts the Cartridge Library Client (CLC) component

The CLC contributes to the automation of the mount/dismount operations for cartridge tapes. It intercepts requests for library cartridge services and transmits them to the Cartridge Library Server (CLS).

The CLC and CLS are connected by a communications link. The CLC becomes operational when the link with the CLS is established. The CLC, CLS and Telecommunications Server may be activated in any order.

Mount requests issued before the CLC is activated or during its initialization are automatically processed when the CLC becomes operational.

Syntax

```
{ START_CLC }  
{           }  
{ SCLC     }
```

Parameters

None.

Rules

None.

Example

```
SCLC           {Start the Cartridge Library Client component.}
```



8.5.5 START_CLX (SCLX)

Purpose

Activates the CLX service.

Syntax

```
{ START_CLX }  
{ SCLX      }
```

LIBRARY = <library name>

SUBFILE = <subfile name>

Parameters

LIBRARY name of the GCOS 7 source library containing the CLX configuration file. The default value is CLX.SLLIB.

SUBFILE name of the subfile containing the CLX configuration file. The default value is CLX_CONFIG.

Example

S: START_CLX



8.5.6 START_CMSR (SCMSR)

Purpose

Starts or restarts a CMSR (Complex Management Server) that was not started at initialization, or that has aborted or was terminated by the `TERMINATE_CMSR` command. Note that CMSR is normally started automatically by `JOBM` at GCOS 7 initialization if the ISL option HA is active. This command is a part of the High Availability product (HA). For more information, see the *High Availability Administrator's Guide*.

Syntax

```
{ START_CMSR }  
{ SCMSR      }
```

Parameters

None.

Rules

The CMSR must have been terminated

The GCOS 7 system on which the CMSR is to run must have been started.

This command must be submitted on the member where the CMSR is to run.

If the CMSR was stopped by the `TERMINATE_CMSR` command with the `FORCE` parameter set, the servers that were running restart automatically after a `START_MEMBER` command.

Example

```
SCMSR {start or restart the CMSR that was terminated.}
```



8.5.7 START_CONSOLE_LOG (SCLOG)

Purpose

Starts or restarts the logging of the operator console into the SYS.LOGC file. This command is to be used only under Service Center supervision.

Syntax

```
-----  
  
{ START_CONSOLE_LOG }  
{ SCLOG }  
  
[CLEAR = { bool | 0 } ]
```

Parameters

CLEAR If 1, the SYS.LOGC file is cleared before the logging is restarted.
 The default value is 0.

Rules

Two SCLOG commands must not be executed simultaneously from two consoles. The SCLOG CLEAR command must not be executed while the PRLOGC utility is being executed.

Examples

SCLOG {Restarts the logging of the operator console without erasing the SYS.LOGC file contents.}

SCLOG CLEAR {Erases the contents of the SYS.LOGC file and restarts the logging.}



8.5.8 START_DEVICE (SDV)

Purpose

Indicates to the system that a disk drive or cartridge device has been set to WRITE PERMIT as a reply to a request to do so.

Syntax

```
{ START_DEVICE }
{ SDV           }

      { DEVICE } = { MSxx }
      {         } = { CTxx }
      { DV      } = {       }
```

Parameters

DEVICE the identification of the disk drive or cartridge device that has just been set to WRITE PERMIT. The activity which was temporarily blocked because the disk or cartridge was in WRITE PROTECT is resumed.

Rules

None.

Examples

```
SDV MS03                    {Disk drive MS03 has been set to WRITE PERMIT.}
SDV MS11                    {Disk drive MS11 has been set to WRITE PERMIT.}
SDV CTB1                    {Cartridge unit CTB1 has been set to WRITE
PERMIT.}
```



8.5.9 START_DEVICE_TRACE (SDVTR)

Purpose

Asks for the logging in the SYS.ERLOG file and/or the displaying of the AV17 debugging message (See *Note 5* below) each time an event occurs for a device, a range of devices or devices of a certain class. The effect of this command can be reversed by the TERMINATE_DEVICE_TRACE command.

Syntax

```
{ START_DEVICE_TRACE }
{ SDVTR                }

[ { DEVICE } { *          } ]
[ {         } = { dvtyp [dvnm] } ]
[ { DV      } { dvc       } ]

[          { WARN      } ]
[ LOGLEVEL = {         } ]
[          { DETAIL    } ]

[ { DISPLAY_ATN }          ]
[ {             } = bool ]
[ { DATN        }          ]

[ { DISPLAY_WARN }          ]
[ {             } = bool ]
[ { DWARN       }          ]

[ { DISPLAY_ABN }          ]
[ {             } = bool ]
[ { DABN        }          ]
```

Parameters

DEVICE the device(s) for which the trace is to be started, expressed as:

* all devices,
dvnm a specified device,
dvtyp all devices of a type,
dvc all devices of a class.

When omitted, all devices (*) are assumed.



LOGLEVEL	<p>specifies the types of events to be logged in the SYS.ERLOG file.</p> <p>If WARN, only I/O errors and warnings are logged.</p> <p>If DETAILED, error logging is forcibly started each time peripheral error handling software is called.</p> <p>When this parameter is omitted, the current level of logging is not modified;</p> <p>If a previous SDVTR command was issued with LOGLEVEL=DETAILED option, that option is retained and is unchanged by the TDVTR command,</p>
DISPLAY_ATN	if 1, logged events that are attentions are displayed.
DISPLAY_WARN	if 1, logged events that are normal terminations with marginal conditions are displayed.
DISPLAY_ABN	if 1, logged events that are abnormal terminations are displayed.

Rules

When no parameter or only DEVICE is specified, then DISPLAY_ATN=1 and DISPLAY_ABN=1 are assumed.

Examples

SDVTR	{Start displaying ATN and ABN for all devices.}
SDVTR MT	{Start displaying ATN and ABN for all tapes.}
SDVTR MS/B10	{Start displaying ATN and ABN for all MS/B10 disks.}
SDVTR MT03	{Start displaying ATN and ABN for tape unit MT03.}
SDVTR DWARN	{Start displaying WARN for all devices.}
SDVTR MT04 DWARN	{Start displaying WARN for tape unit MT04.}
SDVTR MS DETAILED DABN=0	{Force error logging for all disk devices without any display.}



NOTES:

1. The display options (DATN, DABN, DWARN) of the SDVTR command are cumulative; their effect is cancelled by the corresponding TDVTR command only.
2. Display options are recorded in the SYS.ERLOG file for each device and remain in effect from one GCOS 7 session to the next. However, the LOGLEVEL is always reset to standard at the end of a GCOS 7 session, whether the termination is normal or abnormal.
3. Display options, especially with the LOGLEVEL=DETAILED option, may cause an overload of messages; it is advisable to issue a SDVTR DABN=0 command to stop these message displays and to retrieve logged events later either by using the DISPLAY_ERLOG command or by running the PRLOG utility.
4. Using LOGLEVEL=DETAILED for numerous devices may cause the SYS.ERLOG file to be saturated early in the GCOS 7 session; this causes the first events logged to be lost. It is recommended to change this LOGLEVEL=DETAILED option as soon as the necessary information is available to identify the existing problem.
5. The AV17 message is specifically for the purpose of debugging. It is only sent to the main console that activated the trace. The debugging information in this message is recorded in SYS.ERLOG and not SYS.LOGC. The information can be recovered from SYS.ERLOG with the DERLOG command or the PRLOG utility.



8.5.10 START_DYNSAVE_SVR (SDYNSVR)

Purpose

Starts the S-class dynamic save service job on the multi-process server.

Syntax

```
{ START_DYNSAVE_SVR }  
{ SDYNSVR }
```

Parameters

None.

Rules

It may be refused with the message :

```
"A TERMINATE OR START COMMAND IS ALREADY ACTIVE"
```

if it is launched at the same time as the commands indicated by the message (TDYNSVR, SDYNSVR).

Exceptionally an abnormal state of type 0 resources may produce the reply:

```
"TYPE 0 RESOURCES FOR DYNSAV NOT CREATED"
```

or

```
"TURBOCONFIG.DYNSERV0=value"
```

```
"NOT READY TO ACTIVATE SERVER (DYNSERV0 NOT=01) "
```

Example

```
SDYNSVR           {Requests start of the S-class dynamic save service job  
                  on the multi-process server.}
```



8.5.11 START_ERROR_LOG (SERLOG)

Purpose

Starts logging events in the SYS.ERLOG file. This command is to be used only under Service Center supervision.

Syntax

```
-----  
  
{ START_ERROR_LOG }  
{ SERLOG           }  
  
    [ CLEAR = bool ]
```

Parameters

CLEAR if 1, the SYS.ERLOG file is cleared before the logging is restarted.

Rules

None.

Examples

```
SERLOG                      {Restart logging.}  
  
SERLOG CLEAR                {Restart logging after clearing SYS.ERLOG.}
```




8.5.12 START_INPUT_READER (SIR)

Purpose

Specifies that job readers started after an EJR command, remain active during the whole session. This saves time when jobs are started. The command acts as a flip/flop with the TERMINATE_INPUT_READER command. (If you want to modify the number of readers on the site, use MODIFY_DJP command).

Syntax

```
{ START_INPUT_READER }  
{ SIR }
```

PERM = bool

Parameters

PERM This parameter must be set for the command to be effective.

Rules

None.

Example

SIR PERM {After this command is entered, any job reader started by an EJR command remains active.}



8.5.13 START_IO_CACHE (SIOC)

Purpose

This command starts the GCOS 7 disk I/O Cache facility. It remains active until a `TERMINATE_IO_CACHE` or a system shutdown occurs. See the *Large Memory Cache User's Guide* for details.

Syntax

```

{ START_IO_CACHE }
{ SIOC           }

[ { SIZE } = { 4 } ]
[ {      } = {      } ]
[ { SZ  } = { dec4 } ]

[ SILENT = { bool | 0 } ]

[ { MODE } ]
[ {      } = { CACHE | SIMUL | REGUL } ]
[ { MD  } ]

[ { DEFAULT } ]
[ {      } = { FORCE | BYPASS } ]
[ { DFLT  } ]

```

Parameters

SIZE Initial cache size in megabytes. The minimum size is 4 Mbytes or the size of the dedicated I/O cache memory (if this exists). The maximum size depends on the DPS 7000 model and is computed as follows:

$$80\% \times (\text{TOTAL SIZE} - \text{DEDICATED I/O CACHE SIZE}) - 20 \text{ Mbytes } (*) + \text{DEDICATED I/O CACHE SIZE}$$

(*) 20 Mbytes is an average size for GCOS 7 systems.

If the MIRROR DISKS function is active, the I/O Server service job is already started with a 4 MB cache memory. This parameter may be used to increase the cache size.

SILENT If 0, the information and the error messages are sent. If 1, only error messages are sent.



MODE	Specifies the operating mode: If CACHE (default), the cache memory is physically allocated and normal cache processing begins. If SIMUL, cache processing is simulated but cache memory is not physically allocated. If REGUL, files with low hit ratios are dynamically excluded from the cache processing.
DEFAULT	When DEFAULT=FORCE (default) all files are cached, except those whose catalog descriptions or file assignment parameters specify otherwise. When DEFAULT=BYPASS no file is cached unless its catalog description or file assignment parameters specify otherwise.

Rules

This command is reserved for the main operator or the system administrator.

Examples

SIOC {Start the GCOS 7 I/O Cache facility with an initial size of 4 Mbytes}

Dedicated cache not supported:

SIOC SZ=12 {Start the cache facility with an initial size of 12 Mbytes}

Dedicated cache of 256 Mbytes supported:

SIOC SZ=32 {Start the cache facility with an initial size of 256 Mbytes (all in dedicated cache memory).}

SIOC SZ=300 {Start the cache facility with an initial size of 300 Mbytes (256 Mbytes in dedicated cache memory and 44 Mbytes in standard memory).}



8.5.14 START_LOAD (SLD)

Purpose

Enables jobs of specified classes to be selected for execution, creates new job classes, starts/stops job tracing.

Syntax

```

{START_LOAD}
{SLD          }

  [{CLASSLIST}  { *          }]
  [{           } = {         }]
  [{CLASS      }  {aa [-aa] }]

[JTRACE = {bool|0}]

  [{MULTI_LEVEL}          ]
  [{           } = dec3]
  [{MULTI       }          ]

  [{PRIORITY}  ( {           *           } {           *           } )]
  [{           } = ( {0|1|2|3|4|5|6|7}, {0|1|2|3|4|5|6|7|8|9} )]
  [{PRTY      }  ( {           } {           } )]

  [{PRIORITY_LIMIT}  ( {           *           } {           *           } )]
  [{           } = ( {0|1|2|3|4|5|6|7}, {0|1|2|3|4|5|6|7|8|9} )]
  [{PRTYLIM      }  ( {           } {           } )]

[MDMULTI = bool]

  [{SYSTEM}          ]
  [{           } = name4]
  [{SYS            }  ]

```

Parameters

CLASSLIST the class or range of classes to start, expressed as:

- * = all classes
- aa = a specified class
- aa [-aa] = a specified, inclusive range of classes

JTRACE If 1, every step of every job is traced.



MULTI_LEVEL	multiprogramming level for each one of the selected classes. The range is from 0 to 255.
PRIORITY	Priority for the selected class(es) expressed as a pair of digits. There are (spr) scheduling priority and (xpr) execution priority. The lower the number, the higher the urgency. If * (default) the system assigns a default value to the corresponding priority.
PRIORITY_LIMIT	sets limits to PRIORITY values, expressed as a pair of digits as follows: spr limit, xpr limit If * (default), the system assumes PRIORITY values.
MDMULTI	If 1, enables MULTI_LEVEL to be modified later. If 0, MULTI_LEVEL cannot be modified.
SYSTEM	This parameter indicates the system where the command is executed.

Rules

- CLASSLIST or JTRACE=1 must be used, but not simultaneously.
- MULTI_LEVEL, PRIORITY, PRIORITY_LIMIT and MDMULTI are only meaningful when CLASSLIST=aa [-aa].
- When CLASSLIST is not used and JTRACE=0, the command is ignored.
- When a new CLASS is created you must issue a second SLD command to start it, since new classes are in the NSTART state (see example below).

Examples

SLD *	{Start all classes.}
SLD AA MULTI=2 PRTY=(2,2) MDMULTI SLD AA	{Create class AA, then start it.}
SLD AA-QZ	{Start classes AA to QZ. The classes were created by a previous SLD command.}
SLD JTRACE	{Start job tracing.}



8.5.15 START_MEMBER (SMB)

Purpose

This command starts a single Member, several Members or the whole Complex. This command is a part of the High Availability product (HA). For more information, see the *High Availability Administrator's Guide*.

Syntax

```
{ START_MEMBER }  
{ SMB          }  
  
    { MEMBER } = { *          }  
    { MB     } = { ( name4 [ name4 ] ) }
```

Parameters

MEMBER List of the MEMBERS to be started; the star (*) starts all the MEMBERS of the Complex.

Rules

- The CMSR of the MEMBER on which this command is entered must be running.
- This command cannot be submitted between the messages "CMSR RESYNCHRONIZATION IN PROGRESS" and "END OF CMSR RESYNCHRONIZATION".
- The MEMBERS must be described in the COMPLEX description.
- The MEMBERS must have been terminated.
- The GCOS 7 system on which the Member is mapped must be started.
- The CMSR of the MEMBER on which this command is entered must be able to communicate with the CMSR of the MEMBER to be started, i.e. no telecom failure or crash.

Examples

```
SMB  MB=*          {Start the whole COMPLEX.}  
  
SMB  MB=MB1       {Start the MEMBER named MB1.}
```



8.5.16 START_OUTPUT_WRITER (SOW) (SO)

Purpose

Starts a writer on a specified device or file for specified output classes. When used by a station operator, this command starts a remote writer which sends to a station the outputs queued for it.

Syntax

```
{ START_OUTPUT_WRITER }
{ SOW }
{ SO }

      { ALLDV }
      { PRxx }
ON = { name4 }
      { $site:SPOOL }

[ SELCLASS = (a[-a] [a[-a] ...]) ]

[ { DEVCLASS } { } ]
[ { } = { PR [/dvattr] } ]
[ { DVC } { } ]

{ STATION }
{ } = stnm
{ ST }

[ PERM = bool ]

[ { DOF } ]
[ { SARF } ]
[ DATAFORM = { VPF } ]
[ { ASA } ]
[ { SSF } ]

[ LCASE = bool ]

[ USER = user [ PASSWORD = password ] ]

[ PROJECT = project ]

[ BILLING = billing ]
```



```
[ JOR = NORMAL | ABORT | NO ]  
  
[ BANNER_WIDTH = banner_width ]  
  
[ SILENT = bool ]  
  
[OPTIMIZE = bool ]
```

Parameters

ON on which device or file the writer is to be started, expressed as:

- ALLDV: all devices
- PRxx: the named printer
- name4: the station's logical device identification
- \$site: SPOOL the standard SYSOUT file of the named site
- rfile78: the named local or remote (GCOS 7 or DPS 6) file.

rfile78 defines a local or remote file name. Syntax is: [\$SITE:] file-name. When defining a foreign system file (DPS 6), the file-name must be protected by '. When defining a DPS 7000 file, the file-name format is: efn [..subfile][:media:dvc]. As file-name is not a file-literal, \$CAT/\$RES or \$MFT, etc., are not supported.

An operator responsible for a queued writer (see the *Generalized Terminal Writer User's Guide*) can arrange for automatic submission of a job at the end of output processing (when the subfile in the library is closed). This job is conventionally called JCL_CONTROL and must be present in the library.

For remote operations refer to the *DPS6 - DPS7 Remote Facilities Manual* or the *DJP User's Guide* or the *RBF User's Guide*.

If you are using a library, this library must not be cataloged in SITE.CATALOG.



SELCLASS	restricts the scope of the command to the outputs that belong to the specified class(es) or range(s) of classes. This may be expressed as: class for a single class class-class for a range of classes where class is a letter (A to Z). When SELCLASS is omitted, "default classes" attached to the device are assumed. In the case of SOW on SPOOL or rfile78, "all classes" are assumed.
DEVCLASS	(for sequential files) the device to be simulated for presentation of the report on the file. The file can be later processed by use of the DPRINT command.
STATION	Name of the (unattended or rescued) station to which the command applies.
PERM	(applies only to local writers) when 1, the writer is permanent. The default value is 0.
DATAFORM	The data format for the output handled by the writer. One of the following can be entered: DOF Device Oriented Format SARF Standard Access Record Format VPF Virtual Printed Format ASA American Standard Assn. format SSF Standard System Format This parameter is meaningful only for the Writer Queued product.
LCASE	(for DPS 6 stations) when 1, the remote writer is requested to keep lowercase when printing.
USER	Name of the user to be connected to the remote site. Default is the command submitter.
PASSWORD	Password of the user to be connected to the remote site. Default is the password of the command submitter.



PROJECT	<p>Project of the user to be connected to the remote site. Default is the project of the command submitter.</p> <p>If BILLING only is specified (that is, USER / PASSWORD / PROJECT not specified), PROJECT is left blank to enable connection to a remote site with the default project of the user on the remote site (EFTR rules).</p>
BILLING	<p>Billing of the user to be connected to the remote site. Default is the billing of the command submitter.</p> <p>If PROJECT only is specified (that is, USER / PASSWORD / BILLING not specified), BILLING is left blank to enable connection to a remote site with the default billing of the specified project on the remote site (EFTR rules).</p>
JOR	<p>Specifies whether the JOR is to be kept or not on termination of the writer.</p>
NORMAL	the JOR is kept in all cases,
ABORT	the JOR is kept only in the event of abnormal termination,
NO	the JOR is not kept in any circumstances.
	<p>The default value depends on the type of writer: Local magnetic writers: NORMAL Other writers: NO.</p> <p>This parameter is meaningless for SPOOL.</p>
BANNER_WIDTH	<p>Specifies the banner width (in number of columns) of output when it is printed. The default and upper value is 120; the lower value is 55.</p> <p>The BANNER_WIDTH parameter value in the SOW command takes into account the possible characteristics of the banner as follows:</p> <ul style="list-style-type: none">– small banner if BANNER_WIDTH is between 55 and 107,– big banner with big letters separated by 1 blank if BANNER_WIDTH is between 107 and 118,– big banner with big letters separated by 2 blanks if BANNER_WIDTH is between 118 and 120.



The BANNER_WIDTH parameter value in the SOW command also takes into account the format of the paper mounted on the printer. If the paper cannot accommodate the requested format, the printed banner width is the maximum possible.

SILENT

This parameter deletes the messages TW04, TW05 and TW06 sent by GT-WRITER. This parameter is significant only in the case of a queued writer launched on the station GTW. The default value is SILENT = 0.
Example

```
SOW $MAIN :library STATION=GTW
SILENT=1 .
```

OPTIMIZE

The implementation of this parameter modifies the selection mechanism of the outputs by the writer. The outputs are still selected according to their priority, but within a priority, the writer will select first the output which does not require the mount of a new media. The default value is OPTIMIZE = 0. In this case, the outputs are selected by priority and FIFO within the priority.
Example:

```
SOW PR01 SELCLASS=P OPTIMIZE=1 .
```

Rules

- DEVCLASS, PERM, LCASE and DATAFORM are ignored when ON is ALLDV.
- SITE is mandatory in the ON parameter if ON = RFILE and STATION is not the same as #WSTATION.
- When SITE is omitted in the rfile78 syntax, the default site of the station is assumed.
- SITE is mandatory in the rfile78 syntax if the SOW command is managing an output queue other than the default station queue.
- USER, PASSWORD, PROJECT and BILLING are meaningful only for remote writers or SPOOL.
- SOW command on an already started file can be used to add new classes (SELCLASS parameter) to the class list on which the writer has been launched. All parameters other than SELCLASS are ignored (and keep their initial values).



- In the case of SOW..SPOOL, selclass/devclass/STATION queue are added to the GTP selection criteria. Only the operator's right to submit SOW for that STATION is checked.
- If a file situated on a remote site is used, this file must be sequential, and its RECSIZE must be greater than or equal to 600.
- When a writer is started on a cataloged file, this file must pre-exist. The output writer will not create it.

Examples

SOW ALLDV {Starts all local writers on devices.}

SOW PRL6 Z LCASE {Start a writer on the station's logical device PRL6
 with lower case handling.}

SOW OUTSAV :OUT :MT/T9 X DVC=PR/H160
 {All outputs of class X are saved on the named tape
 file, emulating a 160 hammers printer.}

SOW \$SETI:MYFILE {Launches a Writer on the site SETI; the outputs are
 saved in the file MYFILE.}

SOW \$BC03:SPOOL ST=BC03 USER=OP2 PASSWORD=OP
 {Starts a spool towards the remote site BC03; user
 OP2 with password OP is connected to this remote
 site.}

SOW PR01 BANNER_WIDTH=110
 The banner width of the outputs processed by this
 writer is 110 columns.



8.5.17 START_POWER_TIMER (SPWRTM)

Purpose

Starts the timer power control module (H_TPC).

Syntax

```
{ START_POWER_TIMER }  
{ SPWRTM }
```

Parameters

None.

Rules

This command is not available on DPS 7000/2X0/3X0.

Example

```
SPWRTM           {Requests start of the power timer.}
```



8.5.18 START_RMS (SRMS)

Purpose

Enables the connection of a remote maintenance operator.

Before this command is entered, the RMS operator's logon is submitted to a main operator for approval.

Syntax

```
{ START_RMS }  
{ SRMS      }  
  
[ ON = { GCOS | SP } ]  
  
[ { ENABLE | ENBL } = { PL | LC } ]
```

Parameters

ON	Specifies the RMS type: GCOS functions on all DPS 7000 machines. SP indicates the RMS supported by the Service Processors DPS 7000/An and the DPS 7000/5x0/7xx/8x0/Dx0/Mx0 machines.
ENABLE	Designates the PL or the LC object when connecting under SP RMS: LC means the Logical RMS Connection. It is the default for the DPS 7000/2x0/3x0/5x0/7xx/8x0/Dx0/Mx0 machines. PL means the Physical RMS Line. It is the default for the DPS 7000/An. On the DPS 7000/5x0/7xx/8x0/Dx0/Mx0, the default object when connected through the Service Processor (SP), is LC. The LOCK parameter can be used to change the object to PL.



Rules

- Except as explained above for the DPS 7000/5x0/7xx/8x0/Dx0/Mx0, an incompatible object causes the command to be rejected.
- When ON=GCOS, if specified, ENABLE must be equal to LC.

Examples

SRMS {Starts a GCOS type RMS session.}

SRMS ON=SP ENABLE=PL

{Starts RMS giving the physical line (PL) the state ENBL (ENABLE) for the DPS 7000/An and the DPS 7000/5x0/7xx/8x0/Dx0/Mx0. The conditions are different for the two machines:

- DPS 7000/An: the ENBL state is said to be LIMITED; it returns to state LOCK in 5 minutes.
- DPS 7000/7xx/8x0/Dx0/Mx0: The ENBL state is unlimited; there is not time-out.

The command is rejected if the PL state is ENBL and if the LC state is WAIT or USED.}

SRMS ON=SP ENABLE=LC

{This command, used on the DPS 7000/5x0/7xx/8x0/Dx0/Mx0, changes the LC state from WAIT to USED and validates the RMS session.}



8.5.19 START_SERVICE (SSRV)

Purpose

Starts a SERVICE and also its USED_SERVICES if they are not already started. This command is a part of the High Availability product (HA). For more information, see the *High Availability Administrator's Guide*.

Syntax

```
{ START_SERVICE }
{ SSRV           }

      { SERVICE }
      {         } = name8
      { SRV     }

RESYNC = { bool | 0 }

[ { OPTION }           ]
[ {         } = char240 ]
[ { OPT    }           ]

[ { MEMBER }           ]
[ {         } = ( name4 [ name4 ] ) ]
[ { MB     }           ]

[ { USED_SERVICE }           ]
[ {             } = ( name8 [ name8 ] ... ) ]
[ { USRV       }           ]
```

Parameters

SERVICE	Name of the SERVICE to be started.
RESYNC	Used to restart SERVICES in back up mode which had been started on the corresponding MEMBER during failure of the current MEMBER. To use RESYNC the following conditions must be met: <ul style="list-style-type: none"> – the corresponding MEMBER must be crashed, silent or telecom isolated. – the SERVICE to start must be HA.



- the BACKUP SERVER is started on the current MEMBER, which means, the MEMBER parameter, if it exists, must specify first the corresponding member, then the current member.

OPTION	A string of parameters specific to the type of SERVICE (TDS, ORACLE).
MEMBER	<p>List of the MEMBERS on which the SERVICE must be started. Without this parameter, the SERVICE starts on the MEMBERS given in the COMPLEX generation.</p> <p>The SERVICES are started in the order given here. This is useful to be able to start the SERVICE in ACTIVE mode on a MEMBER other than the preferential ACTIVE MEMBER.</p> <p>If this parameter is given and the SERVICE to start is HA, there must be two MEMBERS listed, because an HA SERVICE must be completely started, i.e. on both MEMBERS specified on its mapping.</p>
USED_SERVICE	This parameter is used to designate specific USED_SERVICES to be started if all are not to be started. 0 to 16 services can be designated. By default, these services are declared at generation of the complex.

Rules

- The CMSR of the MEMBER on which this command is entered must be running.
- This command cannot be submitted between the messages "CMSR RESYNCHRONIZATION IN PROGRESS" and "END OF CMSR RESYNCHRONIZATION".
- The SERVICE and also the USED_SERVICES and the MEMBERS must be in the COMPLEX description.
- The specified MEMBERS must belong to the SERVICE mapping.
- The MEMBER parameter must give two MEMBERS.
- The specified USED_SERVICES must have been declared as used by the SERVICE at COMPLEX generation.
- The SERVICE must not be started.



- The MEMBER, on which this command is to start the SERVICE in ACTIVE mode, must have been started with the START_MEMBER command.
- If some of the USED_SERVICES of the SERVICE to start are already running, their ACTIVE SERVER must be running on the same MEMBER as that where the SERVICE is to start in ACTIVE mode.
- A TDS HA can be started only if all the JAS HA of the COMPLEX can be set in the SWITCHABLE mode. Therefore, if the MEMBERS are UNSWITCHABLE, a switchability modification of all the MEMBERS must be done; however, this is impossible if the "co-Member is telecom isolated or if there is already a running TDS non-HA SERVICE using a JAS SERVICE.
- If a telecom isolation occurs during the SWITCHABILITY modification, the command is abandoned.
- A TDS non-HA using a JAS HA can be started only if all the JAS HA of the COMPLEX can be set in the UNSWITCHABLE mode.
- If there is an isolation or crash of the ACTIVE MEMBER, the CMSR rejects the command.
- If one of the USED_SERVICES of this SERVICE refuses to start on the ACTIVE MEMBER, the SERVICE is not started.

Examples

```
SSRV  SRV=TDS1      {Starts the SERVICE named TDS1 on all the
                    MEMBERS specified in its description and also starts
                    all the USED_SERVICES of TDS1.}

SSRV  SRV=TDS1  MB= (MB2 MB1)
                    {Starts the SERVICE named TDS1 on all the
                    MEMBERS specified in its description, but with the
                    ACTIVE SERVER running on MB2; also starts all the
                    USED_SERVICES of TDS1.}

SSRV  SRV=TDS1  USRV=BLUE
                    {Starts the SERVICE named TDS1 on all the
                    MEMBERS specified in its description and also starts
                    only its USED_SERVICE BLUE.}
```



8.5.20 START_SSI (SSSI)

Purpose

To activate the CLX processes (ssi, acsel) on OPEN 7.

Syntax

```
{ START_SSI }  
{ SSSI }
```

Parameters

None

Rules

To use this command, OPEN 7 must be active.

Example

```
S: START_SSI  
11.16 X550 IN CLX_SSI USER=OPERATOR CLASS=P SPR=6 STATION=BCA9  
11.16 X550 STARTED CLX_SSI OPERATOR P  
JB08 X550.1 STEP H_CLX_CMDSSI XPR=8 PGID=22  
TU70 ACSEL AND SSI STARTED  
11.16 X550.1 COMPLETED CLX_SSI OPERATOR P
```



8.5.21 START_SYSTEM_TRACE (SSYSTR)

Purpose

SSYSTR starts the system trace collector service job, starts domain trace activity in defined classes, starts domain wait mode for trace event requests and links domains to named buffers.

NOTE:

This command is hidden and is to be used only under the direction of a Bull support engineer.

Syntax

```
{ START_SYSTEM_TRACE }
{ SSYSTR                }

[ { DOMAIN }      { ALL      } ]
[ { DOM      } = { *        } ]
[ { TDOM     }      { star63  } ]

[ CLASS={NOR|DET|TRM|INR|IDT|CND|DTV|ABN|ARC|TRP|TDP|ALL} ]

{ WHERE }      { FILE }
{         } = {         }
{ TO     }      { MEM  }

[ { BUFFER }      ]
[ {         } = name6 ]
[ { BFR    }      ]

[ WAIT = bool ]

[ { COLLECTOR }      ]
[ {         } = bool ]
[ { CLTR      }      ]
```



Parameters

DOMAIN	It is the name of a system trace domain or, if ALL or "*" is specified, all domains. Maximum length is 63 characters.
CLASS	It is either ALL (the default, meaning all classes) or a system trace class name. Correct values are: NOR, DET, TRM, INR, IDT, CND, DTV, ABN, ARC, TRP, and TDP.
WHERE	This specifies the required domain class state, FILE or MEM. MEM is the default.
BUFFER	It is the name of the system trace collector buffer. Maximum length is 6 characters.
WAIT	If 1, the domain trace event WAIT attribute is specified; if 0, the WAIT attribute is not specified.
COLLECTOR	If 1, the system trace collector service job is started; if 0, the service job is not started.

Rules

- The trace domain attributes WAIT, BUFFER, and CLASS, in FILE state, are meaningless before and after system trace collector execution.
- MEM and closed state are equivalent to TDP class.
- Domain start trace event in WAIT mode is effective only if the connected buffer has the WAIT attribute.
- If COLLECTOR is specified no other parameter may be specified.
- If COLLECTOR is not specified, the DOMAIN must be specified.
- CLASS, BUFFER and WAIT are mutually exclusive.
- WHERE, BUFFER and WAIT are mutually exclusive.



Examples

```
SSYSTR DOMAIN=MYTDOM BUFFER=TRCB1
      {Connect the MYTDOM trace domain to buffer
      TRCB1.}
```

```
SSYSTR DOMAIN=MYTDOM
      {Open all MYTDOM domain classes to memory trace
      events.}
```

```
SSYSTR DOMAIN=MYTDOM WAIT
      {Enable WAIT mode for trace event requests issued for
      MYTDOM domain.}
```

```
SSYSTR COLLECTOR      {Start the system trace collector service job.}
```



8.5.22 START_TERMINAL_OUTPUT (STO)

Purpose

Starts the printing of outputs on a terminal or pool of terminals; modifies the class of outputs handled by a terminal. Only the owner of the object terminal or users declared as Master Operators for the Generalized Terminal Writer may use this command. Refer to the *GTWriter User's Guide* for a complete description.

Syntax

```
-----
{ START_TERMINAL_OUTPUT }
{ STO
    { TERM }
    {      } = name12
    { TN   }
    [SELCLASS = a[-a]]
```

Parameters

TERM	the name of the terminal or pool of terminals where outputs are printed.
SELCLASS	specifies the class (SELCLASS = a) or range of classes (SELCLASS = a-a) of outputs that can be printed on the terminal or pool of terminals. If the terminal or pool is already started, the selected classes are added to the list of those that are eligible for printing on the terminal or pool.

Rules

None.

Examples

STO POOL3	{Start printing on terminal pool POOL3.}
STO TERM3 A-Q	{Start printing on terminal TERM3 for classes A through Q.}



8.5.23 START_TERMINAL_WRITER (STW)

Purpose

Starts one or all of the Generalized Terminal Writer drivers. Refer to the *GTWriter User's Guide* for a complete description of the Generalized Terminal Writer.

Syntax

```
{ START_TERMINAL_WRITER }  
{ STW  
  
    [{ ALL    }]  
    [{ name4  }]
```

Parameters

ALL	The default parameter shuts down all the drivers.
name4	This is the name of the driver to be stopped. When omitted, all drivers (ALL) are assumed.

Rules

None.

Examples

STW	{Start all drivers.}
STW TW01	{Start driver TW01.}



8.5.24 SWITCH_MIRROR (SWMIR)

Purpose

Reverses the roles of the Primary and Secondary disks of a mirror volume so that the Primary disk becomes Secondary and vice versa.

This command is asynchronous.

Syntax

```
{ SWITCH_MIRROR | SWMIR }  
    { VOL | VOLUME } = name6:dvc;
```

Parameters

VOLUME The id of the mirror volume in the form name:device-type (e.g. MIRV01:MS/FSA).

Constraints

The volume must not be allocated.



8.6 "T" Commands

8.6.1 TAKEOVER_MEMBER (TKMB)

Purpose

The operator uses this command to force the TAKEOVER of all the SERVICES of a MEMBER in order to continue execution of the TDS, JAS and Oracle applications on the other MEMBER. This command is a part of the High Availability product (HA). For more information, see the *High Availability Administrator's Guide*.

Syntax

```
{ TAKEOVER_MEMBER }  
{ TKMB }  
  
{ OPTION }  
{ OPT } = { STRONG | FORCE | WEAK }
```

Parameters

OPTION Specifies the kind of TAKEOVER.

WEAK:
The ACTIVE SERVERs running on the co-MEMBER are switched into the BACKUP mode and the BACKUP SERVERS running on the current MEMBER are switched into ACTIVE mode.

STRONG:
Induces the crash of the co-MEMBER and the BACKUP SERVERs running on the current MEMBER are switched into the ACTIVE mode. This option requires a confirmation by the operator. If the operator does not answer in the affirmative, the command is abandoned.

**FORCE:**

The operator must be sure that the co-MEMBER is no longer operational before ordering this kind of TAKEOVER. The BACKUP SERVERs running on the current MEMBER are switched into the ACTIVE mode. This option requires a confirmation by the operator. If the operator does not answer in the affirmative, the command is abandoned.

Rules

- The CMSR of the MEMBER on which the command is entered must be running.
- This command must not be submitted between reception of the "CMSR RESYNCHRONIZATION IN PROGRESS" and the "END OF CMSR RESYNCHRONIZATION" messages.
- The command must be entered on the MEMBER to be switched into the ACTIVE mode.
- If OPTION is WEAK or STRONG, the two CMSRs must be operational, the two MEMBERS must have been started with the START_MEMBER command and not be telecom isolated.
- The command is rejected if entered with WEAK or STRONG while the CMSR required the FORCE OPTION.
- If FORCE is used, the operator must check that the co-MEMBER is no longer running.
- The TAKEOVER of the current MEMBER can only be executed if the MEMBER is SWITCHABLE and there are BACKUP SERVERs running on the current MEMBER.
- If there is already a TAKEOVER in process, the command is accepted only if the OPTION is more important. STRONG overrides WEAK and FORCE overrides both.
- If there is already a TAKEOVER in process, the command is rejected if the current TAKEOVER is almost finished.
- The same kind of TAKEOVER must not already be in execution.
- TAKEOVER with STRONG or FORCE is abandoned if the disk restore processing does not succeed.
- Since TAKEOVER is for all SERVERS, if an ACTIVE or a BACKUP SERVER refuses to switch, the TAKEOVER is not given up but goes on to the other SERVICES.



- If an ACTIVE SERVER refuses to switch to the BACKUP mode, CMSR automatically terminates this SERVER using FORCE.
- If a BACKUP SERVER refuses to switch to the ACTIVE mode, then it remains BACKUP, but the CMSR, using FORCE, must terminate all the MAIN SERVERs which are using it on this MEMBER.

Example

TKMB {Switches the active SERVERS on the corresponding-MEMBER to BACKUP mode and the BACKUP SERVERS on the current-MEMBER to ACTIVE mode.}



8.6.2 TERMINATE_ARS (TARS)

Purpose

Disables the ASM 7 File Migrator (ARS). Current Migration-in requests are allowed to complete, but no new *dynamic* Migration-in requests are accepted.

Syntax

```
{ TERMINATE_ARS }  
{  
{ TARS
```

```
[ STRONG = { bool | 0 } ] ;
```

Parameters

STRONG

If 1, ARS is disabled immediately. All ASSIGNs waiting for a Migration-in to complete are aborted with the return code RC=FUNCNAV. The files concerned are restored but not assigned.

If 0 (default), all ASSIGNs waiting for a Migration-in to complete are allowed to terminate normally. The files concerned are restored and assigned.

Rules

In no case does this directive interrupt file migrations already in progress.

Examples

```
TARS ;
```

ARS is disabled. All ASSIGNs waiting for a Migration-in to complete are allowed to terminate normally.

```
TARS STRONG ;
```

ARS is disabled immediately. All ASSIGNs waiting for a Migration-in to complete are aborted with return code RC=FUNCNAV.



8.6.3 TERMINATE_CLC (TCLC)

Purpose

Terminates the Cartridge Library Client (CLC) component.

Syntax

```
{ TERMINATE_CLC }  
{ TCLC }
```

```
STRONG = { bool | 0 }
```

Parameters

STRONG	This parameter specifies the type of termination: <ul style="list-style-type: none">0 CLC termination is rejected if some devices of the cartridge library are in use.1 CLC is terminated regardless of the device state.
--------	--

Rules

None.

Example

TCLC	{Normal CLC termination.}
TCLC STRONG	{Immediate CLC termination. Pending requests from Device Management and active requests to the Cartridge Library Server (CLS) are displayed on the console.}



8.6.4 TERMINATE_CLX (TCLX)

Purpose

Shuts down CLX on GCOS 7, either non-urgently or urgently.

Syntax

```
{ TERMINATE_CLX }  
{ TCLX }  
  
[STRONG = { bool | 0 } ]  
  
[DUMP = bool ]
```

Parameters

STRONG requests an urgent shutdown.

DUMP optionally allows you to request a dump in SYS.SPDUMP.

Example

```
TERMINATE_CLX STRONG [DUMP]
```

When `TERMINATE_CLX STRONG` is entered, CLX displays the message:

```
TU08 <clx version> CLX IMMEDIATE SHUTDOWN
```

CLX then displays all interrupted requests in the following message format:

```
TU39 <clx version> DVMGT OUTSTANDING REQUESTS:  
<request> <dev><vol><type><job> <msg-stat>-<cmd> IS <cmd-stat>
```

CLX then issues the following message for all queued requests:

```
TU38 <clx version> DVMGT PENDING REQUESTS:  
<request> <dev><vol><type><job> <msg-stat>-<cmd> IS <cmd-stat>
```



8.6.5 TERMINATE_CMSR (TCMSR)

Purpose

This command terminates execution of a CMSR on a Member. It is used by the Main Operator to activate CRCXGEN (Create Complex Generation) and the ENABLE parameter equal to 1, or in an abnormal situation. This command is a part of the High Availability product (HA). For more information, see the *High Availability Administrator's Guide*.

Syntax

```
{ TERMINATE_CMSR }  
{ TCMSR           }  
  
FORCE = { bool | 0 }
```

Parameters

FORCE First causes execution of the TERMINATE_MEMBER command with the FORCE parameter set on this command also, then terminates the CMSR. In this case, CMSR termination is effective even if there were still activated components on the Member.

Rules

If there are active components on the Member, the command is rejected without the FORCE parameter.

Example

```
TCMSR {terminates CMSR execution.}
```




8.6.6 TERMINATE_CONSOLE_LOG (TCLOG)

Purpose

Makes the SYS.LOGC file unavailable to the system and terminates the logging of the current session. This command is to be used only under Service Center supervision.

Syntax

```
{ TERMINATE_CONSOLE_LOG }  
{ TCLOG }
```

Rules

Two TCLOG commands must not be executed simultaneously from two consoles. TCLOG must not be executed while the PRLOGC utility is being executed.

Example

```
TCLOG {Terminate operator session logging.}
```



8.6.7 TERMINATE_DEVICE_TRACE (TDVTR)

Purpose

Terminates the logging and/or the displaying of a message each time an event occurs for a device, a range of devices, or devices of a certain class. This command reverses the effect of a START_DEVICE_TRACE command.

Syntax

```
[ { TERMINATE_DEVICE_TRACE } ]
[ { TDVTR } ]

[ { DEVICE } { * } ]
[ { } = { dvtyp [dvnm] } ]
[ { DV } { dvc } ]

[ LOG = bool ]

[ { DISPLAY_ATN } ]
[ { } = bool ]
[ { DATN } ]

[ { DISPLAY_WARN } ]
[ { } = bool ]
[ { DWARN } ]

[ { DISPLAY_ABN } ]
[ { } = bool ]
[ { DABN } ]
```

Parameters

DEVICE the device(s) for which the trace is to be terminated, expressed as:

* : all devices,
dvnm : a specified device,
dvtyp : all devices of a type,
dvc : all devices of a class.

When omitted, all devices (*) are assumed.

LOG if 1, the logging of events relating to the specified device(s) becomes standard.



DISPLAY_ATN	if 1, events that are Attentions are no longer be displayed.
DISPLAY_WARN	if 1, events that are normal terminations with marginal conditions are no longer displayed.
DISPLAY_ABN	if 1, events that are abnormal terminations are no longer displayed.

Rules

When no parameter or only DEVICE is specified, then logging becomes standard and no further events are displayed.

Examples

TDVTR	{Terminate logging and displaying all events for all devices.}
TDVTR MS	{Terminate logging and displaying all events for all disks.}
TDVTR MS/B10	{Terminate logging and displaying all events for all MS/B10 disks.}
TDVTR MS03	{Terminate logging and displaying all events for disk unit MS03.}
TDVTR DABN	{Terminate displaying attentions and normal terminations for all devices.}
TDVTR MS03 DATN DWARN	{Terminate displaying attentions and normal terminations with marginal conditions for disk unit MS03.}
TDVTR MS31 LOG	{Return to standard logging for disk unit MS31.}



8.6.8 TERMINATE_DYNSAVE_SVR (TDYNSVR)

Purpose

Shuts down the server.

Syntax

```
{ TERMINATE_DYNSAVE_SVR }  
{ TDYNSVR }  
  
[ FORCE = { bool | 0 } ]
```

Parameters

FORCE The FORCE parameter shuts down the server and causes the immediate ABORT of all saves in progress.

Rules

The server may be shut down with or without the FORCE option (by default: NO FORCE).

- **FORCE=1** this option is only used in the following two cases:
 - all dynamic saves in progress must be terminated.
 - there is a system deadlock.

In both cases, the FORCE option causes the immediate ABORT of the saves and shutdown of the server. A system deadlock is always cancelled by this command as it frees all semaphores on the server.

- **FORCE=0** (default value) this option shuts down the server only if there is no dynamic save in progress (and if the server is active). Otherwise shut down of the server will be refused with the following messages:

```
"TERMINATE REFUSED:SAVE RUNNING"  
"USE TERMINATE FORCE OR RETRY AT  
SAVE END"
```



Example

TDYNSVR {Produces the display shown below:}

BYA2: tdynsvr

PLUG MESS4 TERMINATE SERVER STARTED

PLUG MESS17 NO FILE IS UNDER DYNAMIC SAVE(TERMINATE)

BYA2:

BYA2: tdynsvr

PLUG MESS7 TERMINATE SERVER ASKED AND SERVER NOT ACTIVE

BYA2:



8.6.9 TERMINATE_ERROR_LOG (TERLOG)

Purpose

Terminate logging events in the SYS.ERLOG file. This command is to be used only under Service Center supervision.

Syntax

```
{ TERMINATE_ERROR_LOG }  
{ TERLOG }
```

Rules

None.

Example

```
TERLOG {Terminate error logging.}
```



8.6.10 TERMINATE_INPUT_READER (TIR)

Purpose

Specifies that job readers, started after an EJR command, no longer remain active when they have finished reading the currently entered jobs. The command acts as a flip/flop with START_INPUT_READER.

Syntax

```
{ TERMINATE_INPUT_READER }  
{ TIR }
```

PERM = bool

Parameters

PERM This parameter must be set for the command to be effective.

Rules

None.

Example

TIR PERM {After this command is entered, job readers started by EJR command no longer remains active once they have finished reading.}



8.6.11 TERMINATE_IO_CACHE (TIOC)

Purpose

This command is used to terminate the GCOS 7 Input/Output Cache function. See the *Large Memory Cache User's Guide* for full details.

Syntax

```
{ TERMINATE_IO_CACHE }  
{ TIOC }
```

Parameters

None.

Rules

None.

Example

TIOC (Termination of the GCOS 7 I/O Cache is requested).



8.6.12 TERMINATE_LOAD (TLD)

Purpose

Prevents jobs of specified class being selected for execution, deletes job class(s), starts/stops job tracing.

Syntax

```
{ TERMINATE_LOAD }
{ TLD                }

[ { CLASSLIST }   { *           } ]
[ {              } = { aa         } ]
[ { CLASS      }   { aa[-aa]    } ]

[ JTRACE = { bool | 0 } ]

[ { DELETE }      ]
[ {          } = { bool | 0 } ]
[ { DL       }      ]

[ { SYSTEM }      ]
[ {          } = name4    ]
[ { SYS     }      ]
```

Parameters

CLASSLIST	the classes or range of classes to terminate, expressed as: * all classes, aa the specified class, aa-aa the specified inclusive range of classes.
JTRACE	if 1, job tracing is stopped. Default is 0.
DELETE	deletes the class or range of classes specified. Default is 0.
SYSTEM	This parameter indicates the system where the command is executed.



Rules

- CLASSLIST or JTRACE=1 must be used, but not simultaneously.
- DELETE must not be 1 when CLASSLIST=*, and is meaningless when JTRACE=1.
- DELETE is only available where there is no job in the class specified. If this is not the case, introduce the command later.
- when CLASSLIST is not used and JTRACE=0, the command is ignored.

Examples

TLD *	{Terminate all classes.}
TLD QZ	{Terminate class QZ.}
TLD AA-QZ	{Terminate classes AA-QZ.}
TLD AA-AD DELETE	{Delete classes AA,AB,AC,AD.}



8.6.13 TERMINATE_MEMBER (TMB)

Purpose

Stops a single MEMBER, several MEMBERS or the whole COMPLEX. This command is a part of the High Availability product (HA). For more information, see the *High Availability Administrator's Guide*.

Syntax

```
{ TERMINATE_MEMBER }
{ TMB
    { MEMBER } { * }
    { MB      } { ( name4 [ name4 ] ) }
    [ FORCE = { bool | 0 } ]
```

Parameters

MEMBER	List of the MEMBERS to be terminated. The asterisk (*) terminates all MEMBERS of the COMPLEX that have no administered SERVICES running on them.
FORCE	The FORCE parameter terminates the MEMBERS immediately even if there are SERVERS still active on these MEMBERS. The JAS SERVICES are also considered terminated even if there are batch jobs using them. If these jobs are being journalized they are aborted; if not, the operator uses the DISPLAY_JAS command to identify them, then the CANCEL_JOB command to terminate them manually.

Rules

- The CSMR of the MEMBER on which the command is entered must be running.
- This command must not be submitted between reception of the "RESYNCHRONIZATION IN PROGRESS" and the "END OF THE RESYNCHRONIZATION" messages.
- The MEMBERS must be described in the COMPLEX description.
- The MEMBERS must have been started.



- The CMSR of the MEMBER on which this command is entered must be able to communicate with the CMSR of the MEMBER to be terminated (MEMBER not isolated and not crashed).
- If there are SERVICES running on a MEMBER, TMB, without the FORCE parameter, is rejected for that MEMBER .

Examples

TMB	MB=*	{Terminates the whole COMPLEX}
TMB	MB=MB1	{Terminates the MEMBER named MB1 if there are no more SERVERS running on it}
TMB	MB=MB1	FORCE {Terminates immediately the MEMBER named MB1}



8.6.14 TERMINATE_OUTPUT_WRITER (TOW)

Purpose

Stops a writer or a remote writer.

Syntax

```
{ TERMINATE_OUTPUT_WRITER }
{ TOW
    {
        { * | ALL }
        { ALLDV }
        { PRxx }
    ON = { name4 }
        { Wnnn }
        { $site:SPOOL }
        { rfile78 }

    [ SELCLASS = (a[-a] [a[-a]] ...) ]

    [ STATION = stnm ]

    [ STRONG = { bool | 0 } ]

    [
        { LEAVE } ]
    [ END = { } ]
    [
        { UNLOAD } ]
```

Parameters

ON	The identification of the writer to be stopped, expressed as:
* or ALL	all writers (both on devices and files)
ALLDV	all writers operating on devices (but not on files)
PRxx	the writer operating on the named printer
name4	the writer operating on the named station's logical device
Wnnn	the writer where identification (as returned by the START_OUTPUT_WRITER command) is Wnnn
\$site	SPOOL the standard SYSOUT file of the named site



rfile78	<p>the writer operating on the named local or remote (DPS 7000 or DPS 6) file. (See <i>START_OUTPUT_WRITER</i> for syntax and rules.)</p> <p>For remote operations, refer to the <i>DPS6 - DPS7 Remote Facilities</i> manual and to the <i>DJP User's Guide</i>.</p>
SELCLASS	<p>restricts the scope of the command to the outputs that belong to the specified class(es) or range of classes. This may be expressed as:</p> <p>class for a single class class-class for a range of classes</p> <p>where class is a letter (A to Z). When SELCLASS is omitted, all started classes are assumed.</p>
STATION	<p>the name of the (unattended) station to which the command applies.</p>
STRONG	<p>if 1, outputs being printed are stopped immediately; if 0 (default), the stop is deferred until the outputs being printed are ended.</p>
END	<p>specifies action to be taken at termination of the tape writer:</p> <p>UNLOAD requests automatic unloading of the tape. LEAVE (default) specifies no automatic unloading.</p> <p>This parameter is relevant only if the writer was started with PERM=1 (permanent writer) and when all started classes are requesting termination.</p>

Rules

- When ON is * or ALLDV, STRONG is ignored
- If ON is Wnnn, STATION is ignored.

**Examples**

```
TOW * ST=RSL6           {Stop all writers on station RSL6.}

TOW PRL6 STRONG        {Stop immediately the writer on the station's logical
                        device PRL6.}

TOW $MST1 :SPOOL ST=PUNCH
                        {Stop routing card-punch outputs from station PUNCH
                        to host's MST1 standard SYSOUT file.}

TOW W007               {Stop writer whose identification is W007.}

TOW <tape-file-description> SELCLASS=A-D END=UNLOAD
                        {Terminate a permanent writer on a magnetic tape file
                        for the output classes A-D and asks for an automatic
                        unloading of the tape.}
```



8.6.15 TERMINATE_POWER_TIMER (TPWRTM)

Purpose

This command is used to terminate the timer power control activity.

Syntax

```
{ TERMINATE_POWER_TIMER }  
{ TPWRTM }
```

Parameters

None.

Rules

This command is not available on DPS 7000/2X0/3X0 machines.

Example

```
TPWRTM {Requests termination of the power timer.}
```




8.6.16 TERMINATE_RMS (TRMS)

Purpose

Disables the connection of a remote maintenance operator or logs off such an operator when already logged on. The connection of an RMS (Remote Maintenance System) operator is then submitted to a main operator for approval.

Syntax

```
{ TERMINATE_RMS }
{ TRMS
    [ ON = { GCOS | SP } ]
    [ LOCK = { PL | LC } ]
```

Parameters

ON Specifies the RMS type:

GCOS functions on all DPS 7000 machines.

SP indicates the RMS supported by the Service Processors of the DPS 7000/An and the DPS 7000/5x0/7xx/8x0/Dx0/Mx0 machines.

LOCK Designates the PL or the LC object when connecting under SP RMS:

LC means the Logical RMS Connection. It is the default for the DPS 7000/2X0/3X0/5X0/7xx/8x0/Dx0/Mx0 machines.

PL means the Physical RMS Line. It is the default for the DPS 7000/An.

On the DPS 7000/5x0/7xx/8x0/Dx0/Mx0, the default object when connected through the Service Processor (SP), is LC. The LOCK parameter can be used to change the object to PL.



Rules

- Except as explained above for the DPS 7000/5x0/7xx/8x0/Dx0/Mx0, an incompatible object causes the command to be rejected.
- When ON=GCOS, ENABLE must be equal to LC.

Examples

TRMS {Terminates a GCOS type RMS session.}

TRMS ON=SP LOCK=PL {On the DPS 7000/5x0/7xx/8x0/Dx0/Mx0, forces physical disconnection of the line forbidding all incoming and outgoing calls. On the DPS 7000/An, outgoing calls are still permitted.}

TRMS ON=SP LOCK=LC {On the DPS 7000/5x0/7xx/8x0/Dx0/Mx0, when the LC state is WAIT or USED, this command invalidates the logical connection. It allows the main operator to terminate immediately a USED (interactive or transfer out) session. It brings the Logical RMS connection back to the LOCK state; the physical line remains valid and another call can be submitted to the main operator for authorization.}



8.6.17 TERMINATE_SERVICE (TSRV)

Purpose

This command stops a SERVICE on all the MEMBERS. This command is a part of the High Availability product (HA). For more information, see the *High Availability Administrator's Guide*.

Syntax

```
{ TERMINATE_SERVICE }
{ TSRV                }

    RESYNC = { bool | 0 }

    FORCE = { bool | 0 }

    { SERVICE }
    {         } = name8
    { SRV     }

    [ { OPTION }           ]
    [ {           } = char255 ]
    [ { OPT     }           ]
```

Parameters

SERVICE	Name of the SERVICE.
OPTION	String of parameters which are specific to this type of SERVICE, such as the parameters given at the M STOP command of a TDS.
RESYNC	Used to terminate the SERVICES which had been terminated on the "co-MEMBER" during failure of the "current MEMBER" where there has been a DOUBLE FAILURE. To use RESYNC the following conditions must be met: <ul style="list-style-type: none"> – the co-MEMBER must be crashed, silent or telecom isolated. – the SERVICE to terminate must be HA



- the BACKUP SERVER must be running on the current MEMBER

FORCE

The FORCE parameter terminates the SERVICE immediately. The JAS SERVICES are also considered terminated even if there are batch jobs using them. If these jobs are being journalized they are aborted; if not, the operator uses the DISPLAY_JAS command to identify them, then the CANCEL_JOB command to terminate them manually.

Rules

- If the FORCE parameter is not used, a JAS type SERVICE is not terminated until all the steps that use it are terminated. Steps that use the journal terminate normally; all others, which can be identified with DJAS, must be terminated manually with the CANCEL_JOB command.
- If it is a JAS type SERVICE to be terminated, a batch job running on this SERVICE is allowed to finish unless the FORCE parameter is used.
- The CMSR of the MEMBER on which this command is entered must be running.
- This command cannot be submitted between the messages "CMSR RESYNCHRONIZATION IN PROGRESS" and "END OF CSMR RESYNCHRONIZATION".
- The SERVICE must be in the COMPLEX description.
- If the SERVICE to be terminated is still in use by another SERVICE, it is not terminated unless the FORCE parameter is entered in the command.
- If there is an isolation of the ACTIVE MEMBER, the CMSR rejects the command.
- If the RESYNC parameter is entered, there must be a DOUBLE FAILURE, and the service to be terminated must be HA.
- Two successive TERMINATE_SERVICE commands without the FORCE parameter are only accepted if the SERVICE type is TDS. This is useful to modify the OPTION parameter.
- A TSRV command with FORCE has priority over TSRV without FORCE.



Examples

TSRV SRV=GREEN {Terminates the SERVICE named GREEN on all the MEMBERS where it is running.}

TSRV SRV=TDS1 OPT=DISREST
{Terminates the SERVICE named TDS1 on all the MEMBERS where it is running; in this example it corresponds to the TDS command M STOP with the parameter DISREST.}

TSRV SRV=GREEN FORCE
{Terminates immediately the SERVICE named GREEN on all MEMBERS where it is running.}



8.6.18 TERMINATE_SSI (TSSI)

Purpose

Terminates the SSI on OPEN 7.

Syntax

```
{ TERMINATE_SSI }  
{ TSSI }  
  
[ STRONG = { bool | 0 } ]
```

Parameters

STRONG requests urgent termination.

Rules

- The SSI and ACSEL processes are halted. This breaks the connection between the local GCOS 7 and the cartridge library server.
- The STRONG parameter forces an immediate termination of the ssi and acsel processes (kill -9). You should avoid this option.
- In normal use, the TERMINATE_SSI command is necessary if the SSI configuration needs to be modified.
- You can either terminate CLX before terminating SSI or keep CLX running. However, terminating SSI can cause a termination of CLX.

Example

S: TSSI



8.6.19 TERMINATE_SYSTEM (TSYS)

Purpose

Asks for a system shutdown. No further jobs are allowed to start; the main operator is logged off when all jobs have reached the end of their current step or a checkpoint; the system then shuts down and traffic stops. The main operator is logged off if working in CONS operating mode. Otherwise he must enter the "BYE" command or modify the operating mode.

Syntax

```
{ TERMINATE_SYSTEM }  
{ TSYS }
```

GCOS

[INIT = { bool | 0 }]

[PWROFF = { bool | 0 }]

[RESET = { bool | 0 }]

Parameters

GCOS	must be specified for an effective shutdown. This parameter is present to avoid an accidental shutdown as a consequence of wrongly selecting TERMINATE_SYSTEM on a command menu.
INIT	This parameter is valid only for DPS 7000 Systems. if 1, the system is completely reinitialized (hardware and firmware) from the device specified through the MDRSO command, provided that the REPEAT option is active. if 0 (default), no automatic reinitialization takes place. Depending on the value of the REPEAT option, the system is automatically reloaded or not after the shutdown is completed. The REPEAT option value can be set by using the MDRSO command. To display the REPEAT option, use the DRSO command.



The following options are not available on the DPS 7000/2X0 or 3X0.

PWROFF	If 1, the system is automatically powered off once shutdown is completed. If 0 (default), the system remains powered on after termination.
RESET	If this option is available, it specifies that the operator must reinitialize the system to reload it. Default=0 (no reinitialization needed).

Rules

- If some jobs are suspended, the command is denied. All suspended jobs must be released before shutdown.
- If there are Services running under the Complex Management Server (CMSR) product, the command is denied.
- When GCOS=0, the command is ignored.
- INIT, PWROFF, and RESET are mutually exclusive.

Examples

TSYS GCOS	{Asks for system shutdown.}
TSYS GCOS PWROFF	{Asks for system shutdown, followed by system power-off.}



8.6.20 TERMINATE_SYSTEM_TRACE (TSYSTR)

Purpose

TSYSTR stops the system trace collector service job, stops domain trace activity undefined classes, and stops the domain WAIT mode for trace event requests.

NOTE:

This command is to be used only under the direction of a Bull support engineer.

Syntax

```
{ TERMINATE_SYSTEM_TRACE }
{ TSYSTR
  [ { DOMAIN } { ALL } ]
  [ { DOM } = { * } ]
  [ { TDOM } { star64 } ]

  [ CLASS = { NOR|DET|TRM|INR|IDT|CND|DTV|ABN|ARC|TRP|TDP|ALL } ]

  [ WAIT = bool ]

  [ { COLLECTOR } ]
  [ { } = bool ]
  [ { CLTR } ]
```

Parameters

DOMAIN	It is the name of a system trace domain or, if ALL (the default) or * is specified, all domains. Maximum length is 64 characters.
CLASS	It is either ALL (the default, meaning all classes) or a system trace class name. Correct values are: NOR, DET, TRM, INR, IDT, CND, DTV, ABN, ARC, TRP, and TDP.
WAIT	If 1, the domain WAIT mode is to be stopped; if 0, the WAIT mode is not stopped.
COLLECTOR	If 1, the system trace collector service job is to be stopped; if 0, the trace collector is not stopped.



Rules

- The domain WAIT attribute is meaningless, unless a system trace collector is active.
- CLASS and WAIT are mutually exclusive.
- If COLLECTOR is specified no other parameter may be specified.

Examples

TSYSTR DOMAIN=MYTDOM

{Close all MYTDOM domain classes.}

TSYSTR DOMAIN=MYTDOM WAIT

{Reset trace event WAIT mode for requests issued for MYTDOM domain.}

TSYSTR COLLECTOR

{Stop the system trace collector service job.}



8.6.21 TERMINATE_TERMINAL_OUTPUT (TTO)

Purpose

Terminates the printing of outputs on a terminal or pool of terminals or modifies the class of outputs that can be printed on them. Only the owner of the object terminal or users declared as Master Operators for the Generalized Terminal Writer may use this command. Refer to the *GTWriter User's Guide* for a complete description.

Syntax

```
{ TERMINATE_TERMINAL_OUTPUT }
{ TTO                          }

      { TERM }
      {     } = name12
      { TN   }

[SELCLASS = a [-a]]

[STRONG = { bool | 0 }]
```

Parameters

TERM	the name of the terminal or pool of terminals to be terminated or modified.
SELCLASS	indicates the class (SELCLASS = a) or range of classes (SELCLASS = a-a) which are no longer eligible for printing on that terminal or pool of terminals. If no class is left for the terminal or pool, it is stopped.
STRONG	if 1, terminate even if an output is being printed.



Rules

SELCLASS and STRONG are mutually exclusive.

Examples

TTO POOL3	{Stop printing on terminal pool POOL3.}
TTO POOL3 STRONG	{Stop printing on terminal pool POOL3, even if printing.}
TTO POOL3 A-D	{Stop printing for classes A through D on pool POOL3.}



8.6.22 TERMINATE_TERMINAL_WRITER (TTW)

Purpose

Shuts down one or all of the Generalized Terminal Writer drivers. Refer to the *GTWriter User's Guide* for a complete description.

Syntax

```
{ TERMINATE_TERMINAL_WRITER }
{ TTW

    [ { name4 } ]
    [ { ALL } ]
    [ { FORCE } ]

    [STRONG = { bool | 0 }];
```

Parameters

name4	This is the name of the driver to be stopped (4 char.). When omitted, all drivers (ALL) are assumed.
ALL	The default parameter shuts down all the drivers.
FORCE	This parameter shuts down the command handler as well as all the drivers. The TSYs command has the same effect.
STRONG	This parameter is used with one of the other parameters. It immediately stops the output. For example, any output that is printing is frozen.

Rules

None.

Examples

TTW	{shutdown all drivers}
TTW TW01	{shutdown driver TW01}
TTW FORCE	{shutdown all drivers and the command handler}
TTW TW01 STRONG	{shutdown driver TW01 immediately}



8.6.23 TEST_DEVICE (TSTDV)

Purpose

Starts and stops a print test for those printer devices that have no PRINT TEST pushbutton.

Syntax

```
{ TEST_DEVICE }  
{ TSTDV       }  
  
    { DEVICE }  
    {       } = PRxx  
    { DV     }  
  
ACTION = { START }  
         {       }  
         { TERM  }
```

Parameters

DEVICE	the identification of the printer device for which the test is to be started or stopped.
ACTION	If START, the test is started on the named printer device when it is set to the READY state from STANDBY. After the test, the printer is automatically reset to STANDBY. The test is restarted when the printer is again set to the READY state. If TERM, the test is terminated (i.e., setting the printer to READY no longer initiates the test).

Rules

None.

Examples

```
TSTDV PR03 START {start a print test for printer PR03}  
TSTDV PR03 TERM  {terminates the print test for printer PR03}
```



8.7 "U" Commands

8.7.1 UNLOCK_MIRLOG_FILE (UNLMIRF)

Purpose

Unlocks either logical subset (RECAREA, NSTDTAB) of the SITE.MIRLOG file when it has been locked by the other system in coupled systems environment.

This command is available only if the system has been configured with MIRROR = YES.



ATTENTION:

This is a dangerous command, to be used only after one of these messages:

```
MR17 WARNING: RECAREA OF THE SITE.MIRLOG IS LOCKED BY  
THE OTHER SYSTEM
```

```
MR17 WARNING: NSTDTAB OF THE SITE.MIRLOG IS LOCKED BY  
THE OTHER SYSTEM
```

An OCL version exists, intended to be used at GCOS 7 restart (before GCOS READY).

Syntax

```
{ UNLOCK_MIRLOG_FILE }  
{ UNLMIRF }
```

```
      PART = { RECAREA }  
             {           }  
             { NSTDTAB }
```

OCL syntax

```
UMIRF { RECAREA }  
      { NSTDTAB }
```



Parameters

PART This parameter identifies the logical subset of the SITE.MIRLOG file that is to be unlocked:

RECAREA is unlocked to enable resynchronization and restart.

NSTDATAB is unlocked to enable copy invalidation.

Rules

Refer to the *Mirror Disks User's Guide*.

Example

```
UNLMIRF PART=RECAREA
      {Unlocks logical subset RECAREA of the
      SITE.MIRLOG file and displays the message: "MR18
      UNLOCK OF THE RECAREA PART OF THE SITE.
      MIRLOG SUCCESSFUL"}
```




8.7.2 UNLOCK_MIRROR_VOLUME (UNLMIR)

Purpose

Unlocks the MIRLAB of the either disk belonging to a pair of SHARED mirrored disks, when locked by the other system in coupled systems environment.



ATTENTION:

This is a dangerous command, to be used only after the message:

```
MR18 WARNING: MIRLAB OF volname:dvc IS LOCKED BY THE
OTHER SYSTEM
```

It unpairs two mirrored disks of a dual volume.

An OCL version exists, intended to be used at GCOS restart (before GCOS READY).

This command is available only if the system has been configured with MIRROR = YES.

Syntax

```
{ UNLOCK_MIRROR_VOLUME }
{ UNLMIR                }

      { VOLUME }
      {         } = volname-6:device class
      { VOL     }
```

OCL syntax

```
UMIRV volname6 dvc
```

Parameters

VOLUME Identification (volume name:device class) of the pair of mirrored disks. Ex : MIRB55:MS/B10.



Rules

Refer to the *Mirror Disks User's Guide*.

Example

```
UNLMIR MIRB55:MS/B10
    {displays the message:
    MR18 UNLOCK OF MIRLAB OF MIRB55
    MS/B10 SUCCESSFUL}
```



8.7.3 UNPAIR_MIRROR_VOLUME (UNPMIR)

Purpose

Unpairs the two disks of a mirrored pair; the disk specified via the KEEP parameter becomes MIRROR ALONE, while the other disk goes NSTD MIRROR (when it is accessible).

This command is available only if the system has been configured with MIRROR = YES.

Syntax

```
{ UNPAIR_MIRROR_VOLUME }
{ UNPMIR
    { VOLUME }
    { VOL } = volname-6:device class
    KEEP = { PRIMARY }
           { SECONDARY }
```

Parameters

VOLUME	Identification (volume name:device class) of the pair of mirrored disks. Ex : MIRB55:MS/B10.
KEEP	Identification of the copy to be kept as valid copy of the volume i.e. the disk which is to become ALONE. Possible values: PRIMARY or SECONDARY keywords.



Rules

Refer to the *Mirror Disks User's Guide*.

Example

```
UNPMIR MIRB55:MS/B10 KEEP=PRIMARY
      {displays the messages:
MR03 INVALIDATION OF MIRB55 MS/B10
      REQUESTED BY OPERATOR
MR12 UNPMIR ACCEPTED
DV26 dvm1 PREMOUNTED MIRB55 NSTD
      MIRROR
MR08 dvm2 MIRB55 MIRROR ALONE}
```





9. Remote Multiplexed Operator Support (RMOS)

9.1 Introduction

This section introduces the D0F7-RM product, also known as RMOS (Remote Multiplexed Operator Support).

- how to use RMOS commands.
- how to leave RMOS.
- RMOS screens.
- RMOS error messages.

HELP TEXTS

On-line documentation in the form of Help texts is available to you at all levels of operation. These help texts contain a subset of the information in this section. To obtain help texts, enter the question mark (?) in the control field as a reply to the appropriate prompt. The first screen displayed gives a description of RMOS commands.

Table 9-1. Summary Description of RMOS Commands

CONTROL	Specify/modify the mode of operation of DISPLAY
DISPLAY	Display the status of RMOS link(s)
INTERRUPT	Issue a break to the current selected site
LINK	Link a site / user / application to RMOS
PURGE	Delete unwanted pending messages
QUIT	Leave RMOS (implicitly deleting all defined links)
SELECT	Select a link for activation
UNLINK	Delete a link (created using the LINK command)



9.1.1 Definition of RMOS

The Remote Multiplexed Operator Support is an interactive processor that runs under IOF. Its purpose is to monitor one or several sites using the same operator terminal without having to log-on/log-off. RMOS allows you to multiplex several interactive sessions between your local site and one or more applications located on the same or other sites. The connection between the RMOS processor and the monitored site(s) is called a link.

You normally proceed as follows:

1. You establish a link with a site using the LINK command. A link is completely identified by site, user, and application. You can have up to 64 existing links simultaneously. You can establish zero, one, or more links to a single site. The site concerned can be local or remote. If you make more than one link to the same site, then the user and/or the application must be different. A link can be considered as an interactive session with an application. A link remains defined until either you delete it explicitly (UNLINK command), leave the linked application using its exit procedure (e.g., BYE for IOF), or leave RMOS (QUIT command).
2. You select a link to become the current link using the SELECT command. Messages issued by the selected link (in fact by the application for the user at the selected site) are displayed on your terminal. This link remains as the selected link until you select another link.
3. You can display the status of the link(s) using the DISPLAY command. The CONTROL command specifies the operating mode of this display.
4. You can use the INTERRUPT command to cause a break at a selected site.
5. You can suppress unwanted pending messages using the PURGE command.
6. You can delete a link that is no longer needed using the UNLINK command.
7. You leave RMOS using the QUIT command. This implicitly deletes all links.

You must remember two important points:

- Because RMOS is an interactive processor, a user running under a project with the MAIN attribute cannot activate it.
- RMOS only multiplexes connections (links) and does not perform message management. To select messages, use the relevant filter commands at each connection level.



9.1.2 Functions

With RMOS you can:

1. Create/delete up to 64 links between one RMOS processor and the local/remote site(s). The actual number of links are limited by such resources as memory, the multiprogramming level made available and the space occupied by temporary files.
2. Store for each existing link all incoming messages from the remote sites in a Queued Linked subfile. This file is associated with the link when it is set up. This means you can retrieve these messages on the RMOS Operator Terminal when needed.
3. Select at any time one of the existing links to dialog with the remote site as if the RMOS Operator were directly logged on at the site.
4. Display existing links and their status.
5. Display for all links messages requesting the RMOS Operator's attention.
6. Attempt to make an automatic reconnection after an abnormal link disconnection.
7. Support up to nine simultaneous RMOS processors executing concurrently on the same system.

NOTE:

The local site where RMOS is loaded must have GCOS 7 V3 or later releases. The "remote sites", with which the links are established can be in any GCOS 7 release but a link with the MAIN attributes is supported as from GCOS 7 V3. Links with the older Level 64 sites are not supported.

9.1.3 Basic Architecture

RMOS is an interactive processor called under IOF.

RMOS consist of two processes. The first process (P0) is in charge of the RMOS Operator terminal management involving command processing, dialog with remote sites when a link is active, status display. The second process (P1) is in charge of managing remote sites.

RMOS commands are GCL commands belonging to a specific domain named "RMOS" which is automatically initialized at RMOS processor start time.

The RMOS processor is connected to the remote sites using VCAM (Virtual Communications Access Method). These RMOS connections or links are said to be Pass-Through type links.



After you enter the ENTER_RMOS command, you can specify which storage techniques you wish to employ for storing incoming messages by specifying the MODE parameter. If you specify MODE = FILE, you can either name an existing Queued File with the correct allocation parameters or let RMOS create a temporary file to be used during the RMOS session and deleted when it is over. The only advantage of a permanent file is that you may locate it more easily. If you specify a permanent file and RMOS is unable to open it, RMOS displays an error message on your terminal and allocates a temporary file. For each link, one subfile is created.

A change of operating mode implies terminating the current session and restarting a new one.

When the file is full, RMOS automatically allocates additional space up to the maximum file size limits. When these maximum values are reached, the RM53 message (UNABLE TO EXTEND RMOS FILE: SOME MESSAGES MAY BE LOST) is issued. At this point, RMOS deletes the oldest messages in the largest subfiles for links which have not been recently selected.

In the case of overflow, RMOS reduces the size of the largest subfiles and thus equally distributes messages over all the subfiles. Use the SELECT or PURGE command to create more space; then you can resume normal FILE Mode processing.

Characteristics of the file used to store incoming link messages:

VBO	FBO
FILEFORM = BFAS	FILEFORM = LINKQD
FILEORG = LINKQD	FILEORG = LINKQD
BLKSIZE = 6200	BLKSIZE = 6200
RECSIZE = 6196	RECSIZE = 6196
RECFORM = VB	RECFORM = VB
DELREC = YES	DELREC = YES
COMPACT = NO	COMPACT = NO

Any other parameters depend on size, location and number of links to be handled.



9.1.4 RMOS Operator Levels

RMOS contains three levels:

- Command level.
- Display level.
- Select level.

9.1.4.1 Command Level

On executing the ENTER_RMOS command (of the IOF domain), you enter RMOS at the command level. In serial mode, the prompt 0: is displayed at this level. The following commands leave you at command level after execution.

```
CONTROL  
  
LINK  
  
PURGE  
  
UNLINK
```

9.1.4.2 Display Level

The command DISPLAY places you at the display level after execution. To return to command level, you issue a break (if in AUTOMATIC display MODE) or enter a slash (/) command (if in MANUAL display mode). AUTO and MANUAL are possible values for the MODE parameter of the CONTROL command.



9.1.4.3 Select Level

The RMOS commands INTERRUPT and SELECT place you at the select level after execution. To return to command level, you issue a break.

The following table summarizes the above discussion.

Table 9-2. Summary of RMOS Tasks

TASK	USE COMMAND	LEVEL	RETURN TO RMOS COMMAND LEVEL
Specify/modify the display mode	CONTROL	COMMAND	
Display information about a site	DISPLAY	DISPLAY	BREAK (AUT/MAN) or "/" (MANUAL)
Issue a break to currently selected site	INTERRUPT	SELECT	BREAK
Link a site to RMOS	LINK	COMMAND	
Delete unwanted pending messages	PURGE	COMMAND	
Leave RMOS	QUIT		
Select a link	SELECT	SELECT	BREAK
Delete a link	UNLINK	COMMAND	



9.2 Description of RMOS Commands

This section describes each RMOS command in detail, and shows you how to use each command. The commands are presented as follows in alphabetical order, to facilitate ease of reference:

- CONTROL
- DISPLAY
- INTERRUPT
- LINK
- PURGE
- SELECT
- UNLINK

The description of each command begins on a new page. For each command, details include:

- purpose,
- command syntax,
- description of parameters,
- an example of the use of the command.

PMOS GCOS 7 commands have two standard forms which work just the same:

1. The full command (verb): CONTROL.
2. The abbreviation: CTL.

Certain commands and parameters, usually those that are rarely used, are not included in the menu and or prompt mechanism. The parameters in this category are listed in this section below a dashed line:-----.

You have these parameters listed in the menu by entering a plus (+) after its name, rather than a question mark (?). There are no help texts for these parameters.



9.2.1 RMOS (ENTER RMOS)

Purpose

RMOS multiplexes several interactive sessions between your local site (the site at which you execute the ENTER_RMOS command), and one or more applications.

On executing the ENTER_RMOS command, you leave the IOF domain and enter the RMOS domain. When the processor is ready to accept commands, a specific RMOS prompt is displayed (0:).

Syntax

```
{ ENTER_RMOS }
{ RMOS      }

[          { FILE      } ]
[ MODE =  {           } ]
[          { MESSAGE  } ]

[FILE = libnm ]
```

Parameters

MODE	Specifies the mode in which RMOS is to operate.
MESSAGE	Messages issued by a LINKed site (which is not yet SELECTed) are not placed in a file. They are put in the queue at the issuing site. In this case, you must omit the FILE parameter.
FILE	Messages issued by a LINKed site are placed in a file until the site is SELECTed (LINK and SELECT are commands of the RMOS domain). The file is either a temporary one created by RMOS (default), or an existing file specified via the FILE parameter. If you omit MODE, the default is MODE = FILE. Note that in file mode, the number of buffers created for accessing file(s) increases with the number of linked sites. Because the size of each buffer is 6 Kbytes, a large amount of memory is used.



You enter the name of an existing SL library. The file must be present at your local site (i.e. the site where you issued the ENTER_RMOS command), and its block size must be greater than or equal to 6200 bytes.

Messages issued by a LINKed site are placed in this file until the site is SELECTed (LINK and SELECT commands of the RMOS domain).

If you omit FILE (and MODE is not MESSAGE) then by default, RMOS creates a temporary file which will be deleted at the end of the RMOS session.

Examples

- ENTER_RMOS ; (Activate the RMOS processor. By default, messages are placed in a temporary file created by RMOS.)
- ENTER_RMOS FILE = (Activate the RMOS processor. By default, messages are placed in the catalogued SL library P1.MYSL1. This library exists at your local site.)
P1.MYSL1 ;
- ENTER_RMOS MODE = (Activate the RMOS processor to operate in RMOS mode. Messages are not be placed in a file; they are put in the queue at the issuing site.)
MESSAGE ;



9.2.2 CTL (CONTROL)

Purpose

Specifies/modifies the attributes of the RMOS display (produced by the DISPLAY command). The abbreviation for the CONTROL command is CTL.

Syntax

```
{ CONTROL }
{ CTL      }

[          { AUTO  } ]
[  MODE = {      } ]
[          { MANUAL} ]

[{ DISPLAY_DELAY } ]
[{              } ] = [dec2] [.dec2] ]
[{ DELAY        } ]
```

Parameters

MODE	This parameter specifies the mode used to handle a multi-page display.
AUTOMATIC (AUTO)	The display automatically switches to the next page after a specified time period which is determined by the value of the DISPLAY_DELAY parameter. Once the last page is reached, the display automatically restarts from the first page. To leave the display, you issue a break. AUTO is the default.
MANUAL	You control the display by entering command(s) in the control field, indicated by the arrow in the upper right-hand corner of the screen. >[n] Advance page or n pages if n is specified. <[n] Go back page or n pages if n is specified. =n Display page number n R Refresh data and display page 1 / Leave Display and return to RMOS command level.



DISPLAY_DELAY (Abbreviation is DELAY)

Specifies the time interval between the automatic display of successive pages of a multi-page display. Such displays can be produced using the DISPLAY command. You enter a value on the form [min] [.sec], where min = minutes, and sec = seconds. Leading/trailing zeros may be omitted. Seconds must be introduced by '!'. The specified time must be at least 5 seconds. The default value is 60 seconds.

The displays are described in further detail in Section 9.4 *Description of Screens*.

Examples

CONTROL MODE = MANUAL;

(The display is in manual mode; that is, controlled by yourself. In this case the DISPLAY_DELAY parameter is meaningless.)

CONTROL MODE = AUTO;

(The display is in automatic mode. Because the DISPLAY_DELAY parameter is omitted, the default value (60 seconds) is assumed.)

CONTROL DELAY = 0.30;

(The time interval between display of two consecutive pages is 30 seconds.)

CONTROL;

(The display is in automatic mode and the time interval is reset to 60 seconds.)



9.2.3 D (DISPLAY)

Purpose

This command displays information on your terminal about a site or all sites. The abbreviation is D.

Syntax

```
{ DISPLAY }
{ D      }

[      { { GS          } } ]
[      { { GLOBAL_STATUS } } ]
[OPTION = { {          } } ]
[      { { GR          } } ]
[      { { GLOBAL_REQUESTS } } ]
```

Parameters

OPTION Specifies the type of information to be displayed. You can display one of the following:

GLOBAL_STATUS or GS This means a global display of the status of each LINKed site.

Displays the number of messages in the queue coming from each site.

If the flow of messages has been interrupted (for a link), the number of messages is highlighted.

This is the default value if you omit **OPTION**.

GLOBAL_REQUESTS or GR

This means a global display of all requests coming from the linked site(s). The request include all system messages beginning with '*' and all messages requiring a reply. The later are prefixed with the reply number.

NOTE:

To use this option properly, you must activate appropriate filters at the site(s) currently linked to RMOS.



Examples

DISPLAY; or D; {Display information about the status of all sites
 currently linked to RMOS. This is the default value if
 you omit OPTION.}

DISPLAY OPTION = GLOBAL_REQUESTS; or D GR;
 {Display information about the requests from all sites
 currently linked to RMOS.}



9.2.4 IT (INTERRUPT)

Purpose

Used to issue a break to the currently selected site.

The INTERRUPT command (abbreviation is IT) is ignored if you have not selected a site (using the SELECT command).

If you issue a break (by pressing the appropriate key on your terminal, or by entering `*$BRK`), this break is handled locally by RMOS; it is never transmitted to the selected site. Consequently the INTERRUPT command provides the only means of issuing a break that is effective at a remote site.

INTERRUPT does not stop the display of messages coming from a selected site which have already been queued in the RMOS file. To do this, use the PURGE command.

Syntax

```
{ INTERRUPT }  
{ IT }
```

Parameters

None.



9.2.5 LK (LINK)

Purpose

Used to link a site to RMOS.

Syntax

```
{ LINK }
{ LK   }

SITE = sitnm

[ TYPE = PT ]

[ USER = usernm ]

[ { PASSWORD }           ]
[ {               } = char12 ]
[ { PSW         }           ]

[ PROJECT = projnm12 ]

[ BILLING = name8 ]

[ { APPLICATION }           ]
[ {               } = name12 ]
[ { APPL       }           ]

[ STATION = stnm ]

[ OPTIONS = name22 ]
```



Parameters

SITE	<p>Name of the site to which a link is to be made by the LINK command (See OPTIONS parameter). The site can be local (that is the site at which you activated RMOS), or remote (i.e. a site other than the site at which you activated RMOS).</p> <p>You enter the name of a site known on your network; the name must be that used in the RSYS parameter of the CNC (Communications Network Configuration). The name can be up to four characters long. It can be made up of the letters (A- Z) and the digits (0-9). It must begin with a letter.</p> <p>Note that the link being made is identified by SITE, USER (specified explicitly or taken by default), and APPLICATION (specified explicitly or taken by default). The user and the application must exist at SITE, and the user must be known to the application at SITE.</p>
TYPE	<p>Specifies the type of link to be made.</p> <p>The only type available in the current version of RMOS is TYPE=PT (Pass-Through). This is the default value.</p>
USER	<p>Specifies the name of the user to be associated with the link being established by the LINK command.</p> <p>The user name entered must be known at the site to which the link is being made. The user must also be known to the application concerned at this site. If you wish to use the link to perform remote site operation, then the user must also have the MAIN attribute at the site being linked.</p> <p>If you omit USER, the user name under which you activated RMOS is taken as the default. This name must comply with the details in the previous paragraph.</p>



PROJECT	<p>Specifies the name of the project to be associated with the link. The name you supply must be a valid project on the site to be linked. The user name must be registered under that project.</p> <p>If you omit PROJECT, the default project for the user name (as registered in the site catalog of the site being linked) is assumed.</p>
BILLING	<p>Specifies the name of the billing to apply to the session with the site being linked.</p> <p>You cannot specify BILLING unless you also specify PROJECT.</p> <p>The name you supply must be a valid billing for the project specified via PROJECT on the site to which the link is being made.</p> <p>If you omit BILLING, the default billing for the project is assumed.</p>
PASSWORD	<p>The password parameter (abbreviation PSW) specifies the password for the user specified via the USER parameter.</p> <p>If you specify USER and there is a password associated with the user, you must also give the password.</p> <p>If you omit USER (and PASSWORD), then your own user name and password are used. These must be valid at the site to be linked (that is, they must be registered in the site catalog of that site). Your password must be the same at both sites.</p>
STATION	<p>Specifies the name of the station to be associated with the link.</p> <p>The name you supply must be a valid station name on the site to be linked. The station must be known to the application concerned.</p> <p>The name can be up to eight characters long. It can be made up of letters (A-Z) and digits (0-9). It must begin with a letter.</p> <p>If you omit STATION, the default station for the project is assumed (even if you have not explicitly specified the project name).</p>



APPLICATION	<p>Specifies the name of the application to be accessed. The abbreviation is APPL.</p> <p>You enter the name of the application which exists at the site being linked. The application must be one of those permitted to the user (USER parameter) as defined in the site catalog of the site being linked.</p> <p>If you omit APPL, the default is IOF.</p>
OPTIONS	<p>Used to transmit option parameters to the application to which the link is being made.</p> <p>IOF applications option values:</p> <ul style="list-style-type: none">– NS: No optional start-up execution at link connection time.– NEW: Starts a new IOF session (even if a disconnected session is available for reconnection).– NM: Cancel mail display.– INT: Start the IOF session in interactive mode (when creating a link with a user who would otherwise implicitly be in CONS mode).– *envt: Asterisk followed by the name of the environment under which the application is to start. <p>The values given are interpreted by the application concerned.</p> <p>If you omit OPTIONS, there is no default. In this case, no link option is transmitted to the application.</p> <p>Links to TDS applications:</p> <ul style="list-style-type: none">– Application = tdsnm– Options = Refer to the DNS Terminal Management Reference Manual.



Examples

```
LINK SITE = MINE TYPE = PT USER = ME PROJECT = MYPROJ1  
BILLING = MYBILL1 PASSWORD = XYZ STATION = ST3 OPTIONS = 'NS NEW';
```

(A link is to be made to the site named MINE. The type of link to be made is PT (Pass Through); the user is ME; the project name is MYPROJ1; the billing is MYBILL1; the password is XYZ; the station name is ST3; the application is IOF by default; the options are no optional start-up execution and start a new session.)

```
LINK SITE = OURS OPTIONS = 'NM *E3';
```

(A link is to be made to the site named OURS. The display of mail is suppressed and the environment named E3 is used.

The following default values are assumed: the type is PT; the user is the user name under which you activated RMOS; the project is the default project for the user name (as registered in the site catalog of the site being linked); billing is the default billing for the project; your own password, which is registered in the site catalog of OURS; the station is the default station for the project; the application is IOF.)



9.2.6 PRG (PURGE)

Purpose

Used to delete unwanted pending messages from a linked site which is not currently selected. This command is particularly useful when a link is not going to be SELECTed and a large number of messages are accumulating in the file (with a risk of overflow). The command abbreviation is PRG.

Since PURGE deletes messages from a file, PURGE is meaningless if a file is not being used (that is, if you specified MODE = MESSAGE when calling RMOS).

Since PURGE is available only at RMOS command level, it cannot be used to delete messages being issued by the currently selected site. To do this, you must use the INTERRUPT command.

Syntax

```
{ PURGE }
{ PRG   }

SITE = sitenm

[USER = usernm]

[ { APPLICATION }           ]
[ {                   } = name8 ]
[ { APL           }           ]
```

Parameters

SITE Specifies the name of the site for which pending messages are to be purged.

You enter the name of a linked site, that is, you must have previously executed a LINK command for this site during the current RMOS session.

If necessary, the link can be more fully identified using the USER and APPLICATION parameters.

You must enter a value for SITE as there is no default.



USER Specifies the user name associated with the LINK whose messages are to be deleted. You enter the user name given when the link was made using the LINK command.

If you omit USER, the user name associated with the only existing link to the site (as given via the SITE parameter) is assumed.

If there is more than one existing link to SITE, then there is no default value for USER. In this case, if you omit USER, the PURGE command is not executed.

APPLICATION Specifies the application associated with the link.

Examples

PURGE SITE = MINE: (Purge pending messages associated with the link identified by the site named MINE. Because USER and APPLICATION have been omitted, there is only one link with this site.)

PURGE SITE = REM 3 USER = OPER APPL = IOF;
(Purge pending messages associated with the link identified by the site named REM3, the user named OPER, and the application IOF.)



9.2.7 Q (QUIT)

Purpose

Used to leave the RMOS processor.

Once the QUIT command (abbreviation is Q) is executed, all linked sites are automatically unlinked and the session with the current selected site is terminated. All messages stored in the RMOS file, if any, are deleted.

In order to prevent an error, the following question is displayed:

```
RM57 DO YOU WANT REALLY TO QUIT RMOS?
```

to obtain confirmation of the operator's intention to quit.

QUIT causes you to leave the RMOS domain and return to the IOF domain.

Syntax

```
{ QUIT }  
{     }  
{ Q   }
```

Parameters

None.

NOTE:

You can also execute a QUIT command by entering a slash (/) character as the command in non-menu mode.

Example

```
QUIT;                (Leave RMOS.)
```



9.2.8 SEL (SELECT)

Purpose

Used to select a link.

The command abbreviation is SEL.

Syntax

```
{ SELECT }
{ SEL   }

SITE = sitenm

[ USER = usernm ]

[ { APPLICATION }           ]
[ {                   } = name8 ]
[ { APPL         }           ]
```

Parameters

- | | |
|-------------|--|
| SITE | Specifies the name of the site for which a link is to be selected.
You enter the name of a linked site, that is, you must have previously executed a LINK command for this site during the current RMOS session.
If necessary, the link can be more fully identified using the USER and APPLICATION parameters.
You must enter a value for SITE as there is no default. |
| USER | Specifies the user name associated with the link to be selected. You enter the user name given when the link was made using the LINK command.
If you omit USER, the user name associated with the only existing link to the site (as given via the SITE parameter) is assumed.
If there is more than one existing link to the SITE, then there is no default value for USER. In this case, if you omit USER, the SELECT command is not executed. |
| APPLICATION | Specifies the name of the application (abbreviation is APPL) associated with the link to be selected. You enter the application name given when the link was established (using the LINK command). |



If you omit APPLICATION, the application with the only existing link to the site (given in SITE parameter) is assumed.

If there is more than one existing link to SITE, then there is no default value for APPLICATION. IOF is the default application.

Examples

SELECT SITE=MINE; (Select the link to the site named MINE. As USER and APPLICATION have been omitted, there must be only one link to this site, and this is selected.)

SELECT SITE = OURS USER = ME APPLICATION = TDSY;
(Select the link to the site named OURS, identified by the user named ME and the application name TDSY.)

NOTES:

1. If RMOS is operating in FILE mode, once the link has been selected, all messages issued since the previous selection (or time of link-up if this is the first time you have selected this link) are displayed at your terminal. After the messages are displayed, normal dialog is resumed.
This dialog depends on whether the application is in INT (interactive mode) or in CONS (console mode). INT and CONS are parameters of the MODIFY_OPERATING_MODE command.
If you are in INT mode, then after the messages have been displayed, you are given control. The prompt site-name: appears on your terminal. If you are in CONS mode, then after the messages have been displayed, the selected site retains control. To obtain control, do the following:
 - a) Issue a break at your terminal. This break is handled by RMOS which returns you to RMOS command level.
 - b) Enter an INTERRUPT command to send a break to the selected site.
 - c) The site: prompt appears and you are given control.
2. If RMOS is operating in MESSAGE mode, communication is directly established between the RMOS terminal and the application on the target site. Any waiting messages in the application mailbox are sent to the RMOS by the target application and displayed on the RMOS Operator terminal.

To terminate the selection of a selected site, you issue a break (which RMOS handles and returns you to RMOS command level), and enter any RMOS command except INTERRUPT.



9.2.9 UNLK (UNLINK)

Purpose

Specifies the name of the user associated with the link to be deleted.

Syntax

```
{ UNLINK }
{ UNLK   }

SITE = sitenm

[ TYPE = PT ]

[ USER = usernm ]

[ { APPLICATION }           ]
[ {                   } = name8 ]
[ { APPL         }           ]
```

Parameters

SITE	Specifies the name of the site whose link is to be deleted. You enter the name of a linked site, that is, you must have previously executed a LINK command for this site during the current RMOS session. If necessary, the link can be more fully identified using the USER and APPLICATION parameters. You must enter a value for SITE; there is no default.
TYPE OF LINK	Specifies the type of link to be deleted. The only type available in the current version of RMOS is TYPE = PT (Pass Through). This is the default value. Other values will be added in future versions.



USER	Specifies the name of user associated with the link to be deleted. You enter the user name given when the link was made using the LINK command. If you omit USER, the user name associated with the only existing link to the site (as given via the SITE parameter) is assumed. If there is more than one existing link to SITE, then there is no default value for USER. In this case, if you omit USER, the UNLINK command is not executed.
APPLICATION	The application (abbreviation APPL) parameter specifies the name of the application associated with the link to be deleted by the UNLINK command. You enter the application name given when the link was established (using the LINK command). If you omit APPLICATION, the application associated with the only existing link to the site (as given via the SITE parameter) is assumed. If there is more than one existing link to SITE, then there is no default value for APPLICATION. IOF is the default application.

Examples

UNLINK SITE = MINE ;	(Delete the link with the site named MINE Type PT link is assumed. As USER and APPLICATION have been omitted, there must be only one existing link with MINE.)
UNLINK SITE = OURS USER = ME APPLICATION = TDSZ ;	(Delete the link with the site name OURS and identified by the user named ME and the application named TDSZ.)



9.3 Leaving RMOS

To leave RMOS, enter the QUIT (Q) or the slash (/) character. All active links at termination time are closed, and their associated subfile if any, is then closed and the contents erased if it is a temporary file. See below:

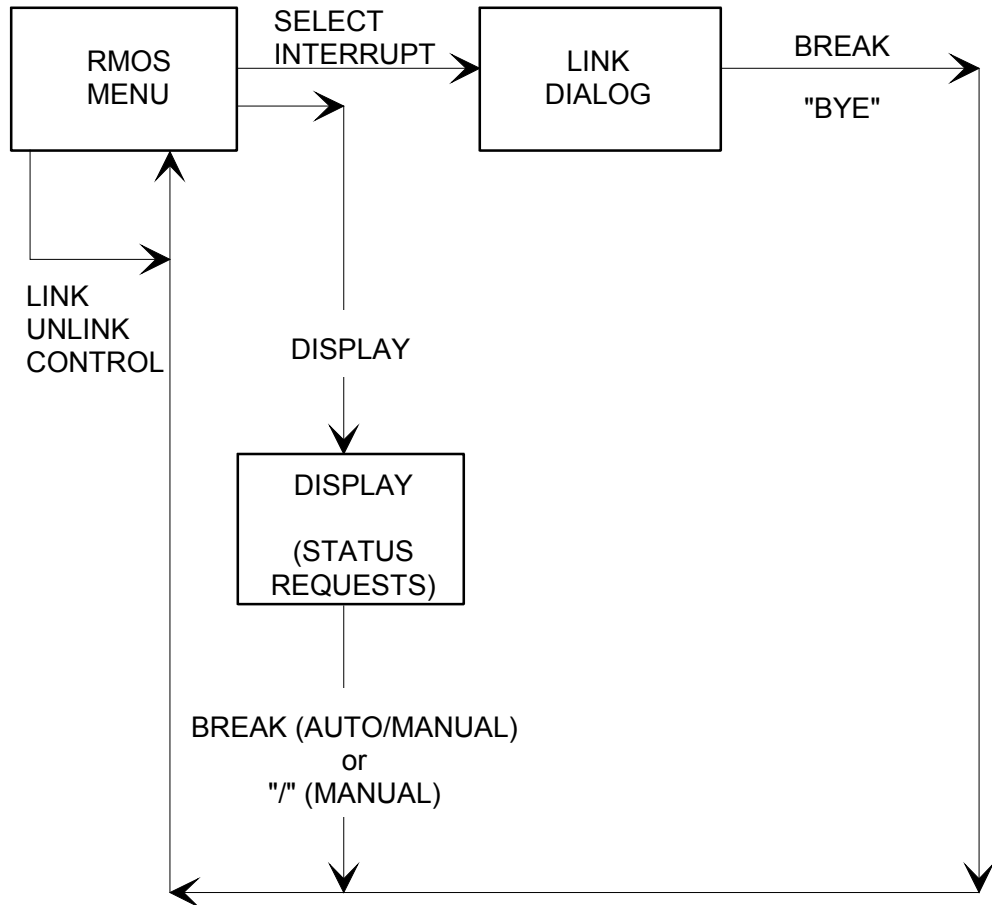


Figure 9-1. Screen Format Transitions



Field Descriptions

n	Number of the current page.
N	Total number of pages of the display (max. is 99).
AUTO	The display is automatically managed according to the time interval specified through the CONTROL command (default is 60 seconds). You control the screen manually by entering commands in a two-character control field. See Section 9.2.2 <i>CTL (CONTROL)</i> .
SITE	Name of the site to which the link is made.
LINK TYPE	Type of link is Pass Through (PT).
USER	User name under which this link has been established.
APPL.	Application to which this link has been made.
STATUS	Gives the current status of the link. See <i>Figure 9-3</i> below. OPENING A LINK command has been issued but actual connection acknowledgement has not been received yet. ACTIVE Link is operating normally. CLOSING The link is being closed either normally (UNLINK command) or abnormally. FAILED The link has been abnormally disconnected. RMOS attempts to reconnect automatically five times.
#MESSAGES	Gives the number of messages stored in the subfile when RMOS is operating in file mode. If RMOS is operating in message mode, the presence of "WAITING" in this field indicates that a message is ready to be received when the link is selected. While the link waits for a response from the operator, this number is highlighted. No more messages are sent over the link until the operator replies.
TOTAL NUMBER OF LINKED SITES	Gives the total number of known links whatever their status.



FILE SPACE AVAILABILITY

Indicates that RMOS is operating in file mode and gives the percentage of free space in the used file. The free available space relates to the current file size. No provision is made for any future increase in file size.

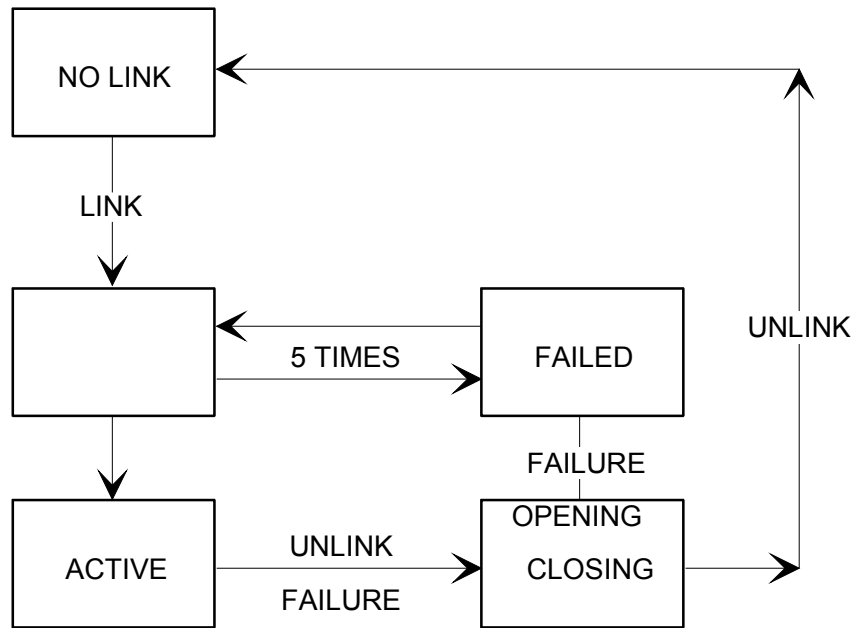


Figure 9-3. Link Status Transitions



9.4.2 Global Requests Screen Layout

This screen is displayed after you enter the command

DISPLAY OPTION = GLOBAL_REQUESTS.

The screen shows you the list of REQUESTS sent by all LINKS for the current RMOS session and consists of one or several pages.

```
n/N          DISPLAY GLOBAL REQUESTS          AUTO
                                                    (or)
                                                    --> : _

WARNING : MESSAGE FLOW INTERRUPTED FOR XXX LINKS

** DV99 -----
* 13.11 -----
----- SITE : ssxx      USER : syso      APPL : IOF -----

17 /QUESTIONS FROM ----- LINE 1 -----
      ----- LINE 2 -----

** 13.56 -----
----- SITE : ssyy      USER : -----
-----
-----
-----

+ + +
```

Figure 9-4. Display Global Requests



Field Descriptions

n	number of the current page.
N	total number of pages of the display (max. is 99).
AUTO	The display is automatically managed according to the time interval specified through the CONTROL command (default is 60 seconds). You control the screen manually by entering commands in a two-character control field. For more information, see the explanation of the CTL command.
SITE	Name of the site to which the link is made.
USER	User name under which the link has been established.
APPL.	Application to which this link has been made.

"WARNING: MESSAGE FLOW INTERRUPTED FOR XXX LINKS"

The second line indicates that a number of XXX links (>0) have stopped sending any more messages. These links are waiting for a response from the operator. For example,

- page synchronization: "+++"
- at system level: "S:"
- ...

On the global-status screen, the message count for such links is shown in reverse video.

NOTE:

To have a continuous flow of messages on a link in console mode, set PAGE MODE = 0 in the GCL user's profile.

The other lines contain the requests (on one or several lines for each request) exactly as they are sent by the links.

Catalogued requests:	* AA11.....
(key)	** AA11..... (repeated)
Uncataloged requests:	* 11.11.....
(hour)	** 11.11..... (repeated)
Questions:	11/..... (question number)



- Only links with pending requests are shown on the screen.
- If no request is pending for any link, the screen is empty: the blinking line "NO PENDING REQUESTS" appears.
- On the screen, links are sorted by alphabetical order.
- Only the beginning (*..., 11/) of the requests is recognized.
- A request is maintained for about four minutes on the screen, regardless of the action(s) that the operator may carry out to satisfy it. Whether the request is satisfied or not, it remains displayed on the screen.
- An unsatisfied request is repeated every three minutes by the link.
- This repeated occurrence activates the request for a further four minute period on the screen.
- A request which has not been repeated within four minutes is discarded. It no longer appears on the screen. This is the case for a waiting link whose message flow has been interrupted.

Note that filters must first be set up on sites to which a link is being made so that requests appear on the Global-Requests screen.



9.5 RMOS Error Messages

9.5.1 Initialization Error Messages

RMOS may issue the following error messages at initialization time. In some cases a return code can help detect the cause of the error. RMOS terminates after issuing the error message.

RM01 RMOS INITIALIZATION ERROR: NO SEMAPHORE AVAILABLE
RC = return-code

Meaning software error.

Action Contact your Service Center.

RM02 RMOS INITIALIZATION ERROR: NO WORKSTATION AVAILABLE

Action Check that you are not attempting to start more than 10 RMOS processors; otherwise contact your Service Center.

RM03 RMOS INITIALIZATION ERROR: UNABLE TO READ DIRECTIVE

Meaning software error.

Action Contact your Service Center.

RM04 RMOS INITIALIZATION ERROR: UNABLE TO CREATE BUFFER
RC = return-code

Meaning Software error.

Action Contact your Service Center.

RM05 RMOS INITIALIZATION ERROR: UNABLE TO GET ENVIRONMENT
RC = return-code

Meaning software error.

Action Contact your Service Center.



RM06 RMOS INITIALIZATION ERROR: UNABLE TO GET CURRENT
JOB IDENTIFICATION
RC = return-code

Meaning software error.

Action Contact your Service Center.

RM07 RMOS INITIALIZATION ERROR: UNABLE TO GET CURRENT
USER IDENTIFICATION
RC = return-code

Meaning software error.

Action Contact your Service Center.

RM08 RMOS INITIALIZATION ERROR: UNABLE TO GET CURRENT
SITE IDENTIFICATION
RC = return-code

Meaning software error.

Action Contact your Service Center.

RM10 RMOS INITIALIZATION ERROR: UNABLE TO CREATE RMOS DOMAIN
RC = return-code

Meaning software error.

Action Contact your Service Center.

RM11 RMOS INITIALIZATION ERROR: UNABLE TO START PROCESS p1
RC = return code

Meaning software error.

Action Contact your Service Center.



9.5.2 Command Processing Error Messages

RMOS issues these error messages whenever an RMOS command applies to an unknown object. Do not confuse them with the normal GCL command error messages.

RM22 LINK ALREADY CONNECTED: COMMAND DISCARDED

Meaning This link already exists.

Action Check that the link exists by using the DISPLAY command and use the correct command.

RM23 NO MORE SPACE AVAILABLE IN SESSION TABLE: COMMAND DISCARDED

Meaning A LINK command has been issued and there is no more space in the user session table.

Action Re-issue the command after terminating at least one previous user session.

RM24 LINK UNKNOWN

Meaning The specified link does not exist.

Action Check that the link exists.

RM25 LINK MUST BE DEFINED MORE PRECISELY

Meaning The LINK name specified in the command is ambiguous.

Action Specify precisely the user and/or application.

RM26 NO SELECTED SESSION

Meaning An INTERRUPT command has been issued and there is no currently selected link. Command makes no sense.

Action None.

RM27 LINK STATE "FAILED"

Meaning A SELECT command has been issued on a link whose status is "failed".

Action Re-issue command when the link has been successfully reconnected.



9.5.3 VCAM Error Messages

RM31 LINK site/user/application
CONNECTION REJECTED - reason

This message is issued when a LINK command cannot be successfully executed.
The reasons for rejection are:

STATUS = V_REJECT (01)

Action None.

SYSTEM RESOURCE OVERLOAD (02)

Action None.

STATUS = V_ARGERR (03)

Action None.

LOCAL MAILBOX SATURATED (04)

Action None.

DESTINATION NOT SPECIFIED (05)

Action None.

STATUS = V_SYSTERR (06)

Action None.

BY REMOTE APPLICATION (07)

Action Check the status of the remote network system.

STATUS = V_ABNREJCT (08)

Action None.

DESTINATION NODE NOT OPERABLE (09)

Action Check the status of the remote network system.

DESTINATION NODE SATURATED (10)

Action None.

DESTINATION MAILBOX UNKNOWN (11)

Action Remote application of a LINK command is not known on the remote site.

DESTINATION MAILBOX NOT OPERABLE (12)

DESTINATION MAILBOX SATURATED (13)

DESTINATION APPLICATION SATURATED (14)

DIALOG REJECTION (NEGOTIATION RESULT) (15)

PRESENTATION REJECTION (NEGOTIATION RESULT) (16)

Action (12-16) Check the status of the remote network system.



TIME OUT (17)

Action Check the status of the Front End.

ACCESS RIGHTS VIOLATION (18)

Action Check the access rights of the specified USER/PROJECT/BILLING to the system/application on the remote site.

SECURITY CHECKS FAILED (19)

Action Enter the correct password at link time.

DESTINATION NODE UNKNOWN (20)

PATH TO DESTINATION NOT AVAILABLE (21)

Action (20-21) Check the network generation parameter against the actual network configuration.

USER ALREADY LOGGED (22)

Action None.

DUPLICATE STATION IDENTIFIER (23)

Action None.

STATUS = V_CLSREJCT (24)

Action None.

RM32 LINK site/user/application
UNABLE TO CLOSE SESSION

Meaning Link is removed from RMOS link table.

Action None.



RM33 LINK site/user/application
UNABLE TO RECEIVE DATA - SESSION IS DISCONNECTED (01)
UNABLE TO SEND DATA - SESSION IS DISCONNECTED (02)
UNABLE TO RECEIVE IT - SESSION IS DISCONNECTED (03)
UNABLE TO SEND IT - SESSION IS DISCONNECTED (04)

Meaning An abnormal termination of the specified operation has occurred. An attempt is made to terminate the link session and then to reconnect automatically.

Action None. If the problem persists, contact your Service Center.

RM34 LINK site/user/appl
REMOTE APPLICATION ABNORMALLY DISCONNECTED

Meaning An abnormal disconnection has been received by VCAM. The connection is interrupted. RMOS will attempt to reconnect the link 5 times (once every 6 minutes).

Action None. If the problem persists, contact your Service Center.



9.5.4 File Processing Error Messages

RM09 RMOS MODE IS FILE: TEMPORARY FILE IS USED
RC = return-code (01)

Meaning Issued when RMOS is started using MODE = FILE option and the specified file is not found. In this case, a temporary file is created for the duration of the RMOS session.

Action None. If you wish to use a permanent file, terminate this RMOS session and restart using a valid file name.

RM09 RMOS MODE IS MESSAGE: UNABLE TO ASSIGN FILE
RC = return-code (03)

RM09 RMOS MODE IS MESSAGE: UNABLE TO CREATE FILE
RC = return code (02)

RM09 RMOS MODE IS MESSAGE: UNABLE TO ALLOCATE FILE
RC = return-code (04)

RM09 RMOS MODE IS MESSAGE: UNABLE TO OPEN FILE
RC = return-code (05)

Meaning (02-05) Because of file problems, RMOS will operate in MESSAGE mode.

Action (02-05) If a file is needed, use the return code to detect the cause of the error (see the *Error Messages and Return Codes Manual*).

RM53 UNABLE TO EXTEND RMOS FILE: SOME MESSAGES MAY BE
LOST
RC = return-code

Meaning The maximum file size limit has been reached and the file cannot be extended any more.

Action Delete the oldest messages in the largest subfiles for links which have not been selected.



RM45 LINK site/user/application
UNABLE TO OPEN SUBFILE
RC = return-code (01)

RM45 LINK site/user/application
UNABLE TO CLOSE SUBFILE
RC = return-code (02)

RM45 LINK site/user/application
UNABLE TO READ DATA FROM SUBFILE
RC = return-code (03)

RM45 LINK site/user/application
UNABLE TO WRITE DATA IN SUBFILE
RC = return-code (04)

Meaning (01-04) These error messages denote a problem in the subfile associated with the specified link. RMOS disables the file and will operate in MESSAGE mode. Note that if the incident occurred during a read/write operation, some messages may be lost.

Action (01-04) Check the cause of the error using the *Error Messages and Return Codes manual* (file management).



9.5.5 Miscellaneous Error Messages

RM20 RMOS INTERNAL ERROR
RC = return-code

Meaning An unexpected return code has been encountered within processing. Although processing is normally not interrupted, depending on the type of error, the results obtained subsequently may not be those expected.

Action You are recommended to abort RMOS activity and take a dump.

RM21 UNABLE TO READ GCL COMMAND
RC = return-code

Meaning An abnormal return code has been encountered in a GCL read procedure.

Action Check return-code value and relevant explanations against GCL manuals.

RM55 TERMINAL TYPE NOT SUPPORTED
RC = return-code

Meaning Self explanatory.

Action RMOS must be operated from a FORM compatible terminal.

RM56 UNEXPECTED RETURN CODE IN FORM PRIMITIVE
RC = return-code

Meaning RMOS malfunction.

Action Processing may continue provided that the DISPLAY command is not used.

RM57 DO YOU REALLY WANT TO QUIT RMOS ?
(PLEASE ANSWER YES OR NO)

Meaning Request confirmation for quitting RMOS.

Action Answer "YES" or "NO".



10. Reconfiguration

10.1 DPS 7000 Reconfiguration

Reconfiguration for DPS 7000 until TA models included, is described in this chapter. The reconfiguration on DPS 7000/XTA and following models is presented in chapter 10.2.

10.1.1 Overview

The larger DPS 7000 systems with redundant components are very reliable due to the Reconfiguration product. Reconfiguration takes place automatically when there is a malfunction, and in other circumstances can be performed manually by the operator.

It is possible to remove certain failed resources from the configuration without stopping processing.

The system components can be acted upon in two circumstances:

1. At system initialization (**Static Configuration**)

A component can be excluded:

- either automatically when the component cannot be initialized
- or upon operator request for example when intermittent faults are not detected at initialization time or when a component needs to be reserved for a diagnostic operation at a future time;

In bisystem mode the system components of the installation are distributed between the two systems via two reservation lists;



2. When the system is running (**Dynamic Reconfiguration**)

- upon component failure, if a component becomes not operational it is automatically isolated. On redundant configurations the system remains running when possible. For example if a path to a device is broken because one of the components in the path becomes not operational an alternate path is found.
- upon operator request, for example:
 - to isolate a component where too many non fatal incidents arise,
 - to exclude a component from a system to allocate it to the other system in a bisystem configuration,
 - to insert a component in a system.

For Reconfiguration purposes, a clear distinction must be made between a system and an installation :

- A system is a set of resources controlled by one GCOS 7 Operating System. This includes Central Subsystem resources and Peripheral Subsystem resources.
- An installation covers a DPS 7000 computer with peripherals which are connected to it. An installation is described by an Installation Resource Table (IRT).

A redundant installation can be used in two ways :

- all the resources of the installation are put together and controlled by a single GCOS 7 Operating System : mono-system mode,
- Central Subsystem resources are split in two parts; each part running a separate GCOS 7 Operating system : bi-system mode.
Peripheral resources may be accessible by only one system or by both systems depending on the physical connections.



10.1.1.1 System Architecture and Resources

The general architecture of the system from the Dynamic Reconfiguration point of view is the following:

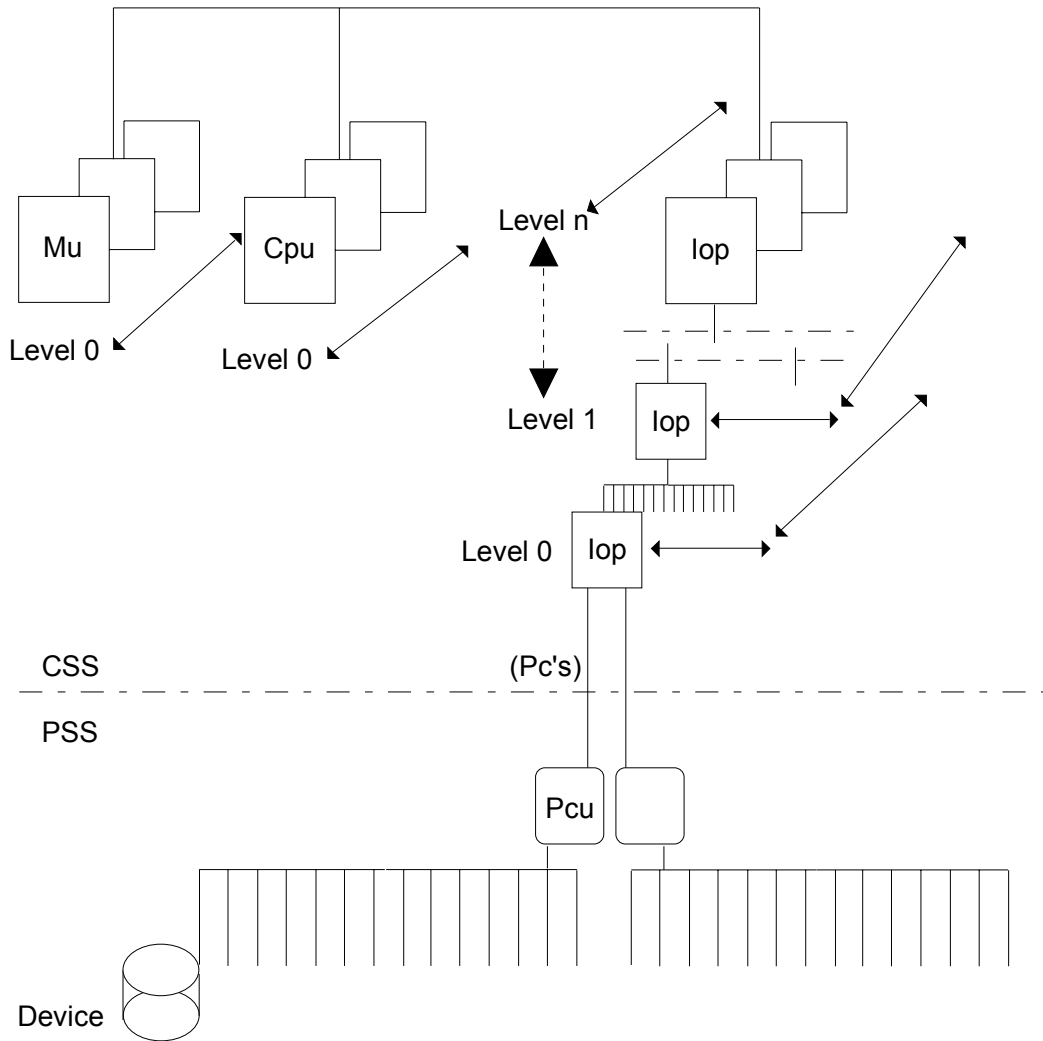


Figure 10-1. System Architecture

The system is structured in two major parts:

- the Central SubSystem (CSS),
- the Peripheral SubSystems (PSS).

The Peripheral SubSystems are connected to the Central SubSystem through Physical Channels (PC).



Central SubSystem

It is made of the following resource types:

- Main Memory Units (MU).
- Central Processors Units (CPU).
- Input-Output Processors (IOP).

The redundancy of major resources provides a better availability.

Some resource classes may be organized into several levels. In this case every action performed at a given level is propagated at the lower levels and to the PSSs which are connected to it.

Peripheral SubSystem

The Peripheral SubSystem includes:

- the controllers: Peripheral Control Units (PCU),
- the peripheral devices.

There exists several types of controllers, each of them managing different kinds of peripheral devices. A controller is referred to by its external identifier which has the following format:

- MCxx for Mass storage Controllers which manage disk devices.
- TCxx for Tape Controllers which manage magnetic tape units or cartridges.
- UCxx for Unit record Controllers which manage "slow" peripherals: printers or local telecom lines.
- CCxx for Communication Controllers which allow the connection of Datanets

A peripheral device can be connected to several PCUs, thus providing several access paths to it from the CSS. This allows to balance the IOs between the access paths according to the actual load of the various components in the access path, and provides a better availability in case of failure of a component in an access path: an alternate path is automatically chosen by the system.



10.1.1.2 When and How to Configure or Reconfigure a System

Static Configuration of the CSS resources takes place at system initialization. The detailed operating actions depend on the machine, they are described in the appropriate *Operator's Guides*:

- 77 A1 71UU for DPS 7000/Ax.
- 77 A1 54UU for DPS 7000/2xx and DPS 7000/3xx.
- 77 A1 81UU for DPS 7000/5xx and DPS 7000/7xx.
- 77 A1 65US for DPS 7000/4xx/Cxx/8xx/Dxx/Mxx and DPS 7000/MTxx.

The Dynamic Reconfiguration of the CSS resources takes place when the system is running.

On DPS 7000/Ax, DPS 7000/2xx, DPS 7000/3xx, reconfiguration is performed through Service Processor commands and is described in the appropriate manuals:

- 77 A1 71UU for DPS 7000/Ax.
- 77 A1 54UU for DPS 7000/2xx and DPS 7000/3xx.

On DPS 7000/4xx, DPS 7000/5xx, DPS 7000/7xx, DPS 7000/8xx, and DPS 7000/MTxx, reconfiguration is performed using GCOS 7 commands as described in this section of the manual.

The Reconfiguration of PSS components:

- is always performed when the system is running,
- is performed through GCOS 7 commands,
- is described in the present Chapter of this manual.



10.1.2 Reconfiguration of Central Subsystem

The Dynamic Reconfiguration takes place when the system is running; it is activated:

- either automatically upon detection of an Hardware incident,
- or upon operator command.

The result of the reconfiguration depends upon the type of the resource and the state of the resource at the time the reconfiguration occurred.

10.1.2.1 CSS Resources

Every resource has an external identifier which is used to designate it in operator commands and messages. The general format of the external identifier is YYxx where YY designates a resource class and xx designates a resource name (number) inside the class.

Central Processor Units

There are several types of Central Processor Units (see also Section 5.10 *Domain and CDP, Customer Dedicated Processor, Overview*):

- **IPxx**, the Standard DPS 7000 Processors,
- **EPxx**, type 1 extended processors, the High Relational Performance Processors present on some DPS 7000 systems
- **FPxx**, type 2 extended processors, and **GPxx**, type 3 extended processors.

Every CPU can be inserted or removed; however, since the system will remain operational only if at least one IPxx is available, the MDHW command refuses to remove the last IPU.

Main Memory Units

The Software provides a logical view of the Main Memory through Memory Units (**MUxx**). An MUxx is a set of contiguous memory space from the addressing point of view.

The Service Processor is used to manipulate physical memory boards (HMxx). You can use the CFG utility of the Service Processor at system initialization to add an HMxx to the configuration of a system, remove it, or exclude it from the configuration.



There is no direct correspondence between the MUs and the HMs because a MUxx may be spread over several physical memory boards.

On the DPS 7000/8x0/Dx0/Mx0/MTxx, however, there is a direct correspondence between the MUs and the HMs.

Example

- Assume that HM02, HM04, HM06, HM08 are present in the installation.
- System S0 has been configured with HM02 and HM04.
- HM06 and HM08 have been excluded.

The DHW MU command on S0 system displays the following:

```
MU00  IN
MU01  OUT  (NOT ACCESSIBLE)
MU02  OUT  (NOT ACCESSIBLE)
MU03  OUT  (NOT ACCESSIBLE)
```

MU02 and MU03 are not accessible because they correspond to HM06 and HM08 which have been excluded.

Actually the two physical memory boards HM02 and HM04 are mapped on to MU00 memory unit in the allocated (IN) state. The MU01 memory unit is a "place holder" unit and cannot be manipulated by the operator commands.

MU01 is displayed as "not accessible" because it is spread within HM02 and HM04 memory boards which are interleaved.

The mapping of HMs on to MUs can dynamically change.

To insert the HM06 memory board after repair, the exclusion list must be modified by the CFG utility of the Service Processor in order to remove HM06 from the exclusion list; a following DHW command issued on GCOS 7 will display:

```
MU00  IN
MU01  OUT  (NOT ACCESSIBLE)
MU02  OUT  (ALLOCATABLE)
MU03  OUT  (NOT ACCESSIBLE)
```

A MDHW MU02 IN will make the memory unit available to GCOS 7.

A MUxx resource cannot be dynamically removed from a system.



Input Output Processors

There are two types of Input Output processors from the Software point of view:

- IOxx** Input Output controllers.
- PXxx** Physical channel multipleXed units.

They are organized hierarchically in two levels. The IOxx are usually at the higher level. Depending on the DPS 7000 series, one of the two levels may be absent.

When a reconfiguration action (manual or automatic) takes place for a resource, the action is propagated on to the lower levels up to the physical device.

On DP 7000/4xx/5x0/7xx/8x0/Dx0/Mx0/MTxx series, the hardware architecture makes a distinction between:

- the proper CSS components : IPxx, EPxx, FPxx, GPxx, HMxx gathered into the Central SubSystem (CSS),
- the Input Output Components: IUxx, XAxx, MBxx, PXxx gathered into the Input Output SubSystem (IOSS).

IOPs are allocated to a system at IOSS level i.e. the IOxx and the related PXxx. Its is not possible to allocate an IOxx to system S0 and a related PXxx to system S1.

Refer to the relevant *Operator's Guide* for a detailed definition of the hardware components.

The GCOS 7 software provides a simplified view of the Input Output SubSystem in order to focus on the major resources from the reconfiguration point of view. Roughly

- IOxx resource corresponds to the XAxx hardware component,
- PXxx resource corresponds to the PXxx hardware component.



10.1.2.2 Resource States

The main states of Central SubSystem resources are listed hereafter. For each of them, a description of the state is given as well as the result of a MDHW command.

IN

Meaning: The resource is allocated to the system

OUT action: The resource is made unavailable to the system. Depending on the resource type and the context, the OUT action may be accepted or rejected.

OUT

Meaning: The Unit is not allocated to the system

IN action: The resource is made available to the system if it can be successfully initialized.

OUT(BY SYSTEM)

Meaning: The Unit is not allocated to the system; it has been deallocated by the system because of an abnormal event.

IN action: The resource is made available to the system if it can be successfully initialized.

OUT(BY OPERATOR)

Meaning: The Unit is not allocated to the system; it has been deallocated by an operator command MDHW.

IN action: The resource is made available to the system if it can be successfully initialized.

OUT(FATHER OUT)

Meaning: The Unit is not allocated to the system, it has been deallocated because one of the upper resource is deallocated.

IN action: The request is rejected as long as the upper resource is deallocated.



OUT(NOT ACCESSIBLE)

Meaning: The Unit is not allocated to the system, usually it has been inserted in the exclusion list.

IN action: rejected

OUT(ALLOCATED TO ANOTHER SYSTEM)

Meaning: The Unit is not allocated to the system, it is allocated to the other system in a bisystem configuration.

IN action: rejected

OUT(NOT INSTALLED)

Meaning: The Unit is not present in the installation

IN action: rejected



10.1.2.3 Machine Incidents

The GCOS 7 Software is warned by the Hardware of any abnormal event. The Software analyzes the error and tries to recover it keeping the system running possibly in a degraded mode.

For every error, a message is sent to the operator and the event is logged in the SYS.ERLOG file. Some events will require a maintenance action.

If the reported event is a permanent failure:

- the failed unit is deallocated as well as the downward related resources,
- the system performs recovery actions. The level of success of recovery actions depends upon the nature and the context of the incident.

Examples

When a CPU becomes unavailable, the system tries to resume on another CPU the execution of the process which was running on the faulty CPU provided that there exists at least two CPUs in the configuration:

- if the context of the process can be retrieved, the process execution will be resumed on another CPU,
- if the context of the process cannot be retrieved, and the process was not executing critical software system functions the process is aborted,
- if the context of the process cannot be retrieved and the process was executing critical software system functions, a System Crash is issued.

When a failure arises in an IOP, the failure may be

- a total failure. In this case the IOP is deallocated as well as the downward related resources (controllers and devices) if the failed IOP was in all the active paths to the resource,
- a partial failure. In this case the system starts a recovery sequence for each downward related controller. If at least one controller is recovered, the IOP stays in the current system, otherwise it is removed.

The Console Messages Directory (47 A2 61UU) gives an explanation of the messages with the prefixes AV and AW that are issued to report the events to the operator console.



10.1.2.4 GCOS 7 Reconfiguration Commands (MDHW and DHW)

The MODIFY_HARDWARE (MDHW) and DISPLAY_HARDWARE (DHW) commands are used by the operator to reconfigure system hardware, after initialization, while the system is running under GCOS 7. This is known as **dynamic reconfiguration**. The commands cover two activities:

- **modification** of the state of hardware resources (MDHW command - Chapter 8 *Description of the Operator Commands*),
- **display** of the state of hardware resources (DHW command - Chapter 7 *Description of the Operator Commands*).

These commands apply to all hardware resources - Central Subsystem units, peripheral controllers and peripheral devices.

For more details, refer to the complete explanation of MDHW and DHW in Chapters 7 and 8 *Description of the Operator Commands*. *The Console Messages Directory* gives an explanation of the messages with the prefixes AV and AW that are issued concerning these commands.

10.1.2.5 Dynamic Reconfiguration and System Restart

When a system is stopped then restarted after previous dynamic reconfiguration operations have taken place, the following rules are applied:

- in the case of Reload, the system is restarted with the current configuration: dynamic reconfiguration operations remain valid.
- in the case of Reinitialization:
 - if the restart is automatic (REPEAT option of MDRSO command and INIT option of TSYs command), the system is restarted with the current configuration: dynamic reconfiguration operations remain valid
 - if the restart is manual, the reinitialization is always performed with the reservation list of CFG Service Processor utility.

Example

Assume that S0 system has been initialized with CPUs IP00 and IP01 in reservation list.

S1 system has been initialized with IP02 in reservation list.

IP01 has been dynamically reallocated to S1 system by MDHW commands on S0 and S1.



The IPxx allocated to S1 system after restart depends on the performed operations. the following table depicts the possible situations.

Table 10-1. Dynamic Reconfiguration

MDRSO option	S1 system stopped by	S1 system restarted by	allocated IPxx at restart	Comments
No Repeat	Crash or TSYS	syc RL	IP01, IP02	Reload with units allocated at the time the system was stopped.
No Repeat	Crash or TSYS	syc IN	IP01	Reinit with units in the reservation list.
No Repeat	TSYS INIT	Automatic	IP01, IP02	Reload with units allocated at the time the system was stopped. The NOREPEAT option of MDRSO command is meaningless.
Repeat	Crash or TSYS	Automatic	IP01, IP02	Automatic reload with units allocated at the time the system was stopped.
Repeat	TSYS INIT	Automatic	IP01, IP02	Automatic reinit with units allocated at the time the system was stopped.



10.1.3 Reconfiguration of Peripheral Subsystems

10.1.3.1 Peripheral SubSystem

In general terms, a Peripheral Subsystem is a set of devices and controllers.

The devices belong to the same family e.g. disk or magnetic tape.

There is one controller per Physical Channel.

Each device can be connected to one or two controllers.

A controller is connected to one Input/Output Processor.

This is a logical representation of peripherals. Depending on the nature of the peripheral the logical representation can be the same as or different from the physical architecture in terms of hardware components.

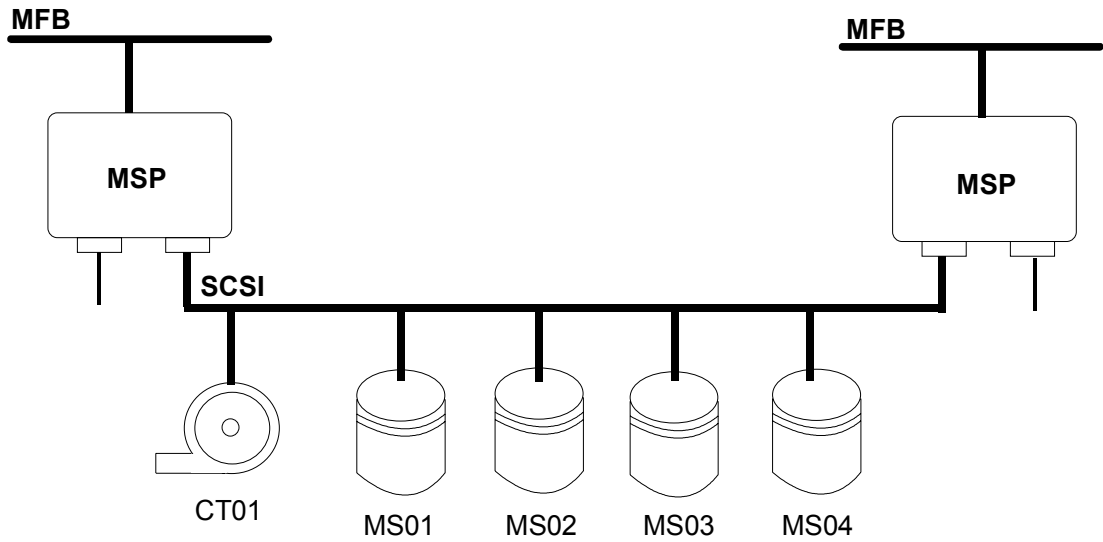
For example:

With a Bi-PSI controller, the mass storage subsystem is connected to the CSS through two physical channels, the physical controller is represented by two logical controllers, one for each physical channel.

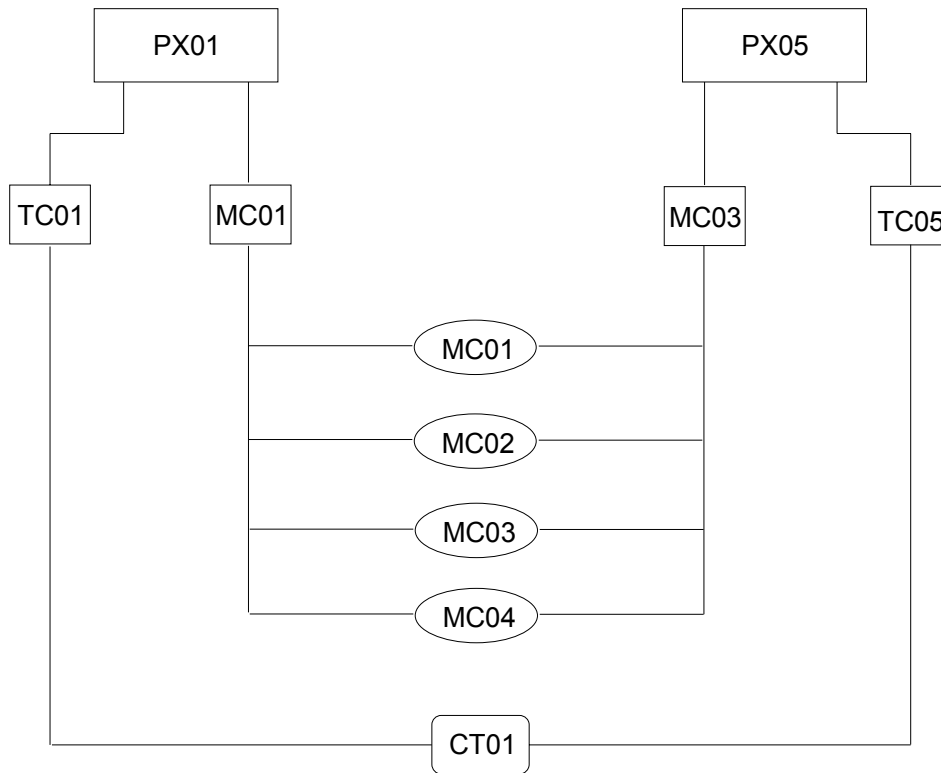
With MSP SCSI Subsystem:

- disks and cartridges can be connected on the same SCSI bus,
- MSP physical board has the IOP, controller and physical channel functions.

Figure 10-2 gives an example of physical representation and logical representation of peripherals on the same SCSI bus connected to two MSPs.



Physical Representation



Logical Representation

Figure 10-2. Peripherals - Physical and Logical



Concerning the Peripheral Subsystems, the following aspects must be considered:

- the physical connections of the devices to controllers and of the controllers to installations,
- the number of simultaneous Data Transfers which can be performed on a set of devices: Level Of Simultaneity (LOS),
- the number of access paths from a system to a peripheral device,
- the level of sharing of the volumes when the device can be accessed by several systems,
- the level of redundancy of the system.

10.1.3.2 Physical Connections

A device can be connected to 1 or 2 controllers.

Some Controllers can support up to 4 channel ports which connect him to one or several installations. In an installation, at most two channels can lead to the same controller.

In the description of an installation (Installation Resource Table: IRT) a controller is described as a set of logical controllers, one for each physical channel which connects the controller to the installation. Thus the IRT of the installation may provide a partial view of the Peripheral Subsystem.

Example : Cartridge Subsystem

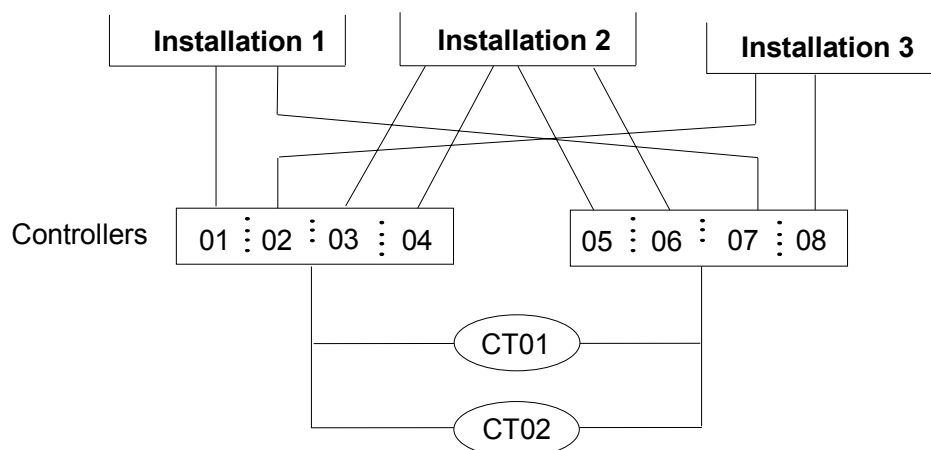


Figure 10-3. Example of Cartridge Subsystem



IRT of installation 1 declares	TC01, TC07, CT01, CT02	logical controllers.cartridge devices.
IRT of installation 2 declares	TC03, TC04, TC05, TC06, CT01, CT02	logical controllers.cartridge devices.
IRT of installation 3 declares	TC02, TC08, CT01, CT02	logical controllers.cartridge devices.

Remember that some installations can be split in 2 separate systems, each of them running their own GCOS 7 Operating System. The two systems of a bi-system installation use the same IRT, thus some devices or paths, although they are known by both systems, may be not accessible on one of the systems.

For Disk Subsystems:

- The maximum number of physical channels for a controller is 2.
- Each channel can be connected to the same installation or to 2 separate installations.
- A disk can be accessed by at most 2 systems belonging to the same installation or to 2 different installations.

For Tape Subsystems:

- The maximum number of physical channels for a controller is 1 (but see later *Dual LOS*).
- On some Tape Subsystems (MTS4572) a manual switch gives the possibility to connect the physical channel to one installation or to another one, thus making the set of tape devices alternatively available on the two installations.

For Cartridge Subsystems:

- The maximum number of physical channels for a controller is 4.
- At most 2 physical channels of a controller can be connected to one installation, the other channels can be used to connect the controller to other installations.



Level of Simultaneity (LOS)

Single LOS

Only one Data Transfer can be handled at a time. The peripheral devices are connected to only one controller.

Dual LOS

Two Data Transfers can be handled at a time on two different devices; two possibilities exist:

- Each device is connected to two controllers; (this is the only possibility for disk subsystems).
- Each device is connected to one controller but the controller is able to handle 2 simultaneous IOs and the controller is connected to the installation(s) by at least 2 physical channels.
In this case the controller is represented as 2 independent controllers with consecutive external identifiers (TCnn, TCnn+1).

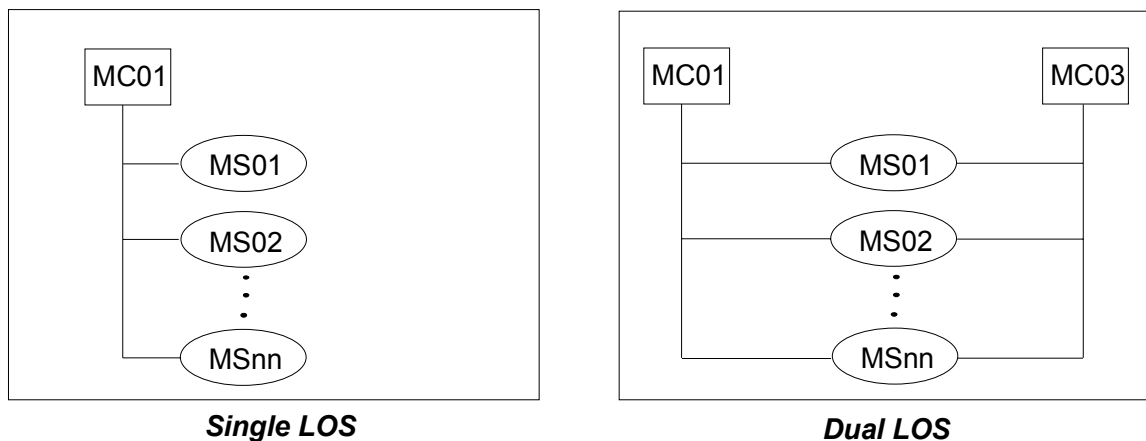


Figure 10-4. Level of Simultaneity (LOS)



Access Paths From a System To a Device

We consider here the access paths as seen by *one* system. The device may be also accessible by some other systems in the same installation or in independent installations.

Single Access

The device is accessed by only one path through a single controller.

Dual Access

The device is accessed by 2 paths; two possibilities exist:

- Through 2 independent controllers:
 - If a controller or physical channel fails, the device remains accessible.
 - The system launches the IOs on one path or the other one, depending on the actual load of the controllers.
- Through the same Bi-PSI controller:
 - This configuration is interesting on an installation which can be split in two separate systems.
 - If the controller fails, the device is no longer accessible. If one physical channel fails, the device remains accessible.
 - If the two paths are valid (IN state), the system launches the IOs always on the same path for a given device, the other path will be used if the first path becomes unavailable (SPARE path); some IOs can be launched on the physical channel implied in the SPARE path, they are related to another device.

More Than Two Access Paths

The device is accessed through two controllers, at least one controller is connected to the system by two physical channels.

If all the paths are valid, the system actually uses one physical channel on each controller to launch IOs on a given device; the other paths are SPARE for the device.



This facility is available only on Disk Subsystems and Cartridge Subsystems.

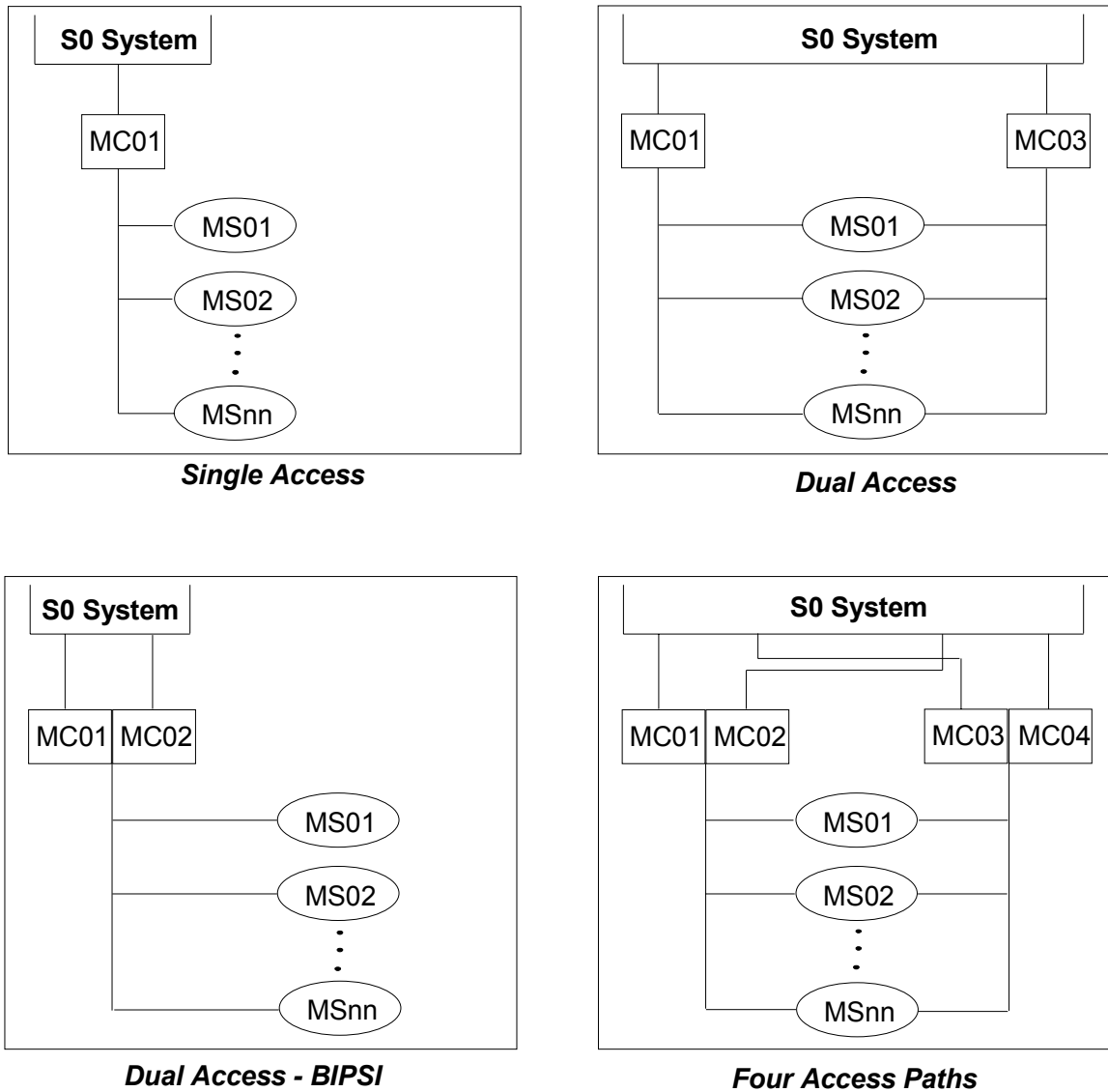


Figure 10-5. Access Paths



Device Sharing

A device and the associated volume are shared if they can be accessed by several systems. The systems can be on independent installations or can belong to the same installation in bi-system mode.

There exist two types of sharing:

- | | |
|------------------------|--|
| Static sharing | During a long period of time, the device is in exclusive use by one system. The other systems cannot access the device; the device is said "SEIZED BY ANOTHER SYSTEM".
This is the only valid type of sharing available for magnetic tapes and cartridge devices. |
| Dynamic sharing | The device can be accessed by two systems. An hardware mechanism prevents IOs on the same device to be simultaneously issued by two different systems. Software locks mechanisms at file level allow to guarantee the file consistency.
Dynamic sharing is available only for disk devices. |

Particular attention must be paid not to mix static sharing and dynamic sharing for the same volume; this could result in the system being completely tied up due to a deadlock.

Regarding the physical connections, device shareability is possible at several levels :

- sharing at controller level:
the device is connected to one controller and the controller is connected to two systems or more.
- sharing at device level:
the device is connected to two independent controllers each controller being connected to only one system.
- sharing at controller and device level:
the device is connected to two controllers and each controller is connected to two systems or more.

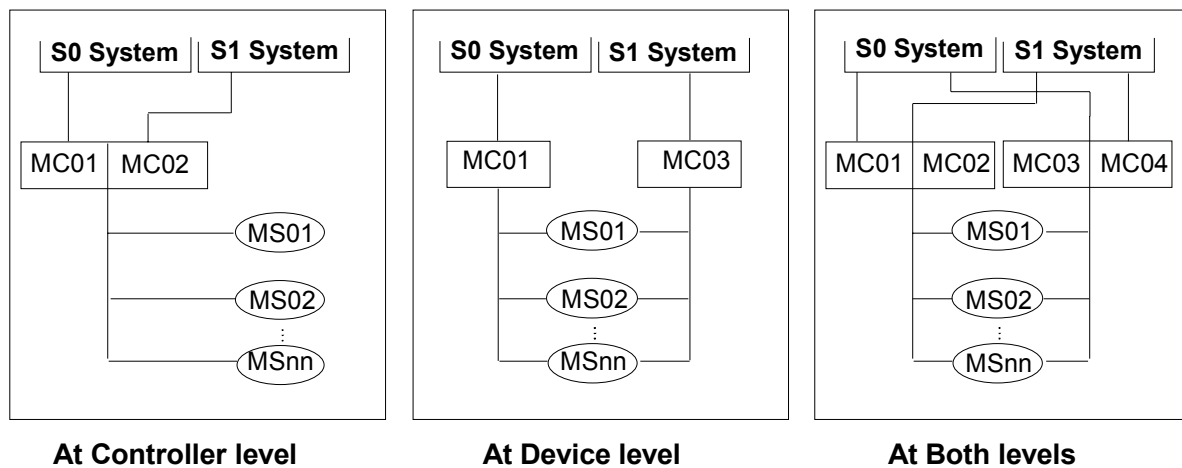


Figure 10-6. Device Sharing

For a detailed description of device sharing, refer to the *Coupled System User's Guide*.

Redundancy

When a unit fails, the reconfiguration is possible only if another unit of the same type can play the role of the failed unit. This is known as REDUNDANCY.

In the access to a device by a system, several units are usually implied: Input/Output processors, Controllers, Physical Channels. The availability of the system in case of unit failure depends on the degree of redundancy of all the units; when an alternate path to the device can be found whatever the failed unit, the system is said fully redundant.



10.1.3.3 Device States and Device Path States

To find out the status of Peripheral SubSystems, the operator uses the MDHW command. This command can display the status of the peripheral devices or the status of paths which lead to peripheral devices.

The state of a device indicates:

- if the device can be accessed by the system,
- the name of the media which is presently mounted on the device,
- what is the present usage of the media by the system.

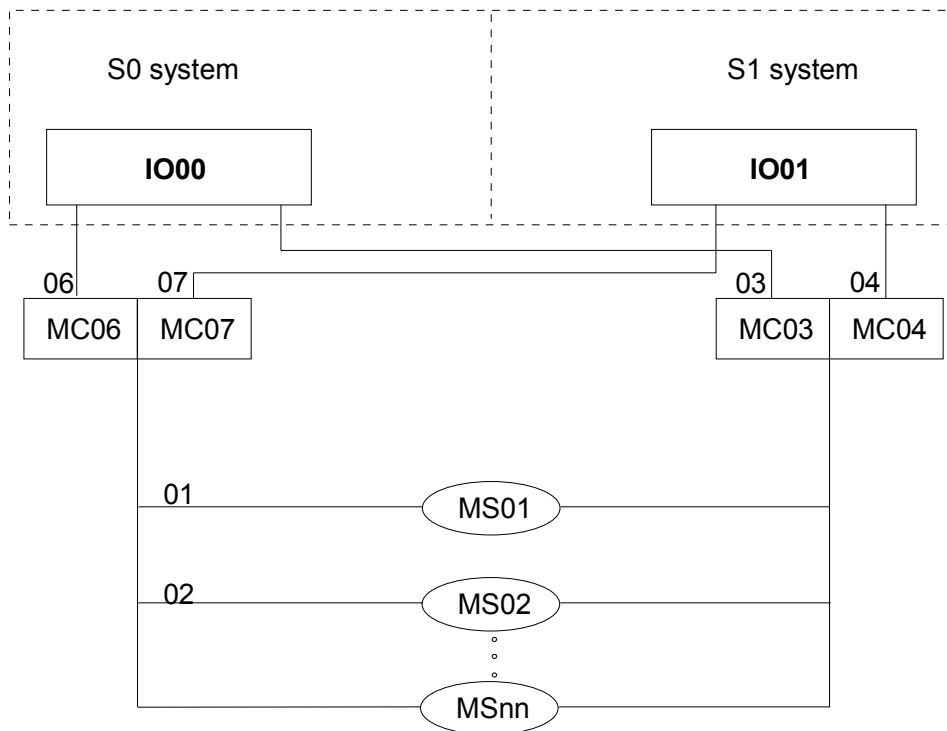
The device states are described in Chapter 6 *Peripheral Operations*.

The state of a device path indicates if a device path is usable or not, actually used or not, by a system. The possible device path states are:

- | | |
|-----------------------------|--|
| IN (READY) | The path is fully operational and currently usable by GCOS. |
| IN (SPARE) | The path is operational but is not currently used by GCOS. It corresponds to one of the two physical channels of a BiPSI controller. |
| OUT (BY OPERATOR) | The path is not usable; it was put out of the configuration by a MDHW xCnm.dvnm OUT command. |
| OUT (DEVICE OUT) | The path is not usable, the device dvnm being out of the configuration for maintenance purpose or because the device is statically shared and presently used by another system. |
| OUT (CONTROLLER OUT) | The path is not usable, the controller xCnm or an upper IOP being out of the configuration. |
| OUT (NOT ACCESSIBLE) | The path is not usable by the current system because a bisystem session has been initiated on the installation and the physical channel related to the path is connected to an IOP which has been allocated to the other system. |
| OUT (BY SYSTEM) | The path is out of the GCOS configuration due to a controller/device failure detected by the system. |



Example



Assume that:

- The installation has been initialized in a bi-system session.
- MS01 is dynamically shared between S0 and S1 systems.
- MS02 is in exclusive use by S1 system (device OUT on S0, device IN NSHR on S1).

The display of device paths on both systems S0 and S1 gives the following results:

On S0 system:

MC06.MS01 IN (READY)	MC03.MS01 IN (READY)
MC07.MS01 OUT (NOT ACCESSIBLE)	MC04.MS01 OUT (NOT ACCESSIBLE)
MC06.MS02 OUT (DEVICE OUT)	MC03.MS02 OUT (DEVICE OUT)
MC07.MS02 OUT (NOT ACCESSIBLE)	MC04.MS02 OUT (DEVICE OUT)



On S1 system:

MC06.MS01 OUT (NOT ACCESSIBLE)	MC03.MS01 OUT (NOT ACCESSIBLE)
MC07.MS01 IN (READY)	MC04.MS01 IN (READY)
MC06.MS02 OUT (NOT ACCESSIBLE)	MC03.MS02 OUT (NOT ACCESSIBLE)
MC07.MS02 IN (READY)	MC04.MS02 IN (READY)

10.1.3.4 Peripheral Incidents

The system behavior in case of peripheral incident depends on the nature of the incident and the peripheral configuration.

Some incidents require a manual intervention by the operator. For example when a volume is dismounted while being used by a running job.

In redundant configurations where a device is accessible by several paths, the system manages path switching automatically in the event of a fault occurring.

In case of an incident on the path to a device, the failed component (controller of physical channel) is isolated (OUT) if it cannot be recovered. When a SPARE path exists, it is made READY. The I/O operations under way are automatically taken up on an alternate path if such a path exists and if the I/O is re-executable.

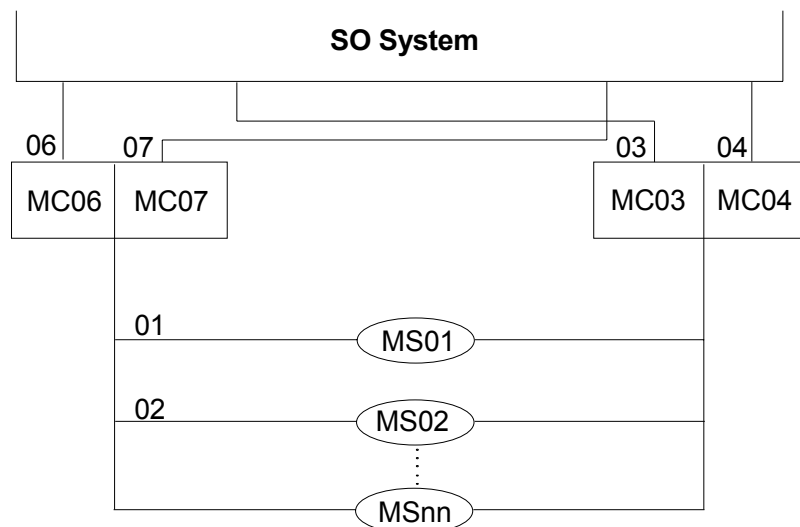
Note, however, that when a device is sharable, an interrupted I/O is never retried on an alternate path of the same system. This is because it cannot be guaranteed that the retry operation will not be interfered with by another I/O started from another system.

If the failed path was the last available one, the I/O terminates abnormally and the device is isolated (OUT).

The operator is warned by one or several messages of every incident and its consequences. The *Console Messages Directory* gives for each message a description of the possible reasons of the incident as well as the possible corrective actions.



Example



Assume the device MS01 is accessible by 4 access paths through 2 Bi-PSI controllers.

The DHW MS01 PATH=1 command gives the following result:

```
MC06.MS01    IN (READY)
MC07.MS01    IN (SPARE)
MC03.MS01    IN (READY)
MC04.MS01    IN (SPARE)
```

If a serious failure makes the first controller (MC06/MC07) unavailable, the following messages are displayed:

```
(AV41) hh.mm MC06    PSI06 FAILED
(AV44) hh.mm MC06    OUT (BY SYSTEM)
(AV41) hh.mm MC07    PSI07 FAILED
(AV44) hh.mm MC07    OUT (BY SYSTEM)
```

The MS01 device remains accessible through the second controller (MC03/MC04).

A display of the MS01 device paths gives the following results:

```
MC06.MS01    OUT (CONTROLLER OUT)
MC07.MS01    OUT (CONTROLLER OUT)
MC03.MS01    IN (READY)
MC04.MS01    IN (SPARE)
```

The OLTDs can be used to diagnose the failed unit and the operator can use the MDHW IN command to reinsert the unit after it has been repaired.



When a device becomes inaccessible while being used, the following situations can arise:

- The volume which was mounted on the device is removable.
The volume can be dismounted and mounted on another device of the same type: the processing of the job which was using the volume will automatically resume. If the operator does not want to mount the volume on another device, he can issue a CANCEL REQUEST (CR) command in response to the RECOVER request.
- The volume which was mounted on the device is not removable and contains only user files.
The jobs which were using the device are aborted.
- The volume which was mounted on the device is not removable and contains BKST files.
The system will terminate with a system crash. A dump is available if the implied volume does not contain the system backing store.

10.1.3.5 GCOS 7 Reconfiguration Commands (DHW and MDHW)

The MODIFY_HARDWARE (MDHW) and DISPLAY_HARDWARE (DHW) commands are used by the operator to act upon peripheral elements : controllers/ physical channels, device paths and devices. The commands cover two activities:

- **modification** of the state of hardware resources (MDHW command - Chapter 8 *Description of the Operator Commands*),
- **display** of the state of hardware resources (DHW command - Chapter 7 *Description of the Operator Commands*).

The MDHW command can be used to :

- isolate for diagnostic purpose a unit or path where too many incidents arise,
- insert a unit after it had been repaired,
- modify the type of sharing of a volume in a Coupled System environment.

The modification of the type of sharing implies operator actions on both systems. Whatever the transition state, from static to dynamic or conversely from dynamic to static, the device must be removed (MDHW OUT) from both systems before being inserted:

- on one system (static sharing) by the MDHW IN command issued on one of the two systems,
- on both systems (dynamic sharing) by the MDHW IN SHR command issued on both systems.

If this method is not followed, it may result in a system deadlock.



For more details, refer to the complete explanation of MDHW and DHW in Chapters 7 and 8 *Description of the Operator Commands*.

In some circumstances the action requested by a MDHW command cannot be immediately satisfied. The main cases are :

- an OUT request is issued for a device which supports a volume presently used by a running job,
- an OUT request is issued for a controller and this controller is implied in the only access path to a device which supports a volume presently used by a running job,
- an IN request is issued for a device which is seized by another system.

The processing of the request will resume as soon as the pre-emption condition disappears.

Every Reconfiguration command displays one or several messages. The messages issued in response to a MDHW command give the status of the requested action: successful or unsuccessful or delayed with the cause of the result if not successful. The *Console Message Directory* gives a detailed explanation of each message.

10.1.3.6 Dynamic Reconfiguration and System Restart

When a device or controller is taken out of the system configuration by an MDHW OUT command, this OUT state is recorded in the SYS.ERLOG Control Records.

When a device or controller is put back in the configuration by an MDHW IN command, the OUT state is reset in the SYS.ERLOG Control Records.

GCOS 7 Restart

At GCOS 7 restart, ERLOG retrieves from the Control Records the previous device/ controller states and restores them. The following rules apply:

1. A device is taken out of the configuration if it was previously out and if the volume mounted on it is not a BKST/SHARED/RESIDENT volume or if it not allocated to a job.
2. ERLOG tries to seize sharable devices not currently shared, if they were not previously out of the configuration or if they were previously out but are now mounted with a RESIDENT volume.



3. A controller is taken out of the configuration if it was previously out of the configuration and if it is not connected to:
 - a device allocated to a job with no other path left;
 - a device supporting a BKST/SHARED/RESIDENT volume with no other path left.

Reconfigured System

At GCOS 7 restart on a reconfigured system, the rules for device/controller restoration are:

1. If the operational configuration matches an old configuration recorded in SYS.ERLOG, the information concerning that configuration is retrieved from the file and used to restore device/controller states.
2. If it is a new configuration or if the SYS.ERLOG file has been automatically re-initialized:
 - all sharable MS devices are taken out of the configuration except those with a BKST/SHARED/RESIDENT volume and those allocated to a job.
 - other devices/controllers are presumed to be IN the configuration and the rule (2) is applied to seizable devices.

SYS.ERLOG Unavailable

If at GCOS 7 restart the SYS.ERLOG file is unavailable, ERLOG does not modify device/controller states, i.e. there is no taking out of the configuration and no seizure.



10.2 Reconfiguration on DPS 7000/XTA models.

10.2.1 Overview

A DPS 7000 /XTA is a Bull platform based on Intel Architecture managed by WINDOWSTM operating system (refer to chapter 2). The Largest models have redundant hardware components .

V7000 is a Bull software component that allows to run GCOS 7 on DPS 7000 /XTA. It is composed of a core part and a System Administration Manager (SAM). The first one is similar to firmware and the SAM corresponds to the Service Processor functions of previous DPS 7000.

No change occurs in GCOS 7 architecture to run on DPS 7000/XTA. The only difference is that some GCOS 7 resources are now virtual because they are mapped on physical resources managed by WINDOWSTM.

As described in 10.1, GCOS 7 resources reconfiguration still takes place automatically when there is a malfunction, and in other circumstances can be performed manually by the operator.

It is possible to remove certain failed resources from the configuration without stopping processing.

The DPS 7000 /XTA configuration envelope depends on the model bought by the customer. The configuration is then defined inside this envelope by Bull support people according to the customer needs.

The resources of this configuration can be acted upon in two circumstances:

1. At system initialization (**Static Configuration**)

A component can be excluded:

- either automatically when the component cannot be initialized
- or upon operator request before initialization time or when a component needs to be reserved for a diagnostic operation at a future time;

2. When the system is running (**Dynamic Reconfiguration**)

- upon component failure, if a component becomes not operational it is automatically isolated. On redundant configurations the system remains running when possible.
- upon operator request, for example:
 - to isolate a component where too many non fatal incidents arise,
 - to insert a component in V7000 configuration.



There is no bisystem operation available.

There is no disk sharing available on DPS 7000/XTA models.

DPS 7000 /XTA Architecture and Resources

The general DPS 7000 architecture is kept for DPS 7000/XTA, so we have the following from the Configuration and Reconfiguration points of view:

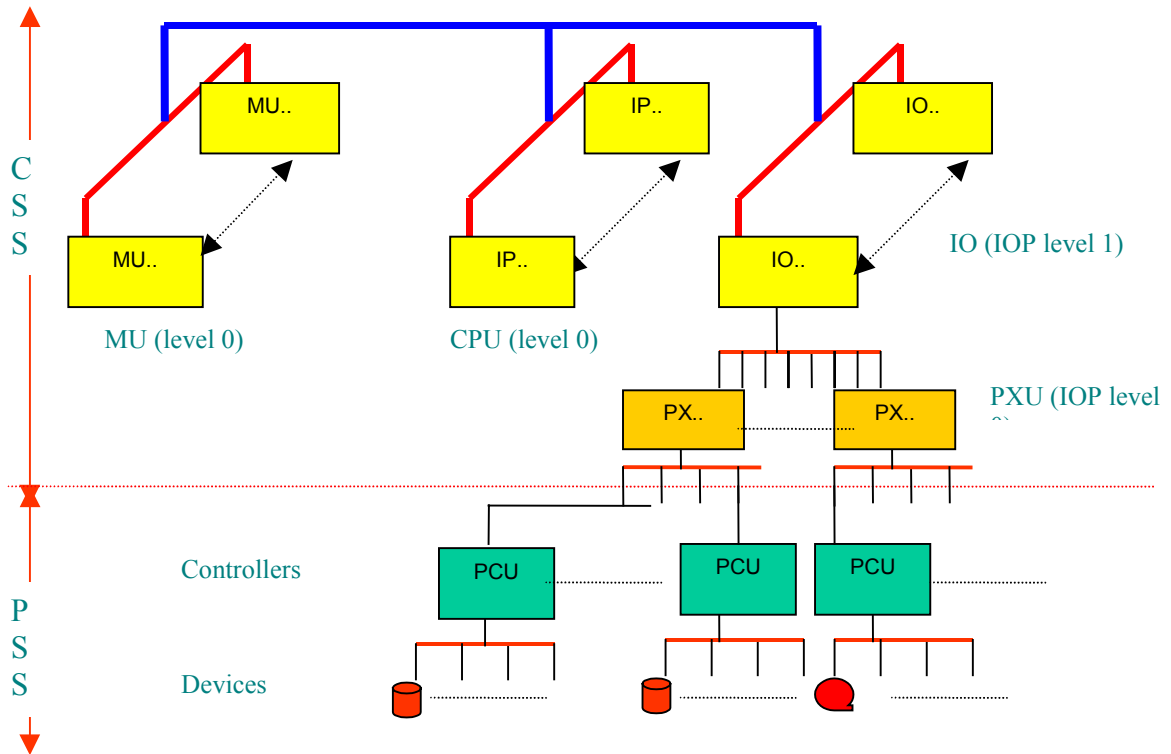


Figure 10-7. DPS7000 /XTA : GCOS7 System Architecture

On DPS 7000/XTA, PXU are typed according to the peripherals that are related to them. There are four PX types :

- Disks
- Cartridges
- Printer
- Telecom

Only one telecom controller can belong to a given IOP.



The system is structured in two major parts which are composed of physical or logical (V7000 software components) resources:

- the Central SubSystem (CSS),
- the Peripheral SubSystems (PSS).

The Peripheral SubSystems are connected to the Central SubSystem through Physical Channels (PC).

Central SubSystem

It is made of the following resource types:

CSS resources	Physical/Logical	Comment
Main Memory Units (MU)	Logical	Windows virtual Memory range reserved to GCOS7.
Central Processors Units (IP)	Physical	Intel processors dedicated to GCOS7
Input-Output Processors (IO and PX)	Logical	V7000 software components dedicated to IO operations

Each IOP is a WINDOWS™ process in charge of managing up to 8 PXU. Each PXU is a WINDOWS™ thread in this process.

CSS IOP resources are organized into several levels. Every action performed at a given level is propagated at the lower levels and to the PSS resources which are connected to it.

PXU resources are specialized upon the device type they controlled.



Peripheral SubSystem

The Peripheral SubSystem includes:

- the controllers: Peripheral Control Units (PCU),
- the peripheral devices.

There exists several types of controllers, each of them managing up to 4 peripheral devices. A controller is referred to by its external identifier which has the following format:

- MCxx for Mass storage Controllers which manage disk devices.
- TCxx for Tape Controllers which manage cartridges.
- UCxx for Unit record Controllers which manage "slow" peripherals: printers or local telecom lines.
- UCx0 for Communication Controllers which allow the connection of Virtual Communication Processors and DSA network. This controller manages up to 6 or 8 pairs of LT.

PSS resources	Physical/Logical	Comment
PCU / Controller	Logical	V7000 software belonging to a PX thread
GCOS 7 Disk devices	Logical	WINDOWS™ NTFS file
GCOS 7 Cartridge devices	Physical	Cartridge device connected to platform
GCOS 7 Printer devices	Physical	Printer device connected to an USB port or serial port
GCOS 7 LT (telecom)	Logical	Managed by VCP7 and OCS

V7000 always uses specific WINDOWS™ APIs to access GCOS 7 devices, even for those that are qualified as “Physical” in the frame above.



10.2.1.1 When and How to Configure or Reconfigure a System

On DPS7000/XTA configuration is first defined and set active by Bull according to the model and the customer needs.

- Static Configuration of the CSS resources takes place at V7000 initialization.. The number of CPU and the memory dedicated to GCOS 7 are checked versus platform physical resources when V7000 starts. So, some GCOS 7 processors (IP) or memory units (MU) may be excluded when V7000 starts :
 - there must be at least the same number of processors left for WINDOWS than the number of processors dedicated to GCOS 7.
 - there must be at least the same size of physical memory left for WINDOWS than the memory size allocated to GCOS 7.
- Static configuration is more simple on XTA models because bisystem feature is not available and only one system can take place.
- The Dynamic Reconfiguration of the CSS resources takes place when the system is running.

When a V7000 IO process or thread aborts a processor check exception is reported to GCOS7, then this resource and all the down linked ones are dynamically set out of the configuration by GCOS 7 if no resident or system device is concerned. Just after, this IOP or PXU has to be set into the configuration through a GCOS 7 operator command.

The Reconfiguration of PSS components is described in 10.1.3.:

- It is always performed when the system is running,
- It is performed through GCOS 7 commands,
- All what is described in section 10.1 applies, except LOS and dual LOS features that are never managed by DPS 7000/XTA modes. The physical channels management is done by WINDOWSTM.
- Dual access to disks and IO load balancing are available or not depending on disks subsystems connected and their associated features (ATF for FDA and Power Path for CDA7). Please refer to section 2 of this manual or related disk subsystem documentation.



A. Operator-System Dialog on a Console

A main operator session may be logged in the SYS.LOGC system file (refer to Chapter 4: Main Operator Console Log).

There are four types of dialog which can occur during a GCOS 7 session:

- Messages from the system to the operator.
- Operator commands to the system.
- Questions by the system requiring an immediate reply.
- Questions by the system which accept a deferred reply.

Messages from the System to the Operator

Informative messages may take one of two forms.

1. Form 1 is preceded by a time stamp, followed by the object and the message itself, as shown in the two examples below:

```
hh.mm dvnrm OUT (BY SYSTEM)
```

(meaning that the specified device is removed from the configuration)

```
hh.mm ron STARTED jnm usernm jclass
```

(meaning that the specified job has started execution)

where:

hh.mm	is the time stamp in hours and minutes
dvnrm	is the device name and number
ron	is the run occurrence number
jnm	is the job name
usernm	is the user name
jclass	is the job class



- Form 2 differs from the first in that it has a 2-character, 2-digit identifier instead of a time stamp, as shown below:

```
DV24 dvnrm UNKNOWN VOLUME
```

(meaning that the system cannot recognize the volume mounted on the device indicated)

```
CG05 ATT SUCCESSFUL
```

(meaning that an ATTACH_CATALOG command has been successful).

where the first two characters of the identifier relates to the message set (e.g., DV = devices, IL = initial load) and the second two digits represent the number of that message in the set.

If operator action is required the message is preceded by an asterisk "*":

```
* hh.mm dvnrm MOUNT volnm FOR ron
```

(meaning the job indicated needs the specified volume to be mounted on the device shown, which is free to be used).

These messages are repeated until the operator takes the necessary action. To indicate a repeated message two asterisks (**) precede the message.

If a message extends to more than one line, the following lines are indented but neither the time stamp nor the identifier are shown. For example,

```
02/hh.mm ron.ssn lm-nm ABORTED [SEV sevn] REPEAT?
      status RC = ccccccc --> siu, return code
```

NOTE:

This first three character positions on the log are reserved exclusively for the action indicator (*), repetition indicator (**) and deferred reply identifier (02/ in the above example).

Operator Commands to the System

GENERAL PROCEDURE (In line mode)

- Press the ATT or BREAK or Insert key on the system console. This causes a flashing underscore character to appear.
- The message S: appears at the start of a new line.
- The flashing underscore is replaced by a flashing cursor.
- The message may now be keyed in.
- When the message is correctly formatted, press the EXECUTE or TRANSMIT key.

If the VDU and the Console Serial Printer are both functioning as the console, the message appearing on the VDU screen is also printed on the serial printer.



Example

S: SOW PR01 (meaning Start Output on Printer 01)
(The underlined part is the command entered by the operator).

ENTERING COMMANDS

When typing in commands, note the following:

- Some commands require several parameters; these parameters are separated by at least one space.
- If a command extends to more than one line, the first line should be terminated by a hyphen. The system outputs -: at the beginning of the next line, and the rest of the command can be typed in.
- Several commands may be typed on the same line.
- On an IOF console, each command must be terminated by a semi-colon(;).

ABNORMAL CASES

There are several abnormal cases to be considered:

- The typed command is incorrect but has not yet been confirmed.
Press the ? (backspace) key repeatedly to delete the incorrect entry, then re-type the command.

- The command does not exist.

The system answers:

```
OP02 cmdnm UNKNOWN COMMAND
```

You should enter a legal command.

- The command syntax is illegal.

The system answers:

```
OP03 cmdnm ILLEGAL SYNTAX
```

You should re-type the command, with the correct syntax.

- The system queues are full or the processor is not active. The command is not taken into account and the system answers one of the following:

```
OP07 cmdnm RETYPE THE COMMAND LATER
```

```
OP08 cmdnm COMMAND NOT CURRENTLY AVAILABLE
```



Questions Requiring an Immediate Reply

This type of question is displayed when the system needs instructions and cannot continue until an answer has been provided.

The syntax of the question is an asterisk, operation code, colon, question subject, and a question mark.

On the line following, the system issues the prompt, I:. You must now type in the answer.

For example:

```
* SV05 GCOS: FILE SALVAGING ?  
I: NO  
(the underlined part is the operator's reply).
```

The question is repeated if the answer is not given within a certain time. No other message is given after the system has displayed I:, until the operator has typed in a reply.

Questions Which May be Given a Deferred Reply

This type of question may be answered at the operator's convenience by using the REPLY command. A reply identifier is given with the question which must be used in the operator's reply.



Example

07/OU02 PR02 OUTPUT X18:4 RESTART FROM ?

.
. .
. .
. .
. .

S: REP 07 BEG

If the answer is incorrect the question is re-issued but with another reply identifier.

ABNORMAL CASES

There are two abnormal cases:

- the reply identifier does not correspond to the one given in the question,
- the job which asked the question has terminated.

In both cases the system answers:

OP01 INVALID REPLY IDENTIFIER OR ALREADY ANSWERED

In the former case, the operator should retype with the correct reply identifier.





B. Calling System Utilities

B.1 DUMPACT

```
EJR DUMPACT LIB = SYS.HSLLIB VALUES = (efn, dvc, volnm, filestat,  
options, lmlib)
```

where:

efn	: user file name	Default = none
dvc	: device class name	Default = MT/T9
volnm	: volume name	Default = WORK
filestat	: file system option	Default = UNCAT
options	: [{ AP OU }]	Default = AP (Append mode)
	[{ OLD V5 }]	Default = current record format
	[DB]	Default = Normal mode (No debug option)
lmlib	: device class name and volnm of LM library	Default = SYS.HLMLIB

**Description of Options:**

AP	User file opened in append mode
OU	User file opened in output mode
OLD	Record format of the V5 Release (the keyword OLD ensures compatibility)
V5	Record format of the V5 Release
DB	Debug option.

NOTE:

Options are separated by a space, and enclosed by quote marks (e.g. 'OU V5 ').

```
-----  
EJR DUMPACT LIB = SYS.HSLLIB VALUES = ([,OPTION = [{ }] [{ }] [DB]);  
                                           [{OU}] [{V5}]  
-----
```

When the DB parameter is used, the Accounting Virtual Memory File is not erased.



B.2 PRLOG

```
EJR PRLOG LIB = SYS.HSLLIB VALUES = ('prlog-options')
```

Prints the contents of the SYS.ERLOG file.
(The 'prlog-options' are decided by the Service Center)

B.3 PRLOGC

```
EJR PRLOGC SYS.HLMLIB [,OPTIONS = 'options']
```

where 'options' is:

```
{ REPORT = { ALL | hex8 } }
{
{ REPORT=ACTIVE [ {LASTP} ] [ ] [ ] }
{ REPORT=ACTIVE [ ,FROM={BEG } ] [ ,TO= hh.mm } [ ,LINES=nnn ] [ ,SUMMARY=/string/] }
{ [ {hh,mm} ] [ ] [ ] }
```

Where:

- | | |
|--------|--|
| ALL | Print the entire contents of the file SYS.LOGC. |
| hex8 | Print console file concerning the specified CPU. The CPU concerned is identified by a string of eight hexadecimal characters. The characters are found in the footer of the SYSOUT banner. |
| ACTIVE | Print console lines concerning the CPU where H_PRLOGC is running. |

The following parameters may be used with the ACTIVE option:

- | | |
|--------------|--|
| FROM = LASTP | Indicates that printing has to be resumed where it was left off. If this has already been done, printing begins from the oldest message. (Default option) |
| FROM = BEG | Indicates that printing must start at the oldest available message. |
| | NOTE: The level LASTP is only kept up to date for these two options if they are used on their own. |
| FROM = hh.mm | Indicates that the listing must begin from the indicated hour, within the limits of the last 24 h. If it is unsuccessful, printing starts from the oldest message. |



TO = hh.mm	Printing stops at the dialog logged at the given time, hh.mm. If no dialog is found for that time, printing stops at the last dialog logged.
LINES	Gives the maximum number of lines to be printed.
SUMMARY	Only prints the lines containing the specified string. The string cannot contain a slash and cannot exceed 12 characters.

NOTES:

1. The use of the last three options prevents the level LASTP from being updated.
2. PRLOGC can also be invoked using the following Load Module in SYS.HLMLIB:

```
STEP H_PRLOGC,SYS.HLMLIB OPTIONS = 'REPORT = ACTIVE, FROM = '&1'';
```

B.4 TD (Tests and Diagnostics)

```
EJR TD LIB = SYS.HSLLIB VALUES = (parameters)
```

Refer to the *Operator Reference Manual* or the *Operator's Guide* for your system, and to the *OLTD Operator Guide* for a description of these parameters.



B.5 DP_SAVE (Saving GCOS 7 Anomaly Files)

B.5.1 Introduction

The main purpose of EXECUTE_DP_SAVE is to generate Software Technical Action Request (STAR) elements. These are automatically saved on tape when an anomaly has occurred on the system.

The elements saved depend on the anomaly type, which can be:

- a system crash
- a NETwork processor (NET) anomaly
- an OPEN 7 crash
- a job (except telecom or OPEN 7) abort that creates a binary dump in the SYS.SP_DUMP file
- any other anomaly without a dump.

System Crash

The following assumes: a crash occurred, the system is now running under normal conditions, and that a dump was saved in the efn (default SYS.SYSDUMP) file by the ISL dump commands:

```
-----  
* IL03 DUMP ? 0, [efn:] media,DEF  
* IL03 DUMP ? EXEC,DPS_STD  
* IL03 DUMP ? END  
-----
```

See also Section 3.4.3.5 *DUMP* command.

NET Anomaly

The following assumes that an anomaly occurred on one or more NETwork servers (Datanet, CNP7), or Local Controllers (TNS, OCS), and that one or several dumps (depending on NET type) are written to the SYS.DSADUMP library by either of these commands:

```
S: STSVR TYPE=FECM RSYS=<FEP_name> ACTION=DUMP  
(If the Network Server is a Datanet or CNP7)
```

```
S: STSVR TYPE=OCS NAME=<OCS_server_name> ACTION=DUMP  
(If the Local Controller is an OCS)
```



OPEN 7 Crash

Note that DP_SAVE is only supported from OPEN 7 V2.0.4.

When an OPEN 7 crash occurs, the following message is sent to the OPEN 7 submitter:

```
From Submitter-name <Xnnnn>: *** CRASH OF THE OPEN7 <OPEN7-name>
```

The RON "Xnnnn" is the RON of the job "OPEN7". The "OPEN7-name" corresponds to the OPEN7 directory name.

OPEN 7 writes the JORs of the OPEN7, OPEN7P, and OPEN7R jobs in the OPEN7 DUMP library.

Several OPEN7P and OPEN7R jobs are associated with one job "OPEN7".

Job Abort

After a job abort (except telecom or OPEN 7) that creates a binary dump in the SYS.SP_DUMP file, the following message is sent to the operator console and the JOR:

```
DP03 ron_dsn_dpindex PGID=jjjj lm-name DUMP CREATED
```

where:

- ron is the Run Occurrence Number of the job.
- dsn is the Dynamic Step Number. It starts at 1 and increments at each executed step. (This enables two executions of the same step to be distinguishable).
- dpindex is the Dump Index. It starts at 1 at the beginning of the step, and increments each time a dump is taken.

Example

```
DP03 X1493_4_1 PGID=16 H_LIBMAINT DUMP CREATED
```

This refers to the first dump of the fourth step of job X1493.

An Anomaly with no Dump

This results either in system traces or a write to the system error log file (SYS.ERLOG).



B.5.2 Processing

Use the primary screen of the EXECUTE_DP_SAVE directive to specify the type of anomaly, plus information about the system and output media. A secondary screen displays specific parameters for each anomaly.

After all of the required parameters have been specified, JOB_DP_SAVE is launched to save the elements on the output media. At each main step of the job execution, messages are sent.

The processing is normally terminated when JOB_DP_SAVE is completed. If it aborts, a message is sent to the submitter. The output of the job must be scanned and after correction (if any) the command EXECUTE_DP_SAVE can be re-launched.

B.5.3 Syntax of the EXECUTE_DP_SAVE Directive

```
-----  
{ EXECUTE_DP_SAVE }  
{ DP_SAVE         }  
{ DPSV           }  
-----
```

B.5.3.1 Parameters Displayed on the Primary Screen

```
-----  
[ { SYSTEM_CRASH }  
[ {               } = bool ]  
[ { CRASH        }  
  
[ { NET_ANOMALY  }  
[ { NET          }  
[ {             } = { TNS | FEPS | OCS | FECM | NO | 0 | 1 } ]  
[ { FEP_ANOMALY }  
[ { FEP          }  
  
[ { OPEN7_CRASH }  
[ {             } = { 0 | bool } ]  
[ { OPEN7       }  
  
[ { OTHER_PROBLEM }  
[ {             } = { 0 | bool } ]  
[ { OTHER        }  
-----
```



```

[ {USER_INFORMATION}
[ {                } = {0 | bool}]
[ {USER            }

[ {SITE            } = {#WSYS | name4}]

[ {REPORT          } = {YES | NO}]

[ {OUTPUT_MD       }
[ {OUTMD           } = (name6 [,name6 [,name6 [,...]])]
[ {OUMD            }

[ {OUTPUT_DVC      }
[ {OUTDVC          } = {MT/T9/D6250 | char20}]
[ {OUDVC           }

[ {SYSCATMD        }
[ {                } = name6]
[ {SYCATMD         }

[ {SYSCATDVC       }
[ {                } = char20]
[ {SYCATDVC        }
-----
[ KEEP              = {0 | bool }]
```

B.5.3.2 Parameters Displayed on a Secondary Screen

More parameters are displayed on a secondary screen. Which parameters are displayed depends on the type of anomaly.

SYSTEM_CRASH = 1

```

[ INEFN           = { SYSSYSDUMP | file44 } ]
[ INMD            = name6                ]
[ INDVC           = char20                ]
[ PRINT           = { NO|MM}              ]
[ DPSZ            = { 40 | dec4 }         ]
[ WKMD            = name6                ]
[ WKDVC           = char20                ]
```



NET_ANOMALY = TNS

```
{ LCT_NAME }
{           } = ( EAnn , [ EAnn ] )
{ LCT      }

[ RON = Xnnnn ]
```

NET_ANOMALY = FEPS

```
NODE = ( name4, [ name4 ], [ name4 ], [ name4 ] )

[ RON = Xnnnn ]
```

NET_ANOMALY = FECM

```
{ LCT_NAME }
[ {         } = EAnn ]
{ LCT      }

{ RSYS_NAME }
[ {         } = name4 ]
{ RSYS      }
```

NET_ANOMALY = OCS

```
OCS = ( EAnn, [ EAnn ], [ EAnn ], [ EAnn ] )

RON = Xnnnn
```

NET_ANOMALY = 1 (previous visibility TS 7458)

```
[ NODE      = name4           ]
[ OCS       = name4           ]
[ RON       = Xnnnn           ]
```

OPEN7_CRASH = 1

```
{ OPEN7_DIRECTORY }
{                 } = { OPEN7 | char20 }
{ OPEN7DIR        }
[ RON             = Xnnnn           ]
```




OTHER_PROBLEM = 1

```
[ RON          = Xnnnn          ]
[ DUMP_ID     = char13          ]
[ TRACE_ONLY  = { 0 | bool }    ]
```

B.5.3.3 Parameters Displayed on an additional Secondary Screen

In the case of USER_INFORMATION = 1, more parameters are displayed in an additional secondary screen.

```
[ UEFN        = file44          ]
[ ULIB        = lib44           ]
[ ULIBTYPE    = { SL | LM | BIN | SM | CU } ]
[ USFN        = { * | char31 }  ]
[ UMD         = name6           ]
[ UDVC        = char20         ]
```

B.5.4 Description of Parameters

SYSTEM_CRASH	Indicates whether system crash anomaly or not. A dump must have been saved in the efn file (see INEFN parameter) at the ISL step. If no type of anomaly is selected by the submitter (e.g. S: EXECUTE_DP_SAVE;), GCL automatically selects SYSTEM_CRASH. No default value.
NET_ANOMALY	Indicates whether the incident occurred on Network frontals (DATANET, CNP7, or OCS). The old visibility TS 7458 is always accessible when NET=1, but it is more suitable to select, depending on the context of the anomaly, one of the new values: TNS, FEPS, FECM or OCS. In addition, these new options allow to save several dumps, depending on the type of frontal impacted. Default value: NO (equivalent value: 0).
OPEN7_CRASH	Indicates whether the incident is an OPEN 7 crash. Default value: 0.
OTHER_PROBLEM	Must be used when the anomaly is not a system crash, FEP anomaly or an OPEN 7 crash, and when a binary dump or only system traces are produced. Default value: 0.



USER_INFORMATION	Must be used when a private file or subfile has to be saved by DP_SAVE with the previous elements. Default value: 0.
SITE	Gives the name of the site where the crash occurred (up to four characters). The name(s) of the tape(s) used by DP_SAVE are assigned at runtime, using the site-name as a prefix. Default value: #WSYS.
REPORT	If REPORT=YES the job execution listing is output. If an error is detected during DP_SAVE, the job execution listing is provided whether REPORT=YES or REPORT=NO. Default value: YES.
OUTPUT_MD	Name of the output media (up to ten value). By default, if this parameter is not given, names are generated by the GCL EXECUTE_DP_SAVE with the parameter <SITE> and the DATE and TIME.
OUTPUT_DVC	Device class of the output media: tape (MT) or cartridge (CT). Default value: MT/T9/D6250.
SYSCATMD	Media name of the volume where SYS.CATALOG was located at crash time.
SYSCATDVC	Device type of the SYSCATMD volume.
INEFN	External file-name of the input file containing the dump. Default value: SYS.SYSDUMP.
INMD	Name of the media containing the input-file in the case of a non-cataloged file.
INDVC	Device-type of the media containing the input-file in the case of a non-cataloged file.
PRINT	PRINT=MM prints a summary. PRINT=NO means no print output. Default value: NO.
DPSZ	Specifies the size (in cylinders) of a temporary file, the image of the SYS.SYSDUMP. The file is created by DP_SAVE and the summary report (see PRINT parameter) is written from it. Default value: 40.



WKMD	When PRINT=MM is selected, the INEFN file must first be transformed into a DPAN (work) file. WKMD indicates the media where this work file is located. If no value is given, the work file is allocated to a resident disk. No default value.
WKDVC	Device type of the media containing the temporary work file for printing. No default value.
RON	Run Occurrence Number of the job that aborted. In NET or OTHER_PROBLEM context, this option saves all the binary dumps produced by the job each time it aborts. In OPEN 7 crash context, it is the ron of the job OPEN7. No default value.
KEEP	If set to 1, indicates in the OPEN7_CRASH context or in the OTHER_PROBLEM context that the JORs and LOG of OPEN7 or the binary dump have to be kept in the library at the end of the DP_SAVE execution. Default value: 0.
DUMP_ID	Indicates the number of the binary dump produced by the abort of a job in OTHER_PROBLEM context. This parameter is required only when the job created several identical binary dumps. Default value: <%RON>_*
TRACE_ONLY	TRACE_ONLY=1 indicates that only system trace files have to be saved without any DUMP in OTHER_PROBLEM context. Default value: 0.
OPEN7_DIRECTORY	Name of the OPEN 7 directory. Default value: OPEN7.
NODE	When NET=1, indicates the name of a Datanet or CNP7 where the incident occurred (old TS 7458 visibility). When NET=FEPS, indicates a list of up to 4 Datanet or CNP7 names where incidents occur.



LCT_NAME	When NET=TNS, indicates a list of up to 2 local controller names (LCT) where incidents occur. When NET=FECM, indicates a local controller (LCT manageable by the local system).
OCS	When NET=1, indicates the name of an OCS controller where the incident occurs (old TS 7458 visibility). When NET=OCS, indicates a list of up to 4 OCS controller names where incidents occur.
UEFN	External file name of a user-defined file. No default value.
ULIB	Name of a user library containing the subfile(s) to save. No default value.
ULIBTYPE	User library type (see ULIB parameter). No default value.
USFN	User subfile(s) name to save (see ULIB parameter). Star convention is allowed. Default value: * (all subfiles).
UMD	Indicates the medium containing the user file or library in case of uncataloged objects. No default value.
UDVC	Device type of the media containing the user file or library, in case of uncataloged objects. No default value.

The <user> parameters should be used to save a file or subfile(s) which may help to solve the problem, for example, a user program source subfile and / or generated listings, or a user application trace file (e.g. a VCAM user trace file).



IMPORTANT:

1. A file or subfile saved using the UEFN or USFN parameters must be identified in the STAR problem description, otherwise its meaning is unknown when it is restored from tape.
2. If a mismatch occurs during the selection of options in a DP_SAVE session, exit the EXECUTE_DP_SAVE directive, restart it, then enter the values again.
3. When KEEP=1 is specified, it is the system administrator's responsibility to delete the elements, and avoid library overflows.



Constraints

Only one of the parameters SYSTEM_CRASH, NET_ANOMALY, OPEN7_CRASH, or OTHER_PROBLEM may be used.

In an OTHER_PROBLEM context, when TRACE_ONLY = 1, RON and DUMP_ID must not be specified.

In the NET_ANOMALY context:

- with the old visibility (TS 7458 compatible), either NODE or OCS must be specified and they are mutually exclusive.
- when NET=FECM, either LCT_NAME or RSYS_NAME must be specified and they are mutually exclusive.
- when NET=OCS, the RON parameter is mandatory.

B.5.5 Elements Saved

Table B-1 describes the parameters to use according to the anomaly context of SYSTEM_CRASH, OPEN7_CRASH or OTHER_PROBLEM and the elements saved for each context.

Table B-1. SYSTEM_CRASH, OPEN7 and OTHER_PROBLEM Anomaly Parameters and Elements Saved by DP_SAVE

Anomaly and Context Parameter	Specific Parameter(s) to Use	Saved Files
System crash [SYSTEM_CRASH=1]	None (default) [INEFN=<file name>]	System information files. System dump.
OPEN 7 crash OPEN7_CRASH = 1	OPEN7_DIRECTORY= <dir name> RON = Xnnnn	JOR of OPEN7, OPEN7P and OPEN7R jobs in OPEN7 DUMP file. LOG subfiles in OPEN 7 DUMP file. System information files.
Other anomaly with binary dump OTHER_PROBLEM=1	RON = Xnnnn or DUMP_ID = Xnnnn_i_j	Binary dump(s). System information files.
Other anomaly without dump. OTHER_PROBLEM=1	TRACE_ONLY=1	System information files.



Table B-2 describes the parameters to use according to the NET_ANOMALY contexts, and the elements saved for each context.

Table B-2. NET_ANOMALY Parameters and Element Saved by DP_SAVE (1/2)

Anomaly and Context Parameters	Specific Parameter(s) to Use	Saved Files
FEPS (Datnet, CNP7) NET_ANOMALY = FEPS	NODE = (<server-name>, <server-name>, <server-name>, <server-name>) with in option: RON=Xnnnn	FEPS dump (s) FEPS history files FEPS patches DSA logfiles FEPS trace files System information files save SYS.SPDUMP binary dump.
TNS (LCT) NET_ANOMALY = TNS	LCT_NAME = (<lct-name>, <lct-name>) with in option: RON=Xnnnn	TNS dump (s). TNS history files. TNS patches. DSA logfiles. TNS trace files System information files. save SYS.SPDUMP binary dump.
OCS (LCT) NET_ANOMALY = OCS	OCS = (<lct-name>, <lct-name>, <lct-name>, <lct-name>) RON=Xnnnn	FEP dump. FDDI dump. TNS patches OCS dump. FEP configuration file. DSA logfiles. OCS controller trace file OCS server trace files. System information files dump(s). save JOR and SYS.SPDUMP binary dump.
FECM (LCT) NET_ANOMALY = FECM	LCT_NAME = <lct-name>	LCT dump (s). LCT history files. LCT patches. DSA logfiles. System information files.
FECM (Remote System) NET_ANOMALY = FECM	RSYS_NAME = <remote-system-name>	RSYS dump (s). RSYS history files. RSYS patches. DSA logfiles. System information files.



Table B-2. NET_ANOMALY Parameters and Element Saved by DP_SAVE (2/2)

Anomaly and Context Parameters	Specific Parameter(s) to Use	Saved Files
(Datnet, CNP7, Local Controller OCS, Local Controller TNS) NET_ANOMALY = 1 (old visibility TS 7458)	NODE = <server-name> or NODE = <lct-name> or OCS = <lct-name> with in option: RON=Xnnnn	idem FEPS case or idem TNS case or idem OCS case save SYS.SP_DUMP binary dump.

The system information files saved by DP_SAVE are:

- The list of all system patches and the contents of site patches.
- Information about system hardware configuration as well as system firmware and software status.
- Messages sent to/from the main console (copy of the SYS.LOGC file).
- Complete report on hardware and system errors (copy of the SYS.ERLOG file).
- Events recorded by the standard system trace (copy of the SYS.SWLOG file).
- Optional system trace backup files (the SYS.GSF.SWLOG_0 and SYS.GSF.SWLOG_1 files).



B.5.6 Examples

B.5.6.1 Examples of Commands

All the system trace files are always saved.

System Crash Context

```
EXECUTE_DP_SAVE;      {save the file SYS.SYSDUMP on tape with the name
                       generated by DP_SAVE}

DP_SAVE SITE=BY12 OUTPUT_DVC=CT;
                       {save the file SYS.SYSDUMP on site BY12 on
                       cartridge with the name generated by DP_SAVE}
```

OPEN 7 Crash Context

```
DP_SAVE SITE=BY12 OPEN7_CRASH=1 OPEN7_DIRECTORY=OP7 RON=X1234;
                       {save on tape the JOR and log of the OPEN 7 "OP7",
                       RON X1234}

DP_SAVE OPEN7 RON=X1234;
                       {save on tape the JOR and log of the OPEN 7
                       "OPEN7", RON X1234}
```

NETwork Anomaly Context

```
DP_SAVE SITE=BY12 NET=OCS OCS=(EA51, EA69, EA91), RON=X457;
                       {save elements for the local OCS controllers EA51,
                       EA69 and EA91; save binaries dumps X457_*}

DP_SAVE SITE=BY12 NET=FEPS NODE=(BY21, BO45, BL74, BY32);
                       {save elements for the datanets BY21, BO45, BL74
                       and BY32}

DP_SAVE SITE=BY12 NET=FEPS NODE=(BY21) RON=X56;
                       {save elements for the datanets BY21; save binaries
                       dumps X56_*}

DP_SAVE SITE=BY12 NET=TNS LCT=(EA51, EA69) RON=X715;
                       {save elements for the local TNS controllers EA51 and
                       EA69 and binaries dumps X715_*}
```




```
DP_SAVE SITE=BY12 NET=FECD RSYS=(BY21) ;  
        {save elements for the remote system BY21}
```

```
DP_SAVE SITE=BY12 NET=1 NODE=(BY21) ;  
        {save elements for the datanets BY21, using the old  
        TS7458 visibility}
```

OTHER_PROBLEM Contexts

```
DP_SAVE SITE=BY12 OTHER_PROBLEM=1 RON=X354 ;  
        {save all binary dumps for RON X354}
```

All Contexts

```
DP_SAVE SITE=BY12 OTHER RON=X1234 USER_INFORMATION=1 UEFN=MY.FILE ;  
        {save the elements linked to the incident plus the user  
        file MY.FILE}
```

```
DP_SAVE CRASH INEFN=SYS.SYSDUMP1 USER_INFORMATION=1 ULIB=MY.LIB  
ULIBTYPE=SL USFN=MY_SUBFILE_* ;  
        {save the elements linked to the crash plus the user  
        subfiles MY_SUBFILE_* in the source library  
        MY.LIB}
```

```
DP_SAVE OTHER DUMP_ID=X_1234_1_2 SYSCATMD=DISK1 SYSCATDVC=MS/FSA ;  
        {save the elements of the incident that occurred on the  
        non-running system set on the volume  
        DISK1:MS/FSA}
```



B.5.6.2 Examples of Contexts

Example 1: on a site called HCSI

1. A system crash occurs.
2. At ISL time the operator chooses the DUMP option.
At the prompt * ILO3 DUMP? he answers:

```
ILO3 DUMP? O, SYS.SYSDUMP, DEF
ILO3 DUMP? EXEC, DPS_STD
```
3. Following GCOS 7 READY, he launches DP_SAVE with the directive:

```
EXECUTE_DP_SAVE;
```
4. DP_SAVE gives a list of 10 tapes to be used, the names of which are prefixed by the site name, e.g.:

```
HCSI1A, HCSI1B, HCSI1C, HCSI1D, HCSI1E, HCSI1F, HCSI20,
HCSI21, HCSI22, HCSI23
```

NOTE:

DP_SAVE can prepare the name with the OUTPUT_MEDIA = <media-name> option, but the actual names of the tapes are assigned at this moment. Therefore it is not possible to prepare the tapes beforehand (VOLPREP). This avoids tape name conflicts at Bull Service Centers.

5. The operator prepares the tape(s). To do this he may use a site-specific command (to be created before), e.g.:

```
PREPARE_TAPE VOL=<oldmd>:MT/T9 NEWVOL=HCSI1A:MT/T9/D6250)
```
6. He waits for the end of the JOB DP_SAVE.
7. Then the operator sends, to the Bull Service Center:
 - the DP_SAVE listings: the JOR, the summary printing (if value PRINT=MM is chosen),
 - the tapes.

**Example 2: on the site called HCSI**

1. The FEPS server BYAB aborts and the following messages are sent to the operator console:

```
DP05 X2121_1_1 PGID=10 H FEPS DUMP CREATED
TF05 START LOG BY SERVER SVR_BYAB ON ABORT
EFN: TRC_FEPS_SVR_BYAB_X2121
TF06 LOG TERMINATED BY SERVER SVR_BYAB ON ITS TERMINATION
EFN: TRC_FEPS_SVR_BYAB_X2121
CA21 SVR_BYAB COMPLETED
16.54 X2121.1 ABORTED FEPS QTYC S SEV3
RC=208504C5->EXCEP 5,PROCEXCP
```

2. The operator launches DP_SAVE:

```
EXECUTE_DP_SAVE NET_ANOMALY=FEPS SITE=HCSI, NODE=BYAB, RON=X2121;
```

3. He follows the steps 4 to 7 of Example 1



B.6 DP_PRINT (Printing a Dump on a Line Printer)

Purpose

The JCL command DP_PRINT is used to print information about a crash or a DPAN (Dump and trace analyzer) readable file.

Syntax

```
EJR DP_PRINT LIB = SYS.HSLLIB VALUES =

( [ INEFN = { SYS.SYSDUMP } ]
  [           { efn } ]

  [ INMD = { NIL } ] [ INDVC = { NIL } ]
  [           { media-name } ] [           { dvtyp } ]

  [ WKMD = { NIL } ] [ WKDVC = { NIL } ]
  [           { work-media } ] [           { work-device } ]

  PRINT = { MM } [           { NO } ] [           { NO } ]
           { FW } [ DPAN = {           } ] [ REPORT = {           } ] )
           { CRASH } [           { YES } ] [           { YES } ]
```

Parameters

INEFN	External file name of the file to be printed by DPAN. Default value: SYS.SYSDUMP.
INMD	Name of the media that contains the file to be printed. Default value: NIL (none).
INDVC	Device type of this media. Default value: NIL.
WKMD	When the file to be printed is SYS.SYSDUMP it must be first transformed into a DPAN type file (work file). This parameter indicates the media where this work file is located.
WKDVC	Device type of the work file media. Mandatory when WKMD is used. Default value: NIL.



PRINT	Type of printing to be performed. PRINT=MM is the quickest global printing command. It gives a summary of all processes comprising: <ul style="list-style-type: none">– The SYSTEM BASE– The STATUS of all processes– The RETURN CODES of the processes– The PCB and STACK FRAMES of the processors. PRINT=FW gives a summary of all processes comprising: <ul style="list-style-type: none">– The TYPE 0 writable segments– VMM information. PRINT=CRASH prints an automatic summary concerning the crash reason. The summary given depends on the type of crash analyzed by DPAN.
DPAN	Indicates if the input file is of DPAN type or not. Default value: NO (for SYS.SYSDUMP file).
REPORT	Is used to obtain the reporting of the job. Default value: NO.

Example

1. A crash occurs.
2. The dump is saved on the SYS.SYSDUMP file at ISL time.
3. To print the dump, enter:

```
EJR DP_PRINT LIB=SYS.HSLLIB VALUES= (INMD=UTOPIE
INDVC=MS/B10 PRINT=CRASH)
```
4. Then send the listing to the Service Center or use it for a first analysis of the problem.

NOTE:

The patches are not printed.

Resources required:

Note that DP_SAVE with PRINT=MM and DP_PRINT use at least 50 cylinders (work file on WKMD).



B.7 SVLOGF (Saving TDS/UFAS/GAC Logfile)

SVLOGF performs a continual saving of events produced by the TDS subsystem (including UFAS and GAC).

The GCL command SVLOGF may be activated after assigning "MWINLIB BIN SYS.GSF.BINLIB".

```
SVLOGF [SVLALC=0|1] [SVLMD=media_name]
        [SVLDVC=devclass]
```

Description of Parameters

SVLALC	If 0, there is no pre-allocation of required sequential files for the storage of events (default); if 1, pre-allocation is required, two cataloged sequential files are pre-allocated: SYS.GSF.SVLOGF_0 and SYS.GSF.SVLOGF_1.
SVLMD	This is the media name of the volume where the pre-allocation is to be done (valid only if SVLALC=1).
SVLDVC	This is the device class of the volume for the pre-allocation.

NOTE:

The SVLOGF command without parameters assumes the two sequential files to already be present and dynamically starts a batch process for the saving of events.



B.8 PRINTJOR (Print the JOR of a Job in Execution)

Purpose

The JCL command PRINTJOR is used to print a JOR of a JOB in execution.

The user must specify the RON of the JOB and if necessary the PRTFILE where the information is to be stored, default is SYSOUT.

Syntax

```
EJR PRINTJOR LIB=SYSHSLLIB VALUES=(ron prtfile)
```

Where:

ron	RON number, mandatory .
prtfile	PRTFILE, default is SYSOUT.

Description of Options

RON	RON number of the job whose JOR is to be printed. The RON number can be specified as positional or keyword value of the form Xnnnn. Mandatory .
PRTFILE	The report file. When omitted, the report is printed in the SYSOUT (default value). The user can specify another PRTFILE, which must be in the form: - for cataloged library: LIB=mylib MB=mysubfile LIB=mylib SUBFILE=mysubfile - for uncataloged library: LIB=mylib MB=mysubfile MD=md DVC=dvc LIB=mylib SUBFILE=mysubfile MD=md DVC=dvc MB or SUBFILE can be used as keyword for specifying subfile. See examples of use below.



CAUTION:

Due to the JOR mechanism which is based on a VMF file, only the VMF blocks written in backing store can be printed. The current block which contains the latest records is still in memory and cannot be printed. For a job with outputs, the JOR remains accessible by PRINTJOR until the message OUTPUT COMPLETED is issued. This is true even if the JOR has been cancelled.

Constraints

The RON parameter is mandatory.

If SUBFILE, or MD, or DVC, is specified, LIB must be specified.

Only a MAIN operator or SYSADMIN user can access all JORs. Other users can access only the jobs they have submitted or of which they are users.

Error Messages

The following error messages are sent to the submitter console and in the JOR.

In case of JOR or PRTFILE access failed, the erroneous return code is printed in the JOR after the error message.

PRINTJOR : ERRONEOUS OR UNKNOWN RON: ron

PRINTJOR : PRTFILE ACCESS ERROR: OPEN FAILED

PRINTJOR : PRTFILE ACCESS ERROR: WRITE FAILED

PRINTJOR : JOR ACCESS ERROR: OPEN FAILED

PRINTJOR : JOR ACCESS ERROR: READ FAILED

PRINTJOR : ILLEGAL ACCESS TO JOR



Examples

```
EJR PRINTJOR LIB=SYS.HSLLIB VL=X123;  
                                {result in the SYSOUT.}
```

```
EJR PRINTJOR LIB=SYS.HSLLIB VL=(RON=X123);  
                                {result in the SYSOUT.}
```

```
EJR PRINTJOR LIB=SYS.HSLLIB VL=(X123 LIB=MYLIB SUBFILE=MYSUBFILE);
```

or

```
EJR PRINTJOR LIB=SYS.HSLLIB VL=(X123 LIB=MYLIB MB=MYSUBFILE);  
                                {result in the subfile MYSUBFILE of the cataloged  
                                library MYLIB.}
```

```
EJR PRINTJOR LIB=SYS.HSLLIB VL=(X123 LIB=MYLIB SUBFILE=MYSUBFILE  
MD=BFU033 DVC=MS/B10);
```

or

```
EJR PRINTJOR LIB=SYS.HSLLIB VL=(X123 LIB=MYLIB MB=MYSUBFILE  
MD=BFU033 DVC=MS/B10);  
                                {result in the subfile MYSUBFILE of the uncataloged  
                                library MYLIB.}
```

B.9 V7000 Information Collector

DPS 7000/XTA series include a new maintenance tool to gather all information located on the Windows™ side.

This tool is only accessible by Bull services. It is a V7000 utility and it uses a specific graphical interface to enter commands.

The operator can select the type of problem he wants to document, and the V7000 information collector builds a compacted file on a 4 mm cassette.

This media has to be sent to Bull center for analysis.



C. Index of Commands by "Object"

CATALOG	ATTACH (ATT)
CLC (Cartridge Tape Library Client)	DISPLAY (DCLC) EXECUTE (EXCLS) MODIFY (MDCLC) START (SCLC) TERMINATE (TCLC)
CLX (Cartridge Tape Library Unix)	GET (GTCLXF) EXECUTE (VOLRPT) DISPLAY (DCLX, DSSI) MODIFY (MDCLX) START (SCLX, SSSI) TERMINATE (TCLX, TSSI)
CONFIGURATION	DISPLAY (DC) MODIFY (MDC)
COMPLEX MGMT	DETACH (DTSRV) DISPLAY (DCX) (DMB) (DSRV) START (SCMSR) (SMB) (SSRV) TAKEOVER (TKMB) TERMINATE (TMB) (TSRV)
CONSOLE LOG	DISPLAY (DCLOG) START (SCLOG) TERMINATE (TCLOG)
DEVICE	START (SDV) TEST (TSTDV)



DEVICE TRACE	DISPLAY (DDVTR) MODIFY (MDDVTR) START (SDVTR) TERMINATE (TDVTR)
DIMENSION	CONNECT (CNDIM) CREATE (CRDIM) DELETE (DLDIM) DISCONNECT (DISDIM) DISPLAY (DDIM) MODIFY (MDDIM)
DIRECTIVE	EXECUTE (EXDIR)
DJP (Distributed Job Processing)	DISPLAY (DDJP) MODIFY (MDDJP)
DPRINT (DPR)	
ERROR LOG	DISPLAY (DERLOG) MODIFY (MDERLOG) SEND (SNDERLOG) START (SERLOG) TERMINATE (TERLOG)
FILE	DISPLAY_ASSIGNED_FILES (DASGF)
FILE TRANSFER REQUEST FILTER	ENTER (EFTR) CREATE (CRFLT) DELETE (DLFLT) LIST (LSFLT) MODIFY (MDFLT)
FILTER SET	CREATE (CRFLTST)
FUNCTION	CONNECT_FUNCTION (CNFUNC) DISCONNECT_FUNCTION (DISFUNC) DISPLAY_FUNCTION (DFUNC)
GCL CACHE	DISPLAY (DGC) HOLD (HGC) RELEASE (RGC)
HARDCOPY	START (SHC) TERMINATE (THC)



HARDWARE CONFIGURATION	DISPLAY (DHW) MODIFY (MDHW)
INPUT READER	MODIFY (MDIR) START (SIR) TERMINATE (TIR)
IO CACHE	DISPLAY (DIOC) START (SIOC) TERMINATE (TIOC)
IOF	DISPLAY (DIOF) MODIFY (MDIOF)
JOB	CANCEL (CJ) DISPLAY (DJ) FORCE (FJ) HOLD (HJ) MODIFY (MDJ) RELEASE (RJ)
JOB REQUEST	ENTER (EJR)
LIBTRANS REQUEST	ENTER (ELTR)
LOAD	CONNECT (CNLD) DISCONNECT (DISLD) DISPLAY (DLD) MODIFY (MDLD) START (SLD) TERMINATE (TLD)
MAIL	
MIRLOG_FILE	DEASSIGN (DASGMIR) UNLOCK (UNLMIRF)
MIRROR_MODE	DISPLAY (DMIR) MODIFY (MDMIR)
MIRROR_VOLUME	UNPAIR (UNPMIR) UNLOCK (UNLMIR)
MIRROR_SHARED	RESYNC (RSCMIR)
MOT (Message Of Today)	DISPLAY (DMOT) MODIFY (MDMOT)
OPERATING MODE	MODIFY (MDOM)



OUTPUT	CANCEL (CO) DISPLAY (DO) FORCE (FO) HOLD (HO) MODIFY (MDO) RELEASE (RO)
OUTPUT WRITER	START (SOW) TERMINATE (TOW)
POWER TIMER	DISPLAY (DPWR) (DPWRM) MODIFY (MDPW) (MDPWRM) (TPW) START (SPWRM) TERMINATE (TPWRM)
REPLY (REP)	
REQUEST	CANCEL (CR) DISPLAY (DR)
RESTART OPTIONS	DISPLAY (DRSO) MODIFY (MDRSO)
RMS	DISPLAY (DRMS) START (SRMS) TERMINATE (TRMS)
SAVE SERVER	DISPLAY (DDYNSVR) START (SDYNSVR) TERMINATE (TDYNSVR)
SEND	
STATION	MODIFY (MDST)
SUBMITTER	DISPLAY (DS)
SYSTEM	TERMINATE (TSYS)
SYSTEM TRACE	DISPLAY (DSYSTR) INITIALIZE (ISYSTR) MODIFY (MDSYSTR) START (SSYSTR) TERMINATE (TSYSTR)



TERMINAL OUTPUT	CANCEL (CTO) DISPLAY (DTO) HOLD (HTO) MODIFY (MDTO) RELEASE (RTO) START (STO) TERMINATE (TTO)
TERMINAL WRITER	DISPLAY (DTW) MODIFY (MDTW) START (STW) TERMINATE (TTW)
TIME	DISPLAY (DTM) MODIFY (MDTIME)
USER REQUEST	CANCEL (CUR) DISPLAY (DUR) FORCE (FUR) HOLD (HUR) MODIFY (MDUR) RELEASE (RUR)
VOLUME	MODIFY (MDV) NAME (NV)
Working Environment	MWENVT
XL_CLASS	DISPLAY (DXLC) MODIFY (MDXLC)





D. The SYS.URCINIT Printer Control File

D.1 Introduction

Printers can be operated under the supervision of a Unit Record Controller, which is a microprogrammed unit capable of simultaneously controlling several such devices connected to the same central processor(s).

The Unit Record Controller can be initialized from a set of parameters contained in a system file called SYS.URCINIT. The SYS.URCINIT file can be accessed and modified by means of a Unit Record Initialization program called URINIT. The URINIT utility can be used, for example, to insert into SYS.URCINIT a new character set and new form control information corresponding to a new print belt.

Certain standard information is supplied by BULL S.A. for insertion into SYS.URCINIT. For instance, in the case of a printer, the following information is supplied at delivery:

- the character set(s) corresponding to the standard print belt image.
- physical format information for standard paper forms: page length, margin width, etc.
- channel positions to be used for vertical format control.
- a printing test routine.

The URINIT utility program can be used for subsequent expansion, modification, deletion, or printing of the information contained in SYS.URCINIT.

In actual use, JCL is used to define the format required for the particular application and the unit record drivers then access the appropriate information for use by the unit record devices during their input/output operations.



D.2 General Programming Considerations

This section provides general programming information, with emphasis on the SYS.URCINIT file and the URINIT commands.

D.2.1 The SYS.URCINIT File

Printers often serve as interfaces between the user and the system: they print out data for human inspection. However, the very nature of this interface means that the devices cannot be entirely automatic; they require a certain amount of user attention.

In order to reduce this requirement to a minimum, while still ensuring all the advantages inherent in the unit record principle, BULL has developed user tools such as the SYS.URCINIT file and the URINIT utility program.

The SYS.URCINIT file allows you to store all the parameters which you will most commonly require for your printer applications. Rather than defining the same requirements over and over again, you can simply reference the appropriate parameters recorded in SYS.URCINIT. The URINIT utility program is designed to enable you to use your SYS.URCINIT file most effectively; with it you can insert, remove, or print out data contained in the file, or you can redefine the inter-relations of certain types of data with other data.

The SYS.URCINIT file is delivered in two forms: a minimum SYS.URCINIT file containing standard parameters, and a source subfile (called UREXT, a member of the SYS.HSLLIB file) which contains generation commands for all the standard parameters for all the devices. You will usually wish to record SYS.URCINIT on your system disk, in the form and containing the information best suited to your site.

The space allocated to the SYS.URCINIT file is two disk cylinders. The delivered content of the file constitutes 10%, the remaining space is left available to the user. SYS.URCINIT is a queued linked file with relative addresses; it can be transferred by the FILSAVE and FILREST utilities (which are described in the Data Management Utilities manual).



IMPORTANT:

It is essential to retain a copy of the source each time you modify the SYS.URCINIT file. Without this copy you will not be able to rebuild the file if a problem occurs. The items included in this file are totally different from one site to another and the source UREXT only enables you to rebuild the minimum required for creation of standard states.



D.2.2 The URINIT Utility

D.2.2.1 The URINIT Statement.

The URINIT utility program is the basic tool used in updating and printing out the contents of the SYS.URCINIT file. The URINIT statement has the following form:

```
URINIT    COMFILE = (sequential-input-file-description)
          [PRTFILE = (print-file-description)];
```

Parameter Description

COMFILE	Describes a sequential input file containing the URINIT commands described below; the commands can be read from diskette or from a source library member. See the <i>JCL Reference Manual</i> for the syntax of a sequential file.
PARTFILE	Describes the assignment to URINIT of a permanent file to be accessed through the SYSOUT access method.

D.2.2.2 JCL Card Arrangement

Updating the SYS.URCINIT file is a single-step job requiring the following JCL statements:

```
URINIT COMFILE = *URC-TAB;
$INPUT URC-TAB;
```

(commands and parameters)

```
$ENDINPUT;
```

For detailed information on JCL statements, see the *JCL Reference Manual*.



D.2.2.3 URINIT Commands

A URINIT command begins with an Order (one of the 7 functions described below) and ends with a semicolon (;). Between these two points are entered specific parameters according to the type of item to which the command relates. The parameters themselves are described in later sections of this document devoted to the individual devices. The command format and command sequence are both free. No continuation character is required and all blanks are deleted.

URINIT ORDERS

URINIT recognizes the following orders:

MOVE	Add a new item to the file
DELETE	Delete an item from the file
RENAME	Rename an item to the file
LIST	Print out the list of item names
PRINT	Print out the contents of all selected items
LINK	Link item together
UNLINK	Cancel links between items

URINIT ITEM TYPES

The orders listed above may be applied to the item types in the SYS.URCINIT file.

There is a specific keyword for each item type. This keyword is 4 characters long; the 3 last letters define the type of device attributed to this item.

There are 2 item types which differ from the rest:

EPRU-IPRU

Which is because their contents are made up of the names of other item types.

Certain item types are interleaved by the system of links shown in Figure D-1. It is not necessary to have these links. It is advisable to define EPRU and IPRU item types instead of linking.

In the following command formats, the ``item-type`` is one of the type keywords given in the first column of Table D-1.



MOVE COMMAND

The MOVE command is used to enter a new item in the file or to modify an item that already exists, it has the following form:

```
MOVE name [REPLACE] item-type = (parameters) [NCHECK];
```

If the item already exists in the file, the REPLACE option must be used to modify the existing item and all the parameters must be given.

If used for a PPRU item, REPLACE must be specified immediately following the name and before PPRU

Example

To define a special form, named BLUE, for the printer:

```
MOVE BLUE FPRU = (parameters);
```

Table D-1. URINIT Items

TYPE KEYWORD	MAXIMUM LENGTH	ITEM TYPE	UNIT RECORD
BPRU	8	BELT	PRINTER
KPRU	2	PRINTER INIT	PRINTER
IPRU	2	INDEX TABLE	PRINTER
FPRU	4	FORM	PRINTER
EPRU	4	EDITING TABLE	PRINTER
CDDU	8	CHARACTER SET	DISKETTE

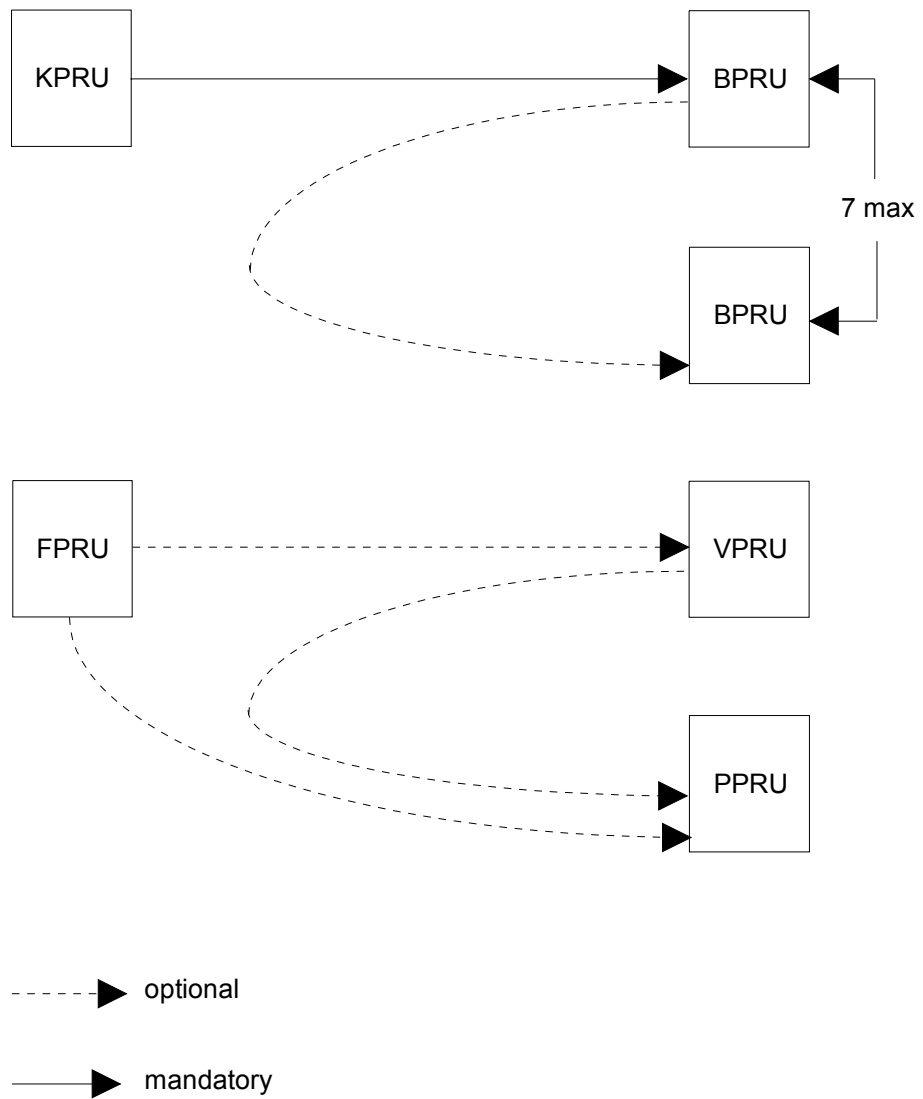


Figure D-1. Item Links



DELETE COMMAND

The DELETE command is used to remove an item already existing in the file. This command has the following format:

```
DELETE name item-type;
```

Example

to delete form BLUE from the printer:

```
DELETE BLUE FPRU;
```

RENAME COMMAND

The RENAME command is used to update the corresponding record in the directory by changing its name. It has the following form:

```
RENAME name NEW = name item-type;
```

Example

The user has a form called BLUE which he wishes to rename RED:

```
RENAME BLUE NEW = RED FPRU;
```

LINK COMMAND

NOTE:

It is recommended that you use the Index Table (IPRU to KPRU/BPRU) instead of the LINK command.

The LINK command is used along with character set and index table to define relationships among items. The items must already exist in the file before LINK can be used. This command has the following form:

```
LINK { [KPRU = name BPRU = ( name [, name] ... ) ] }  
      { [BPRU = ( name, name [, name] ... ) ] }
```

NOTES:

1. At least one of the optional parameters must be entered.
2. The belts will be linked in the order specified.



Example

to link print belt E502 to character set S2:

```
LINK KPRU = S2 BPRU = E502;
```

UNLINK COMMAND

The UNLINK command is used to cancel relationships among items. It has the following form:

```
LINK { [KPRU = name BPRU = ( name [, name] ...)] }
      { [BPRU = ( name, name [, name] ...)] }
```

NOTES:

1. At least one of the optional parameters must be entered.
2. The belts must have been linked in the order specified.

Example

to cancel the link between print belt E502 and character set S2:

```
UNLINK KPRU = S2 BPRU = E502;
```

LIST COMMAND

The LIST command causes each item in the catalog to be printed out. The command has the form:

```
LIST [*];
```

The printed listing has the format:

```
TYPE : Unit Record name-type
*ITEM-MANE = XXXXXXXXX
LINK-NAME = XXXXXXXXX     TYPE = XXXX

*ITEM-MANE = XXXXXXXXX
LINK-NAME = XXXXXXXXX     TYPE = XXXX

TYPE : Unit Record name-type
*ITEM-MANE = XXXXXXXXX
LINK-NAME = XXXXXXXXX     TYPE = XXXX
```



PRINT COMMAND

The PRINT command causes one or more items in the file to be printed. It has the format:

```
PRINT
  { PRU }
  { { name } }
  { { * } item type }
```

That is, an asterisk followed by PRU will cause a printout of all items in the file related to printers. Or, alternatively, a given item having a specific name can be printed; or, if an asterisk is first entered, all KPRU items can be printed.

The printout has the format:

```
ITEM-NAME = XXXXXXXX TYPE = XXXX
LINK-NAME = XXXXXXXX TYPE = XXXX
```

```
DATA
****
```

```
KEYWORD = XXXXXXXX depending upon the item type
KEYWORD =
```

D.2.2.4 Command Restrictions

The following notes and restrictions should be kept in mind when using the URINIT commands.

MOVE

- The item name must not already exist (except for the case of the REPLACE option)
- For the Index Table (IPRU) and editing table (EPRU) the name specified as parameters must exist.
- There is no name independence between the index table (IPRU) and character set (KPRU) or between the editing table and the form (FPRU);



DELETE

- The item name must exist, otherwise a warning will be sent.
- If other items are linked to the delete item, the links will be destroyed.
- If the item is a linked belt, the other items will be relinked.
- Character set I1 and form 0000 must not be deleted.
- If the item is a translation table (KRPU) or a belt (BPRU) the contents of the index table which refer to it are modified (deletion of the name). After several deletions of BPRU when no more BPRUs exist in the index table it is deleted.
- If the item is a printer (PRU) the contents of the editing table are modified.
- If the item is a form, the editing tables are also deleted.

RENAME

- The old item name must already exist.
- The new item name must not already exist.
- If the other items are linked to the item, the links are updated.
- Character set I1 and form 0000 must not be renamed.
- If the item is a translation table or a belt, the contents of the index tables, which refer to it, are modified.

LINK

- All the items to be linked must already exist.
- These items must not already be linked (except for the last).

UNLINK

- All the items to be unlinked must already exist and must be linked as described in the LINK command.
- If the last belt specified in the UNLINK command has a link, then the first belt so specified (or its character set) will be linked to that belt which was previously linked to the last one.
- Character set I1 must always have a belt linked to it.



D.2.3 Parameter Overriding

Many of the parameters originally specified (or assigned by default) for use by the unit record devices are subject to later modification by the user, especially by means of the JCL statement DEFINE. The specific parameters and overriding methods are discussed under the individual descriptions of each device type, in the following sections of this documents. See the *JCL User Guide* or the *IOF Terminal User's Reference Manual* for further information.

D.2.4 Protected Character Strings

Certain of the tables and parameters used by URINIT are created in the form of a character string which must be protected; that is, it must be enclosed between single quote characters (').

For more information on protected character strings, see the *JCL User Guide*.

D.2.5 Direct Use of *Devices

For most applications, the recommended method for using the unit record devices is the indirect mode, in which the program performs a spooling function via the system files SYS.IN and SYS.OUT. See the *JCL User Guide*. The printer, can also be accessed directly by specifically assigning to the device the internal-file-name defined in the program.

D.2.6 Related Information

For general information on programming and execution techniques, see the *COBOL User Guide* and the *JCL User Guide* or the *IOF Terminal User's Reference Manual*. For information on return codes and error messages, see the *Messages and Return Codes Directory*.

For further information on data formats (including SARF and SSF), see especially the *Library Maintenance Reference Manual* and the *COBOL User Guide*.



D.3 Line Printer

This section covers the parameters which can be used in URINIT to describe the character set (KPRU), belt arrangement (BPRU), print form (FPRU), vertical format control facility (VPRU), index table (IPRU), and editing table (EPRU).

D.3.1 Device Naming and Allocation

In many applications it is irrelevant which physical printer handles the output; if an installation contains several printers, the output may be routed to any one of them. However, if you so require you can select a specific printer model (not a recommended technique) or a printer having certain specific characteristics, by making use of the DEVCLASS parameter in any of the JCL statements SYSOUT, WRITER, or OUTVAL. (If the printer is being used directly, the ASSIGN statement is used for this purpose).



The characteristics which can be specified are summarized in the following Table.

Table D-2. Printer Characteristics

GENERIC NAME	DEVICE	SPEED Lines per minute	PRINT MECH.	Number of HAMMERS or Number of Columns	HORIZ. PRINT DENSITY Char. per inch	VERTICAL PRINT DENSITY		FORM HEIGHT		FORM WIDTH	
						6	8	Min	Max	Min	Max
PR54	PRU4112	900	Belt	120/136/160	10	6	8	Min	Max	Min	Max
	PRU4113	1200				Yes	Yes	4	24	4	22
PR88	PRU4120	325	Belt	120/136	10	Yes	Yes	4	16	4	19
	PRU4121	650									
PR90 See note (1)	PRUD001 See note (1)	Up to 1200	See note (2)	132 See note (3)	10	Yes	Yes	1	24 See note (4)	3	16
PR800 See note (1)	PRUD002 See note (1)	Up to 800	See note (2)	132 See note (3)	10	Yes	Yes	1	24 See note (4)	3	16
PR900 See note (1)	PRUD003 See note (1)	Up to 1200	See note (2)	132 See note (3)	10	Yes	Yes	1	24 See note (4)	3	16
PR701 See note (1)	PRUG004 See note (1)	Up to 500	See note (2)	136 See note (5)	10 to 20	Yes	Yes	1	12	3	16
PR801 See note (1)	PRUG008 See note (1)	Up to 800	See note (2)	136 See note (5)	10 to 20	Yes	Yes	1	12	3	16
PR901 See note (1)	PRUG007 See note (1)	Up to 1200	See note (2)	136 See note (5)	10 to 20	Yes	Yes	1	12	3	16

NOTES:

1. Dot matrix printer.
2. "Software" belt.
3. 132 in standard configuration but can be set to up to 160 on the printer panel.
4. 11 in standard configuration but can be set to up to 24 on the printer panel.
5. 136 in standard configuration but can be set to up to 272.



D.3.2 Printer Attributes

Table D-3. Printer Attributes

Device Type	Speed Attribute		Hammer Attribute or Number of Columns		Model Attribute	
	Name	Meaning	Name	Meaning	Name	Meaning
PR	FI	Fast Imperative	H<n>*	N hammers are needed	PR54	PRU4112
	A	Any speed				PRU4113
	SI	Slow Imperative				PRU4120
					PR90	PRUD001
						PRUD002
						PRUD003
					PR701/ 801/901	PRUG004
						PRUG007
						PRUG008

SPEED ATTRIBUTE

FI: Fast Imperative - You request the fastest printer.

NOTE:

If you specify FI, the request will be queued for that type of printer if the printer is not available.

HAMMER ATTRIBUTE

H <n> n may have any value from 1 to 160. Device Management will allocate a printer with at least as many hammers as specified.

NOTE:

"H112" will always be interpreted as a model attribute, not as a hammer attribute. In practice this is not restrictive, since less than 120 hammers will rarely be specified.



This facility is provided to make it possible to adjust the printer line length to the printed line length as specified in the program, so as to avoid line folding or similar complications. For instance, assume that the program contains printed lines of up to 129 characters in length, and the installation is equipped with two printers: printer PRO1 has 160 hammers and printer PR02 has 120 hammers. You can then specify:

```
DEVCLASS = PR/H129
```

to ensure that this output will be directed to printer PRO1.

Two additional points should be noted in this respect:

1. In the above example, if the program should happen to contain a printed line of, say, 135 characters, the line will be folded at character 130 despite the fact that the printer has 160 hammers. In other words, the H attribute really specifies print columns rather than hammers.
2. If the MARGIN option is used (see *Form Definition* below), the number of columns reserved for the margin must be taken into account in specifying the value of H.

MODEL ATTRIBUTE

You are strongly advised not to specify the printer model. The other attributes are retained here only in the interest of compatibility with previous GCOS releases.

ATTRIBUTE PRIORITIES

Attribute definitions are processed in the following order of priority, from high to low:

1. Model.
2. Number of hammers.
3. Printing speed.

Examples

```
DEVCLASS: PR/H120
```

A printer having 120 or more hammers will be allocated.

```
DEVCLASS: PR/PI/H129
```

The fastest printer is allocated provided that it has at least 129 hammers. If the hammer number is insufficient, the printer speed is not taken into account.

```
DEVCLASS = PR
```

Any printer will be allocated.



D.3.3 Printing Procedures

You have a choice between the following output techniques.

- Direct use of the printer. The output is printed at the same time as it is created by the user program.
- Use of an intermediate file before printing (spooling technique). The output created by the user program is stored in a temporary system subfile (known as a standard SYSOUT subfile) or in a permanent user file (referred to as a permanent SYSOUT file). The actual printing is performed by a system element called the Output Writer, and may take place long after the output was created.

The spooling technique is generally preferable, since it makes more efficient use of processing time and printer capabilities. For detailed information on SYSOUT files and the Output Writer, see the *JCL User Guide*.

The user program is independent of the output method used; the output method is selected in the JCL that initiates the program.

The following example illustrates these different techniques.

Assume that we have a user program written in COBOL, containing

```
-----  
SELECT OUT  
ASSIGN TO OUTFILE _ SYSOUT  
.  
.  
.  
FD OUT  
RECORD CONTAINS 129 CHAR  
.  
.  
.  
-----
```

For each possible output method selected, the JCL is given below.



D.3.3.1 Direct Use of the Printer

```
-----  
$ JOB...;  
STEP PROGA, ...;  
ASSIGN OUTFILE, DEVCLASS = PR/H129, MEDIA = I20001;  
ENDSTEP;  
$ENDJOB;  
-----
```

REMARKS:

1. A printer with at least 129 hammers will be assigned to the job for the duration of the step.
2. Index table (or character set) I2 and editing table (or form) 0001 will be used in printing this output. (Index tables/character sets and editing tables/printing forms are discussed later in this section.)

D.3.3.2 Use of a Standard SYSOUT Subfile

```
-----  
$JOB...;  
STEP PROGA, ...;  
SYSOUT OUTFILE, DEVCLASS = PRO, MEDIA = I20001;  
ENDSTEP;  
-----
```

REMARKS:

1. No ASSIGN statement is required because a standard SYSOUT subfile is used.
2. The output data will be deleted from a SYSOUT file after it has been printed by Output Writer.
3. The SYSOUT statement indicates that the index table and editing table to be used are different from the standard ones. If the Index table set and editing table were standard (I10000 in most installations) then the SYSOUT statement could be omitted.

The MEDIA parameter is defined by the SYSOUT, ASSIGN and WRITER JCL statements. For the printer MEDIA is a six figure character value which specifies the print belt, character set and paper identification; default value is I10000.



To change the define parameters within the step enclosure, SYSOUT is used, whereas WRITER is used outside the step enclosure. DEFINE does not use MEDIA but defines parameters from inside the step enclosure.

media name is made up of two parts: XXYYYY

XX identifies either a character set code or an index table

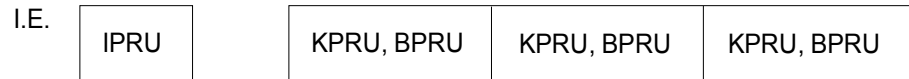
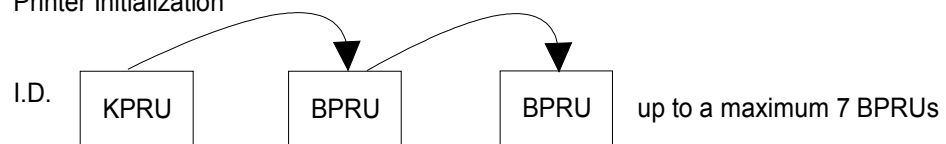
YYYY identifies either a form number or an editing table

i.e., ASSIGN DEVCLASS = device class MEDIA =
 XXYYYY

These two parts refer to tables stored in the system; as a consequence, these tables must be stored in the system before being used by a user program.

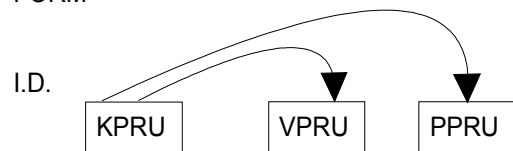
XX

Printer Initialization



—————▶ 7 couples

YYYY
 FORM



XX KPRU + IPRU ARE MUTUALLY EXCLUSIVE
 YYYY FPRU + EPRU ARE MUTUALLY EXCLUSIVE

Figure D-2. Item Links



D.3.3.3 Use of a Permanent SYSOUT File

```
-----  
$ JOB...;  
STEP PROGA,...;  
ASSIGN OUTFILE, PRINT, DEVCLASS = MT/T9, MEDIA = 1601;  
SYSOUT OUTFILE;  
ENDSTEP;  
$ENDJOB;  
-----
```

NOTES:

1. The output created by the PROGA step is stored on a magnetic tape having the name 1601.
2. Since the MEDIA parameter is not present in the SYSOUT statement, the standard character set and paper form are to be used.
3. The output data will be printed by the Output Writer. If the SYSOUT statement had been omitted, this job would not print the output data.
4. After it has been printed, the output data remains available on the magnetic tape and can be printed again in another job by means of the JCL command WRITER:

```
$JOB...;  
WRITER (PRINT, DEVCLASS = MT/T9, MEDIA = 1601);  
$ ENDJOB;
```

D.3.4 Index Table - as an Alternative Character Set

An Index Table is a set of 7 couples (maximum) translation table-belt image. A Translation Table may be associated with each belt image, but is not mandatory for a belt image to have a Translation Table. By this means it is possible to put belt images or different classes (i.e. 48 characters, 63 characters, 94 characters, 64 characters) in the same group; this is not possible with the character set. The space value of all belts in the same index table must be the same.

This can best be understood in an example:

Assume that the installation has two belts:

- the first belt is able to print 63 EBCDIC characters (upper-case letters, numeric digits and special characters). It is called E500,
- the second belt is able to print 94 EBCDIC characters (lower-case letters, capital letters, numeric digits and special characters). It is called E600.



Now, a memory byte containing a character code may be one of the following types:

- printable by both belts (e.g., upper-case letters, numbers),
- printable by one belt only (e.g., lower case letters),
- not printable by either belt.

Assume that a user program creates output records containing both upper-case and lower-case letters. Belt E600 must be used to print this output correctly: however, it might be useful to be able to print the output using belt E500, in the event that belt E600 should be unavailable. This can be done by translating every lower case letter into the corresponding upper-case form.

From the point of view of the software two index tables are defined:

- an index table called I1 which contains the E500 belt image with a translation table and the E600 without the translation table
- an index table called I2 which contains only the E600 belt image.

With the index table I1, the output is printed either with the E600 or E500 belt. If there are some lower-case letter codes in the output and the E500 belt is mounted on the printer, the lower case letters will be translated into upper-case. If the E600 is mounted, upper and lower-case will be printed. Only one belt image is necessary to describe a belt. As can be seen from this example, the index table offers more flexibility than the character set.

In this case, if the belt mounted on the printer is E500 and the index table specified is I2, then the E600 belt will be requested for mounting.

To continue our example, the URINIT run which would define the index table would be as follows:

```

MOVE      J1          ,  KPRU = ( HEXTBLS = '-' );

MOVE      EH112-63   ,  BPRU = ( BELTCODE = 0
                           ,  SP VALUE = 40
                           ,  IN VCHAR = 70
                           ,  LENGTH   = 63
                           (  HEXTBLS = '-' );

MOVE      E600       ,  BPRU = ( BELTCODE = 600
                           ,  SP VALUE = 40
                           ,  IN VCHAR = 70
                           ,  LENGTH   = 240
                           (  HEXTBLS = '-' );

```



```

MOVE   E700      ,  BPRU = ( BELTCODE = 700
                          ,  SP VALUE = 40
                          ,  IN VCHAR = 70
                          ,  LENGTH   = 80
                          (  HEXTBL = '-' ) );

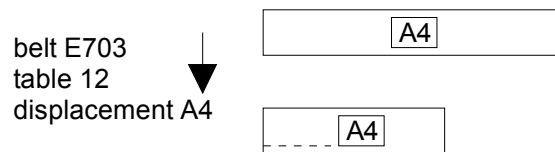
MOVE   I1        ,  IPRU = ( ( KPRU = J1 , BPRU = E700)
                          ,  ( KPRU = J1 , BPRU = E500)
                          ,  ( KPRU = J1 , BPRU = E501)
                          ,  ( KPRU = J1 , BPRU = E502)
                          ,  ( KPRU = J1 , BPRU = E503)
                          ,  ( BPRU = E600) );

MOVE   I2        ,  IPRN = ((BPRU = E600)) ;

```

All the characters found in all the belts linked to or associated with a translation table must also be found in the table at the location specified by their hexadecimal value.

EXAMPLE:



If in table I2, 4A is replaced by DO it would be a fatal error.

You can translate any character not located in the belts into a belt character.

☐

EXAMPLE:

to translate code "20" into letter A(C1):

- "20" must not be found in the belt images linked to the table.
- "C1" must be found in all the belt images linked to the table.
- the "C1"th entry of the table must contain C1.
- the "20"th entry of the table must contain C1.

☐



D.3.5 Printer Character Sets

A character set is a group of characters that can be printed out by a given printer belt or drum. It is recorded as a configuration of coded bytes in memory, and may or may not involve a translation table.

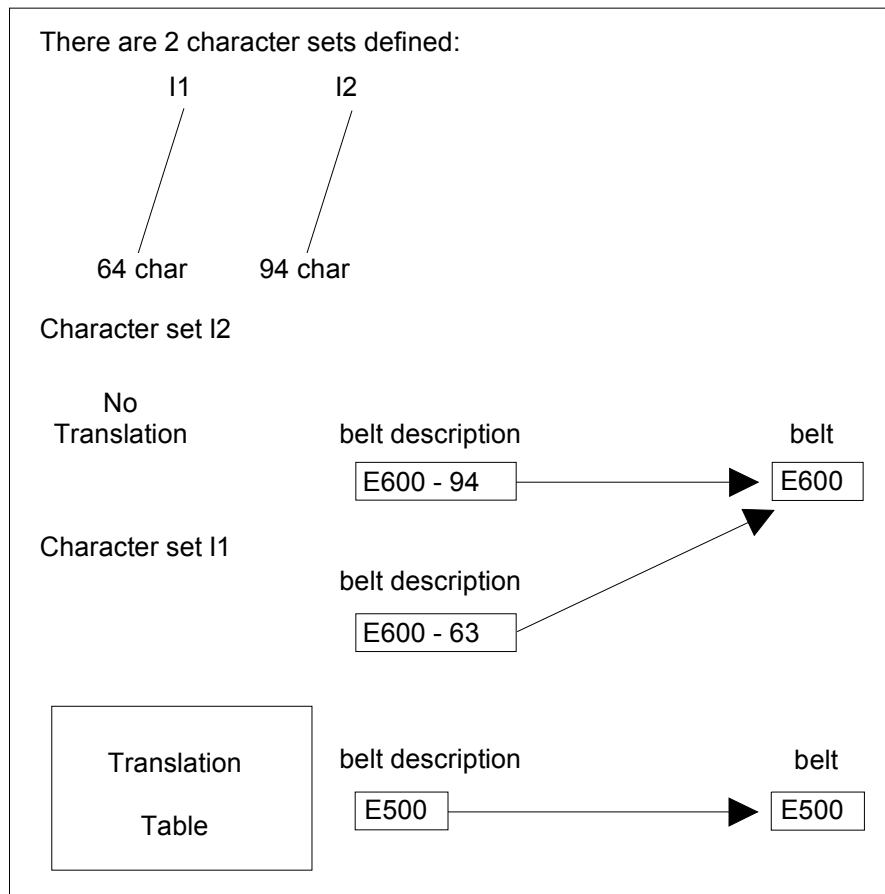


Figure D-3. Character Set Rules

NOTES:

1. Belt descriptions should be provided for E600: one for character set I2, and one for character set I1.
2. When I1 is requested, the lower-case letters are translated into capitals even if E600 is used for actual printing.
3. Translation tables and belt descriptions are correlated.

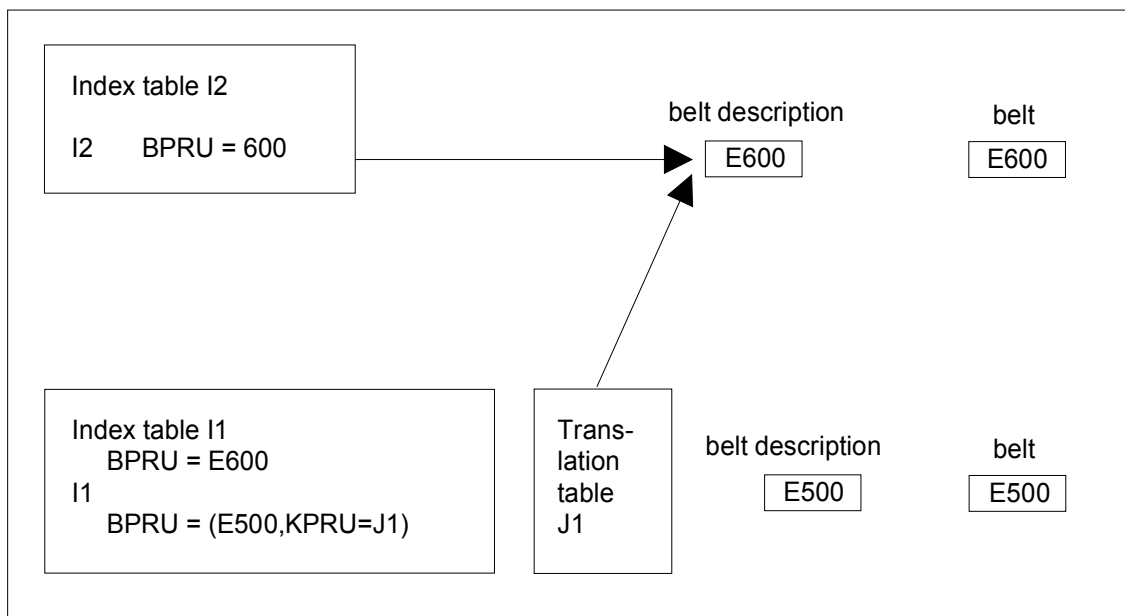


Figure D-4. Index Tables Rules

NOTES:

1. Only one belt description for belt E600.
2. When I1 is requested and belt E600 is used, the translation table is not used i.e. lower-case letters are printed as such.

A character set therefore consists of:

- a given group of characters,
- an optional translation table to establish a correspondence between every memory byte configuration and:
 - a character belonging to the defined group,
 - a character used to indicate an invalid code (i.e., memory byte configuration not corresponding to any character in the set).

For our purposes, a character set is defined by:

- a belt segment image table,
- an associated translation table (optional).



A belt segment image table is an image in memory of one segment of a printer belt; you will recall that a belt consists of several segments, each containing all the characters the belt is able to print. The segments are repeated to form an endless loop.

Assume, for example, that belt E500 consists of segments containing:

```
A B ... Z 1 2 ... 9 + - ? ...
```

whereas a belt E600 segment contains:

```
A B ... Z a b ... 9 + - ? ...
```

Belt E600 could be used to print both the XY and the WZ character sets. When used to print WZ, the belt image (in hexadecimal code) will be:

```
HEXBELT = `C1C2 ... E9 81 82 ...A9 F1 F2 ...F9 4E 60 6F ...'
```

For XY, the corresponding belt image will be:

```
HEXBELT = `C1C2 ... E9 00 00 ...00 F1 F2 ...F9 4E 60 6F...'
```

The "invalid" characters a b ... z in the above example have been assigned the value "00".

To continue our example, the URINIT procedure which defines these two character sets and their related belts is:

```

MOVE E500, BPRU = (BELTCODE=500, SPVALUE=40, INVCHAR=7C,
  HEXBELT=`C1 C2 ... E9 F1 F2 ...F9 4E 60 6F ...');

MOVE E600-63, BPRU = (BELTCODE=600, SPVALUE=40, INVCHAR=7C,
  HEXBELT='C1 C2 ... E9 00 00 ...00 F1 F2 ...F9 4E 60 6F
  ...');

MOVE E600-94, BPRU = (BELTCODE=600, SPVALUE=40, INVCHAR=7C,
  HEXBELT='C1 C2 ... E9 81 82 ...A9 F1 F2 ...F9 4E 60 6F
  ...');

MOVE J1, KPRU = ('00 ... C1 C2 ... C1 C2 ....');

MOVE J2, KPRU = ();

LINK KPRU = J1, BPRU = (E500, E600-63);

LINK KPRU = J2, BPRU = E600-94;

```



The first MOVE command, above, establishes in SYS.URCINIT a belt image having the name E500, associated with physical belt type 500. The code for a space (blank) character is defined as hexadecimal 40 (SPVALUE). The code for an invalid character (INVCHAR) is hexadecimal 7C; that is, any invalid byte configuration in memory will be printed out as \$. The characters contained in each segment of the belt are listed, in the same order as they appear on the physical belt, in the HEXBELT parameter.

The second MOVE command establishes another belt image, named E600-63. This belt image is associated with physical belt type 600, which itself contains the full 94-character EBCDIC set, but all the lower-case letters are here defined as non-printable by means of 00 codes in the HEXBELT list. The same values are used for space and invalid characters as in E500.

The third MOVE command establishes a belt image named E600-94, also associated with physical belt 600, but this time allowing both the upper-case and lower-case letters to be printed. Again SPVALUE and INVCHAR are the same as in the other belt images.

The next two MOVE commands establish character sets J1 and J2 in SYS.URCINIT. Character set J1 (no lower-case letters) has a translation table associated with it so that it can be used with the E600-63 belt as well as with the E500 belt.

Finally, there are two LINK commands to establish the relationships between character sets and belt images: J1 is linked to belt images E500 and E600-63, J2 to belt image E600-94.



Table D-4. Character Sets and their Belt Images

INDEX TABLE NAMES	CONTENTS	RELATED BELT
	63/94 EBCDIC characters	E703, E500
I1	printing of lower case letters if belt E600	E600 for PRU 1200/1600 EH112-63 for PRU 600/800 B600 for PRU 0615
I2	94 EBCDIC characters (upper and lower case letters)*	E600 for PRU 1200/1600
I3	96 characters	EH112-96 for PRU 600/800 B600 for PRU 0615
I4	64 characters	E700
I5	63 EBCDIC characters	E501, E502, E503, E504, E505, E511 E511, E512, E513 for PRU 1200/1600
I7	64 characters (of which 15 CMC7)	E151-64 for PRU 0711
G1	64 characters G100	G700, G500 for G600 PRU 1200/1600 GH112-63 for PRU 600/800 G601 for PRU 0615
G2	64 characters G100	G501, G502 for PRU 1200/1600
G3	64 characters G100	GH112-96 G601
G7	64 characters G100	G151-64

* Kept for compatibility



D.3.5.1 Standard Index Tables

The following standard index tables are supplied by BULL:

I1, I1, I3, I4, I5, I7; and, for G100, G1, G3, G7.

- I1 identifies the EBCDIC set. All lower-case letters are changed to capital letters except for the E600. The commercial at symbol (@) is substituted for all non-printable characters. The standard space code (blank character) is hexadecimal 40.
- I2 identifies the standard 94-EBDIC set, including lower-case and upper case characters.
- I4 identifies the standard 64-character set.
- I5 identifies the 63-character EBCDIC code (like I1, above).

These character sets and their associated belt images are summarized in Table D-4.

D.3.5.2 KPRU Parameter

The SYS.URCINIT item KPRU defines the translation table. It contains a single parameter, HEXTB_L. In the URINIT MOVE command, for example, it has the form:

```
MOVE table-name, KPRU = ([HEXTBL = 'Hexa _ 512']);
```

HEXTBL is the hexadecimal translation table (optional) used to translate from internal code to print belt image. It contains 256 entries, each entry being a 2 digit value. When creating or modifying a translation table, keep the following points in mind:

- All the characters found in a translation table must also be found in the belts linked to that table, or in the associated belt in the index table.
- Any character found in any belt linked to or associated with a translation table must be found in the table at the location specified by its hexadecimal value. For example, character 4A in belt E500 must be found in table J2 at displacement 4A.
- Any code not already referring to a belt character can be related to a belt character.

**NOTES:**

1. If this type of translation function is not required, then a translation table is not necessary.
 If a belt character is to be changed into another belt character, then the belt image rather than the translation table should be changed.
 For example, if the letters A and S are to be interchanged, when A is CI and S is EI, simply change S to CI and A to EI in the belt image.
 If a certain character is never to be printed, insert hexadecimal ``00`` in the corresponding entry of the belt image.
2. The character string must be protected. (See Section D.2.5 *Protected Character Strings*)

D.3.6 Belt Definition

A character set can have up to seven belt image records associated with it, each identified by a BPRU item.

An index table can have up to seven couples of translation tables and belt images.

D.3.6.1 BPRU Parameters

Again taking the MOVE command as an example, BPRU has the following form:

```

MOVE table-name, BPRU = (BELTCODE = 'digits-3',
                        SPVALUE = 'hexa-2',
                        INVCHAR = 'hexa-2',
                        HEXBELT = 'hexa-n belt image',
                        CHARBELT = 'char-n belt image'
                        [LENGTH = { 240          } ]
                        [          { 'digits-3' } ] );
  
```

BELTCODE	Physical identification of the belt, or drum code (0000) in the case of a drum-type printer.
SPVALUE	Space code value; character to be interpreted as a space and never printed, even if it belongs to the belt image.
INVCHAR	Value of substitution character to be printed if an invalid character is requested (i.e., a character which does not belong to the belt image). The substitution character specified must be a printable character.



HEXBELT-CHARBELT	Belt segment image, in hexadecimal or alphanumeric characters. It does not refer to any physical characteristic of the drum, as the belt codes do in the case of a belt.
LENGTH	Length of the belt image, in characters (bytes). The default value is 240.

NOTES:

1. With respect to CHARBELT, it is possible to introduce an internal code different from EBCDIC.
2. For both HEXBELT and CHARBELT, the character string must be protected. (See Section D.2.5 *Protected Character Strings*)

D.3.6.2 IPRU Parameters

This table is used to associate several belt images (BPRU) with their translation tables (KPRU)

```

-----
MOVE table name, IPRU ((BPRU = name [,KPRU=name])      1 couple
                        [(BPRU = name [,KPRU=name])...] 2 couples
                                                                .
                                                                .
                                                                .
                                                                etc

```

BPRU = name of belt image

KPRU = name of translation table

NOTE:

IPRU is better and simpler to use than the links between KPRU and BPRU (character set definition).

For example, an installation has two belts:

- the first belt, named E500, has 63 characters,
- the second belt, named E600, has 94 characters.



From the point of view of the software, the two index tables are defined as follows:

- Index table I1. I1 contains the E500 belt image with a translation table (lower case to upper case) and the E600 without translation.

Using index table I1, the output can be printed either with E600 or E500. If there are lower case letter codes in the output and E500 is mounted on the printer, the lower case letters will be translated into upper case. If E600 is mounted, upper and lower case will be printed.

- index table I2, which is used to print with the E600 belt only. If the belt mounted on the printer is E500, the E600 belt will be requested, as explained below.

If the index table specified is not supported by a currently mounted belt, a message will request the operator to mount the proper belt, whose identification code is given in the message. In the case of a non-standard belt, and especially one modified by the user, care must be taken to ensure that a change in belt image is reflected by a change in belt code, and that a correlation list is available to the operator.

As an example, belts E700, E501, associated with character set I1, would be defined as follows:

```

MOVE E700, BPRU = (BELTCODE = 700, SPVALUE = 40,
                  INVCHAR = 7C, CHARBELT = '01234.....');

MOVE E501, BPRU = (BELTCODE = 501, SPVALUE = 40,
                  INVCHAR = 7C, CHARBELT = '5678.....CDEF" ');

MOVE I1, IPRU = ( (KPRU = J1, BPRU = E700)
                 (KPRU = J1, BPRU = E501)...);

```

D.3.7 Form Number Encoding

The form is specified by the user in the MEDIA parameter of the ASSIGN JCL statement. However, the form number is only part of it (the last four bytes), the first two bytes being reserved for the character set code or the index table.



D.3.8 Editing Table - as an Alternative to Form-Name

An editing table is a group of three elements:

- form number, which is mandatory,
- vertical form unit,
- printing test.

The last two elements are optional. When the three elements are present, it is equivalent to a form having a link to a VFU (Vertical Form Unit), this VFU having a link to a printing test.

D.3.9 Form Definition

The form (i.e., the paper to be used for printing) is treated by the system as an unlabelled volume which must be mounted at the beginning of a work session. It is identified by a form number, which is specified in the second field of the MEDIA parameter. (See *Device Naming and Allocation*, above.) The form number is recorded in SYS.URCINIT, and certain characteristics and printing rules can be recorded at the same time, as described below. Three standard forms are predefined by GCOS, as explained below.

D.3.9.1 Standard Editing Table

The editing table is initially defined by a code in the last four bytes of the MEDIA parameter at assignment time. The printing parameters of the form which belong to the standard editing table are:

```
Code ``0000``

Form height           : 66 lines
Margin                : 0
Head of form          : 5
Full form 1           : 60
Printing density      : 1/6 inch
Vertical format stops: Channel 1, level 05
                    : Channel 2, level 10
                    : Channel 3, level 40.
```

Code ``0000`` is identical with the above except for printing density, which is 1/8 inch.



D.3.9.2 EPRU Parameters

EPRU parameters contain the names of the items defined for printing:

```

FPRU = name
[, VPRU = name]
[, PPRU = name]

```

```

FPRU = name of existing form
VPRU = name of an existing vfn
PPRU = name of existing printest

```

EXAMPLE:

```

MOVE      F000          , FPRU = ( FORMHT = 66+n1
                               , HOF      = 5
                               , FF1      = 60
                               ) ;
MOVE      F001          , FPRU = ( FORMHT = 88
                               , HOF      = 5
                               , FF1      = 80
                               , PRDEN    = 8
                               ) ;
MOVE      H-DF-VFU     , VPRU = ( CH1 = 5
                               , CH2 = 10
                               , CH3 = 40
                               ) ;
MOVE      0000         , EPRU = ( FPRU = F0000
                               , VPRU = H_DF_VFU
                               ) ;
MOVE      0001         , EPRU = ( FPRU = F001
                               , VPRU= H_DF_VFU
                               ) ;

```

□



D.3.9.3 FPRU Parameters

The following parameter keywords are specified in item FPRU to define the physical characteristics of the paper mounted on the printer.

```

MOVE form-name, FPRU = (FORMHT = 'digits_3'

                        [           { 0           } ]
                        [,MARGIN = {           } ]
                        [           { margin-value } ]

                        [           { 1           } ]
                        [,HOF =   {           } ]
                        [           { head-of-form } ]

                        [           { FORMHT      } ]
                        [,FF1 =   {           } ]
                        [           { full-form-1 } ]

                        [           { 6 } ]
                        [,PRDEN = {   } ] ) ;
                        [           { 8 } ]

```

FORMHT	Height of the paper form in number of lines; 3 digits maximum.
MARGIN	Number of columns to be skipped at the left of the paper form; 3 digits maximum.
HOF	Head of form. Line number of the first line printed in a output page; 3 digits maximum.
FF1	Full form level 1. Line number from the upper fold of the paper form. Used for indicating a full page; 3 digits maximum.
PRDEN	Printing density in number of lines par inch. The distance between any two consecutive lines may be 1/6 inch or 1/8 inch; ``6`` in this parameter means 1/6 and ``8`` means 1/8.



The following remarks may be helpful in understanding how these parameters are used:

- Assume that form 0009 is used, with the values FORMHT = 66, MARGIN = 0, HOF = 1, and FF1 = FORMHT.

As indicated in *Figure D-1*, line 1 is the first possible line on the page, lying just beneath the fold of the paper form, and line n is equal to line 66 (defined by FORMHT). Let P be the current printing position on the paper. Now, if the COBOL program specifies:

```
WRITE record-1 AFTER ADVANCING PAGE
```

then record-1 will be printed in the position indicated as P1 (i.e., line 1 of the next page). In the same fashion, if the program specifies:

```
WRITE record-1 AFTER ADVANCING 4 LINES
```

then P0 and P1 would be separated by three blank lines.

- Now assume that the Head of Form and Full Form 1 parameters are as normally specified in form 0000 (HOF = 5, FF1 = 60) and that MARGIN = 7.

In effect, these parameters define a logical page lying within the physical page, as illustrated in *Figure D-2*. The first line which will actually be printed (logical line 1) will be on physical line 5, as defined by HOF, and the last line printed on the page will be line 60 (defined by FF1). As a result, each printed page will have four blank lines left at the top and six blank lines left at the bottom. In addition, seven columns will be left blank at the left-hand side to serve as a margin. Now if the COBOL program specifies

```
WRITE record-1 AFTER ADVANCING PAGE
```

then record-1 will be printed on line 5 of the next page; if the program specifies:

```
WRITE record-1 AFTER ADVANCING 4 LINES
```

then the two printed lines will still be separated by three blank lines, as in the previous example.

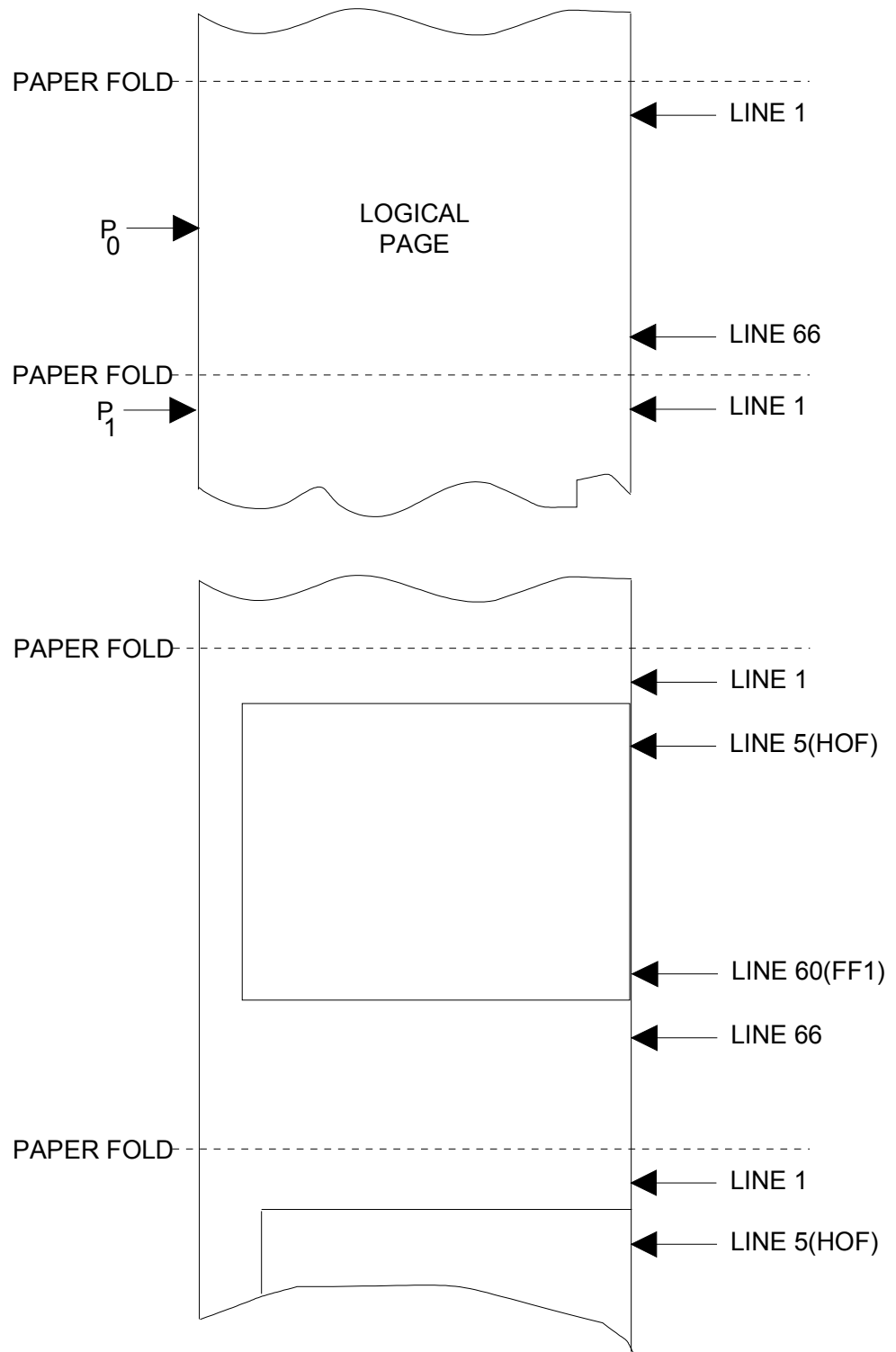


Figure D-5. Form Parameters



The MEDIA parameter (see *Device Naming and Allocation*, above) is used to define both the index table and the editing table. For example, index table 12, editing table 0003 would be specified as follows:

```
SYSOUT PRINT, DEVCLASS = PR, MEDIA = 120003;
```

Use of the URINIT MOVE command for the editing table and form parameters in the SYS.URINIT file can be illustrated as follows:

(Recording of standard print form F000)

```

URINIT COMFILE = *CATA;
$INPUT CATA;
MOVE F000, FPRU = (FORMHT = 66, PRDEN = 6, HOF = 5, FF1 = 60)
MOVE 0003, EPRU = (FPRU = F000);
$ENDPUT

```

See also *Parameter Overriding* below.

D.3.10 Vertical Format Definition

VPRU Parameters

The parameters specified in item VPRU define the distances, in line spaces, between the upper fold of the paper form and the various form feed stop levels. This facility therefore provides a software simulation of the VFU tape loop often used in printers for vertical format control.

At least one of the following parameters must be specified:

```

MOVE table-name, VPRU = ([CH1 = (channel-1-1 [, channel-1-2] ...)]
                        [, CH2 = (channel-2-1 [, channel-2-2] ...)]
                        [, CH3 = (channel-3-1 [, channel-3-2] ...)]
                        .
                        .
                        [, CH12 = (channel-12-1 [, channel-12-2] ...)]);

```

CH1 through CH12 refer to channel numbers; the content of each CHn is a list of all stop levels in that channel. The levels for a given channel must be entered in ascending order, and the highest level must be less than the form height value. Up to 20 levels are allowed, distributed over 12 possible channels.



Assume, for instance, that stop levels are desired for line 5 on channel 1 and for lines 20 and 48 on channel 2. This information would be entered into SYS.URCINIT as follows:

```
URINIT COMFILE = * CATA;
$INPUT CATA;
MOVE VFUI, VPRU = (CH1 = (05), CH2 = (20, 48));
$ENDINPUT;
```

VFU parameters are frequently associated with the form specifications, so that whenever the form is used the associated VFU is also available for use. To associate VFU1 to paper form F001 the following command would be given:

```
MOVE 0001, EPRU = (FPRU = F001, VPRU = VFU1);
```

Certain VFU formats are normally associated with standard paper forms, as noted in ``Form Definition``, above.

D.3.11 Printing Test Definition

The printing test consists of one or more lines which are printed out whenever the operator presses the PRINTER TEST switch (e.g., at the beginning of a job or following a fault condition such as torn paper).

It is used to ensure that the paper form is correctly mounted in the printer, both horizontally and vertically. The standard printing test supplied by BULL consists of a line of Es printed on line 1 of the form (just below the fold). You may specify your own printing test by means of the PPRU parameter.

PPRU Parameters

The parameters specified in item PPRU define the format and content of the test printout, as follows:

```
MOVE table-name, PPRU= ( (FORMCTL=form-control,
                          {HEXLINE='hexadecimal-line-description' }
                          {CHARLINE='alphanumeric-line-descriptionn' }
                          .
                          .
                          [ (FORMCTL=form-control,
                              [ {HEXLINE='hexadecimal-line-description' }]}
                              [ {CHARLINE='alphanumeric-line-descriptionn' }]}
                          )
```

FORMCTL: this 2-byte header parameter defines the number of lines the paper is to be moved prior to or following the printing of the test line, as explained in Table D-5.

**EXAMPLES:**

FORMCTL = 0001: Move paper one line before printing test line.

FORMCTL = 8205: Jump to line 5 after printing test line.

□

Table D-5. Test Format Control Bytes (FORMCTL)

Character 1	Character 2	Characters 3 and 4
0 = move paper before printing	0 = move paper n lines as defined in binary by	00 to FF1 (or FORMHT)
	1 = jump to reference level*	00
	2 = jump to line defined in binary by	01 to FORMHT
8 = move paper after printing	3 = jump to channel n, as defined in binary by	01 to 0C
	4 = same as 0 but with FF1 protection	00 to FF1 (or FORMHT)

HEXLINE/CHARLINE:

Describe the line to be printed in hexadecimal or alphanumeric characters, respectively. For both HEXLINE and CHARLINE the character string must be protected. (See Section D.2.4 *Protected Character Strings*).

With respect to CHARLINE, it is possible to introduce an internal code different from EBCDIC.

To define a 2-line alphanumeric printing test in SYS.URCINIT, for example, the user can specify:

```
MOVE AB PPRU= (
    (FORMCTL=0001, CHARLINE='XXXXXX')
    (FORMCTL=0204, CHARLINE='YYYYYY') );
```

A printing test can also be associated with a paper form or with a VFU format. For form:

```
LINK FPRU=0002, PPRU=AB;
or
LINK VPRU=VFU1, PPRU=AB;
```



If a VFU format is already linked to a specific form, the printing test must be linked to the VFU, not to the form.

For editing table:

```
MOVE 0003, EPRU = (FPRU=F002  
                  PPRU=AB);
```

or

```
MOVE BLUE, EPRU = (VFRU = VFU1  
                  PPRU = AB);
```

D.3.12 Parameter Assignment Restrictions

Initially, the following relationships must hold true among the printing parameters:

```
FORMHT    > 0  
HOF       < FORMHT  
FF1       < FORMHT  
If HOF    not = 0, then HOF < FF1
```

Parameters must also be consistent with the printer capabilities as noted in Table D-2. For example, for the PRU 1200/1600 printers

```
If PRDEN=6 then 24 <= FORMHT <= 144  
If PRDEN=8 then 32 <= FORMHT <= 192
```

If the user changes the printing density without changing the form height value, the form-height value must fit the printing density within 1/2 inch.

The margin value may vary in accordance with the type of printer and form used. However, when using this parameter, the user must ensure that the margin plus the number of printed characters does not exceed the number of hammers on the printer (as may be specified in DEVCLASS).

- The absolute stop levels given in vertical format control must not exceed the value of FORMHT.
- The substitution character must be a printable character.
- The line images specified for the print test must be consistent with the character set codes.



D.3.13 Parameter Overriding

The character set code/index table and the form number/editing table, are entered into the system during Device Naming and Allocation, as described above. The following information must be stored in the system with the form number:

- the form height (FORMHT),
- the printing density (PRDEN),
- the margin (MARGIN),
- the full form and head of form levels (FF1, HOF),
- the vertical format tape image (VFU).

A print test may be linked to the VFU information at this level; if no VFU is specified the print test may be directly linked to a form.

The following information may be entered at execution time (e.g., by means of the JCL Statement DEFINE) and if given overrides the values entered by the MEDIA parameter at device naming and allocation.

- form height (FORMHT),
- printing density (PRDEN),
- margin (MARGIN),
- full form and head of form levels (FF1, HOF),
- vertical format with or without linked print test, by redefining each channel (CH1, CH2, etc.),
- print test.

With regard to the print test, the override rules are summarized below:

Table D-6. Print Test Overriding

ORIGINAL VALUES (MEDIA)	PRINT TEST RESULTING FROM DEFINE CH1/CH1
VFU, PRTEST	Print test associated with Form.
VFU, no PRTEST	Standard Print Test.
no VFU, PRTEST	Print Test associated with Form.
no VFU, no PRTEST	Standard Print Test.



D.3.14 ASA Data Format

On the programming level, several internal data formats can be used in preparing the records to be printed out. SSF and SARF formats are treated in other manuals (see especially the *COBOL User Guide*); this section summarizes the American Standards Association (ASA) format. The ASA data format can apply to standard SYSOUT subfiles and to permanent SYSOUT files. Refer to the *JCL User's Guide* for information on the different types of SYSOUT files. The use of ASA format with these SYSOUT facilities is illustrated below.

D.3.14.1 General Programming Notes

In ASA format, each internal record corresponds to a line to be printed out. The first byte in each record is a control character as defined in Table D-5. The control character is not printed, but it must be created by the user.

When preparing ASA records in COBOL, note that SYSOUT and PRINTER must not be specified in the ASSIGN clause. Rather, the specification should be only:

```
ASSIGN TO i fn
```

In the file description, specify

```
WITH ASA
```

In the WRITE verb, the vertical line positioning options must not be used.

D.3.14.2 ASA Format Using Standard SYSOUT Subfiles

Printing or editing parameters can be specified by means of the JCL statement SYSOUT.

For printing out ASA records via a standard SYSOUT subfile the JCL would be:

```
STEP PROGA, . . . ;  
SYSOUT OUTFILE ;  
ENDSTEP ;
```

and the COBOL program would be:

```
SELECT OUT  
ASSIGN TO OUTFILE  
WITH ASA.
```

```
FD OUT
```




```
RECORDS CONTAIN 101 CHARACTERS  
LABEL RECORD IS OMITTED.
```

```
OPEN OUTPUT OUT
```

```
WRITE record-name
```

```
CLOSE OUT
```

D.3.14.3 ASA Format Using Permanent SYSOUT Files

A SYSOUT JCL statement must be specified to enable the file to be printed out.

FOR EXAMPLE:

- JCL statements:

```
STEP PROGA, ...;  
ASSIGN OUTFILE, ASA-FILE, DEVCLASS=MT/T9, MEDIA=TO38;  
SYSOUT OUTFILE;  
ENDSTEP;
```

- COBOL program

```
SELECT OUT  
ASSIGN TO OUTFILE  
WITH ASA.
```

```
FD OUT
```

```
RECORDS CONTAIN 101 CHARACTERS  
LABEL RECORD IS OMITTED.
```

```
OPEN OUTPUT OUT
```

```
WRITE record-name
```

```
CLOSE OUT
```





E. WHEN, EVERY and RRESTART

E.1 WHEN

E.1.1 Absolute Addressing

WHEN or WHEN=IMMED allows the request to be enqueued immediately.

WHEN=date_time allows the request to be enqueued and executed on the given date at the specified time. However, if the time is prior to the current time, the request is cancelled with an error message.

Date_time has the format:

month.day.[yy]yy.hh.mm,

where the separators can be: . or : or /

month can be a number from 1 through 12 or the first 3 letters of the month such as JAN or MAR.

day can be:

- if the month is after the current month and is therefore explicitly specified:
 - either a number from 1 through 31,
 - or LAST-n* where n is a number from 1 through 3 concatenated to the name of the day, being the first 3 letters such as JAN:LAST-1*MON 1992, meaning the last but one Monday of January 1992
- if the month is current and therefore omitted:
 - either a number from 1 through 31 concatenated to *DAY, meaning the nth day of the current month
 - or a number from 1* through 4* concatenated to the name of the day, being the first 3 letters of the day such as 2*TUE, meaning the second Tuesday of the current month



- or LAST concatenated to *DAY, meaning the last day of the current month
- or LAST-n where n is a number from 1 through 15, meaning n days before the end of the current month.

The expression *DAY can also apply to *WKD (working day) or *NWKD (non-working day).

- if the day is relative to today either in the form of TOM (tomorrow) or NXW (next week), in which case the month can either be current or the month following, see E.1.3 *Fast Options*.
- if the day and/or time have lapsed in which case, FORCE enables the request to be immediately enqueued for execution, see paragraph E.1.4 *FORCE Option*.

year can be expressed either as 2 digits such as 91 or 4 digits such as 1991.

hours and **minutes** are expressed as digits in the order indicated.

EXAMPLES:

WHEN=20*DAY:10:00

request to be enqueued and executed on the 20th day of the current month at 10:00.

WHEN=LAST*DAY/7:00

request to be enqueued and executed on the last day of the current month at 7:00.

WHEN=3*MON/18.00

request to be enqueued and executed on the third Monday of the current month at 18:00.

WHEN=LAST-2:10.34

request to be enqueued and executed 2 days before the end of the current month at 10:34.

WHEN=JAN.LAST-1*MON.1992.12:30

request to be enqueued and executed on the last but one Monday of January 1992 at 12:30.

WHEN=12.25.91.6.30

request to be enqueued and executed on December 25, 1991 at 6:30.





E.1.2 Relative Addressing

WHEN=+delay gives the interval before the request is enqueued.

Delay is a signed (+) number of:

- weeks (W or WEEK)
- days (D or DAY)
- hours (H or HOUR)
- minutes (M or MIN).

Delay must be greater than 1 minute. The maximum delay period, specified as a number of up to 5 digits, is 45 days, 12 hours and 15 minutes.

EXAMPLES:

WHEN=+H30M
enqueuing and execution in 1 hour and 30 minutes.

WHEN=+2D
enqueuing and execution in 2 days.

WHEN=+4W2D
enqueuing and execution in 4 weeks and 2 days.



E.1.3 Fast Options

Abbreviations are available as acronyms, namely:

- TOM for tomorrow
- NXD.WKD for next working day
- NXD.NWKD for next non_working day
- NXD for next day
- NXW for next week
- and the first 3 letters of the day such as WED for Wednesday.

The default day for next week is Monday and the default time is 00H00.

EXAMPLES:

WHEN=TOM
the request is enqueued tomorrow at 00H00.

WHEN=NXD.WKD
the request is enqueued next working day at 00H00.



WHEN=NXD.NWKD.10.00
the request is enqueued next non-working day at 10H00.

WHEN=NXD
the request is enqueued next day (tomorrow) at 00H00.

WHEN=NXD/TUE
the request is enqueued next Tuesday at 00H00.

WHEN=NXD.TUE.8.45
the request is enqueued next Tuesday at 8H45.

WHEN=TOM:14:30
the request is enqueued tomorrow at 14H30.

WHEN=NXW
the request is enqueued next week on Monday at 00H00.

WHEN=NXW/TUE
the request is enqueued next week at Tuesday at 00H00.

WHEN=NXW.TUE.8.45
the request is enqueued next week at Tuesday at 8H45.

□

E.1.4 FORCE Option

FORCE is useful when the requested date and/or the requested time have lapsed.

The request is enqueued immediately for execution on the current date and at the current time irrespective of the submission date and/or time **without an error message** being issued.

EXAMPLES:

WHEN=FORCE.10.00 submitted today at 11H00 causes the request to be enqueued today at 11H00 (i.e. immediately, because it is already later than 10H00). There is no error message.

WHEN=10.00 submitted today at 11H00 without FORCE results in the error message *****WHEN: ILLEGAL VALUE** and the WHEN clause being rejected.

□



E.2 EVERY

EVERY specifies the interval and the number of times that the request is to be subsequently enqueued for execution.

The interval expressed in EVERY is equivalent to delay in WHEN, see paragraph E.1.2 *Relative Addressing*. However, the value of interval for EVERY must be greater than 15 minutes but less than 45 days, 12 hours and 15 minutes.

The format of EVERY is:

```
EVERY = ( [frequency] interval [times] ),
```

where

interval is limited to 12 characters and is a number qualified by:

- weeks (W or WEEK)
- days (D or DAY)
- hours (H or HOUR)
- minutes (M or MIN)
- months (MONTH)
- working days (WKD)
- non-working days (NWKD).

times is the number of times up to a maximum of 65535 that the request is to be enqueued. When omitted, the request will continue to be enqueued until the user intervenes or the request is ultimately executed.

If MONTH is specified, the request will be handled on the same day as the submission day next month. If the submission day is 31, the request will be handled on the 1st. of the month following the next. For example, if the submission day is May 31st., the request will be handled on July 1st.

Working days (WKD) are days that belong to neither the list of non-working days nor the list of holidays. The system contains default lists for non-working days and holidays that can be modified using the MODIFY_DJP command.

**EXAMPLES:**

EVERY=5W

requests are to be enqueued indefinitely every 5 weeks.

EVERY=(1D 2)

requests are to be enqueued once every day for 2 days.

EVERY=(20M 72)

requests are to be enqueued 72 times every 20 minutes.

EVERY=(4H 6)

requests are to be enqueued 6 times every 4 hours.

EVERY=1WKD

requests are to be enqueued indefinitely every working day.

WHEN=MAR.31.16.30 EVERY=MONTH

requests will be enqueued at 16H30 on March 31st., May 1st., May 31st., ... until the user intervenes or the requests are ultimately executed.

WHEN=DEC/21/1995/12/45 EVERY=WKD

requests will be enqueued at 12H45 on December 21st, 22nd, 26th, etc. Saturdays, Sundays and Christmas are non-working days or holidays in the default lists.

WHEN=OCT.1*WKD/7.30 EVERY=MONTH

requests will be enqueued each month beginning on the first working day of October at 7H30.

WHEN=NXD.NWKD.8.30 EVERY=NWKD

requests will be enqueued every non-working day beginning on the next working day.

WHEN=JAN.LAST*WKD/00.00 EVERY=MONTH

requests will be enqueued each month on the last working day beginning January at midnight.

□



E.3 RRESTART

RRESTART specifies what to do with the requests which normally should have been handled at a time when the system was stopped for whatever reason and then restarted by WARM RESTART. RRESTART takes into account all system stoppages being normal shutdown, hardware and software failures. Requests whose dates are subsequent to the restarting time are not affected and will be enqueued normally when their time is due.

IMMED: the request is immediately enqueued.

CANCEL or NO: the request is cancelled.

delay: delay is **added** to the normal enqueueing time.
Delay is expressed in

- minutes (M)
- hours (H)
- days (D)
- weeks (W).

Delay can be qualified by E to denote elapsed time.
DelayE specifies the maximum delay allowed for enqueueing the request, on expiration of which the request is cancelled.

Maximum elapsed time for **delayE** is set at 45 days, 12 hours and 15 minutes.

**EXAMPLES:**

```
WHEN=10.30 RRESTART=1D
```

The request is to be enqueued at 10H30 in a day's time. If on the day at 10H30 the system is down, the request will be enqueued tomorrow at the same time. If come tomorrow the system is still down, the request is further enqueued for a subsequent delay and so on, indefinitely.

```
WHEN=10.30 RRESTART=1DE
```

The request is to be enqueued at 10H30 in a day's time. If the system was shut down (for whatever reason) and restarted, one of two actions concerning the request will be taken:

- either the request will be enqueued and executed immediately if the system was restarted before the elapsed time expired
- or the request will be cancelled if the system was restarted after the elapsed time.

□

NOTE:

If the request is repetitive, the delayE must be less than the delay between two normal executions of the request specified by EVERY.



F. Device Types and Classes

Communication controller

Device type: CC

Communication link

Device type: CL

Cartridge

Device type: CT

Device classes:

CT [/LIB] [{ /M5 | M6 | 36T }] [/ { S35 | S75 }] [/C]

where:

LIB	specifies a cartridge library
M5	specifies a device using half-inch cartridges with 18-track recording format
M6	specifies a device using 8mm cartridges or a Galileo Controller.
36T	specifies a device using half-inch cartridges with 36-track recording format
S35	specifies a density of 35Mb / sq. inch
S75	specifies a density of 75Mb / sq. inch
C	indicates compaction

Diagnostic interface

Device type: DI



Foreign controller

Device type: FC
(Transparent peripheral controller attached to PSI-2.)

Device class: FC/M1

Foreign device

Device type: FD
(Transparent peripheral device.)

Line

Device type: LN

Logical terminator

Device type: LT

Mass storage controller

Device type: MC

Device classes:

MC [/ { M1 | M5 }]

where:

M1	is for MS/B10 and MS/D500 disks
M5	is for MS/FSA disks

Maintenance feature

Device type: MF



Mass storage (disk)

Device type: MS

Device classes:

MS [/ { FSA | B10 | D500 }]

Magnetic tape

Device type: MT

Device classes:

MT [/T9] [/ { D6250 | D1600 | }] [/S]

where:

T9	indicates 9-track tape
D6250	indicates density 6250 bpi
D1600	indicates density 1600 bpi
S	indicates streaming mode

On DPS 7000/XTA platform, the magnetic tape devices are seen as CT/M6 devices.

Printer

Device type: PR

Device classes:

PR [/Hnnnn] [/ { SI | FI | A } }] [/ { PR90 | PR54 }]

where:

Hnnnn	indicates nnn print positions
SI	indicates slowest printer
FI	indicates fastest printer
A	indicates any speed
PR90	specifies a dot matrix printer
PR54	specifies a non-dot matrix printer



Tape or cartridge controller

Device type: TC

Device classes:

TC [/ { M3 | M4 | M5 | M6 | M7 }]

where:

M3	specifies a magnetic tape ATP controller (such as MTP4501/02)
M4	specifies a magnetic tape MTC-G controller (such as MTS4501)
M5	specifies a CTS4800 cartridge controller (for CTU4800 drives)
M6	specifies an 8mm cartridge controller or a cartridge controller for CTSD020, CTSD030 or CTSD032
M7	specifies the cartridge controller for CTS6510, CTCC001, or CTCC002 drives.

Unit record controller

Device type: UC

Device classes:

UC [/ { M4 | M51 | M52 | M53 }]

where:

M4	specifies the Service Processor Controller attached to PSI-H, or a multi-purpose controller attached to PSI-H, PSI-K, or PSI-N (CTF4502), or an AUSP emulated peripheral controller attached to PSI-E.
M51	specifies the Service Processor Module (controller attached to PSI-E).
M52	specifies the printer input / output module controller attached to PSI-2
M53	specifies the FDDI input / output logical controller attached to PSI-2, or the networking channel controller attached to PSI-2.



G. How GCOS 7 Handles Dates Before and After the Year 2000

With the upcoming arrival of the year 2000, the following conventions are used in interpreting and entering dates in GCOS 7.

G.1 If the Year Is Entered In Two Digits

If a JCL or GCL command is entered with the year specified in only two digits (YY), GCOS 7 applies the following "61 rule" in determining the first two digits of the year (the century):

- If $YY < 61$, the first two digits of the year are interpreted as 20 (the date has the format 20YY)
- If $YY \geq 61$, the first two digits of the year are interpreted as 19 (the date has the format 19YY)

The number 61 was chosen because GCOS 7 did not exist in 1961, 1961 was not a leap year, and January 1, 1961 was a Monday.

EXAMPLES:

- In files (when format is YY/MM/DD):
EXPDATE = 03/07/11 Date is July 11, 2003
- In journals (when format is YY.MM.DD):
ENDDATE = 02.04.12 Date is April 12, 2002
BEGDATE = 98.01.17 Date is January 17, 1998

□



G.2 Entering YEARS in GCL

In GCL, the year can be specified using two digits *or* four digits at any time. If a year is entered using four digits, GCOS 7 checks that the year is between 1961 and 2060.

For example, to specify the date March 2, 2001 (format: YY/MM/DD), the user can enter:

```
EXPDATE = 01/03/02
```

or

```
EXPDATE = 2001/03/02
```

In the first example, GCOS 7 applies the 61 rule and interprets the year as 2001. In the second example, GCOS 7 checks that the year is between 1961 and 2060.



H. Queued Writer

H.1 Queued Writer Overview

The QUEUED WRITER is a writer that manages the printing of all outputs via an SL library, whereas usual writers can only send outputs to sequential files.

It is started and terminated like any writer by the `START_OUTPUT_WRITER` (SOW or SO) and `TERMINATE_OUTPUT_WRITER` (TOW or TO) commands.

The QUEUED WRITER is automatically activated when a writer is started for a file and this file is found to be a library. The selection and priority rules for the outputs are the same as for any other writer. The members will be created in SSF or ASA format in accordance with the specification in the SO command. The treatment of output banners is the same as for other writers.

The only prerequisite for using the Queued Writer is that the library to which you send outputs must already exist (the Queued Writer will not create it).



H.2 Definitions of Terms

The following terms and abbreviations are used in this Appendix:

SYS.OUT	Resident Linked-Queued file in Virtual Printing Format (VPF).
RON or ron	Run Occurrence Number given by the JOB Manager (JOBM) to each Job: the name "Xron" is assigned to the Job.
outindex	Suffix given to each output generated within a given Job. It is the number "i" in: "Xron:i".
JOB_REP or JOR	Job Occurrence Report: it is an SSF file with the name "Xron:1". It is created by the JOB Manager to report Job execution. Its format is SSF (System Standard Format).
JOB_OUT	Output created automatically by the Output Writer with the name "Xron:2". It contains the outputs created by the Job Xron if their output parameters (class, copy, etc.) are the same as those of the JOR. These outputs are named "Xron:2:j" where "j" is a sub-index. Other outputs will receive a name in the form "Xron:k" with k greater than 2. If the JOB_OUT is empty, it is deleted at the end of the Job.
destination	A destination is attached to each output. A destination is either a station or a compound name of the type "station.station1" (see below). The default destination is the attached station of the Job submitter. Otherwise it is specified by JCL (example: OUTVAL DEST=destination), or by operator commands (example: MO Xron DEST=destination).
station	A station is a logical entity used to group several outputs. For the Output Writer, a station is an output writer queue.
secondary_station	Name given to "station1" when the destination is in the form "station.station1". The secondary_station is used by the Queued Writer to prefix the subfiles it creates. It may be a pseudo station. Its default value is MAIN. It must be 8 characters maximum.



H.3 Activation of the Queued Writer

The Queued Writer is activated by the GCL Command `START_OUTPUT_WRITER` (SO or SOW). The format is:

```
-----  
{START_OUTPUT|SOW|SO}   ON=[$site:]library  
  [SELCLASS=oclist]  
  [DEVCLASS=devclass]  
  [STATION=station]  
  [DATAFORM={SSF|ASA}]  
-----
```

Description of the Main Parameters

site	Specifies the name of the site to which the outputs are sent. The site must not be a remote site.
library	Specifies the name of the SL library to which the outputs are sent. This library must exist (it must already have been created, for instance with the <code>BUILD_LIBRARY</code> GCL command). <code>SHARE=DIR</code> is mandatory.
oclist	Specifies the list of classes on which the writer is started. The default is the class which was specified at configuration time.
DEVCLASS	Specifies the device class and the attributes of the output device. The default is the value which was defined at configuration time.
STATION	Specifies that the outputs the writer will take into account are those present locally in this station queue. If it is not specified, the operator's station is assumed.
DATAFORM	Specifies the format of the output file: SSF (System Standard Format: default) or ASA (American Standard Association format).



The Queued Writer starts when the file to which the outputs are sent is found to be a library. The parameters used in the startup command are checked, but not the actual existence of the library (until the Queued Writer selects an output for it).

When the Queued Writer starts, the following message is sent to the operator:

```
(0U06 SO Wnnn STARTED)
```

```
hh.mn SOW successful: Wnnn started  
on lib: filename  
for station oclist dvc.
```

where Wnnn is the name given to the writer (nn is a number attached to this writer).



H.4 Name of the Subfile to be Printed

The Queued Writer creates a subfile for each output. The names of the subfiles are as follows:

```
<secondary_station>_Xron_outindex_date_hour
```

Where "Xron" (4 characters) and "outindex" (5 characters) are the run occurrence number and the outindex of the output; "date" (5 characters) is the year and day of the creation of the job; "hour" (4 characters: HHMM) is the hour and minute of the creation of the job.

NOTE:

For the JOB_OUT, the sub-index is not used (only the outindex is used). Thus, the Queued Writer creates only one subfile for the JOB_OUT.

<secondary_station> is the name of the secondary station (see above). It may be a pseudo station. This name is taken from the specification of the destination (in JCL, IOF, or by program):

```
DEST=station.secondary_station
```

The default value for the secondary_station is MAIN. It must be 8 characters maximum.



H.5 JCL_CONTROL Subfile

The member JCL_CONTROL contains a user-defined JCL.

If JCL_CONTROL is present in the SL library, the Queued Writer executes it automatically.

NOTE:

The only exception is that JCL_CONTROL is not executed when you use GTWRITER or OPEN 7 on the SL library in which it is located.



IMPORTANT:

The JCL contained in JCL_CONTROL is run after each and every output processed by the Queued Writer.

EXAMPLE:

The following is an example of the user-defined content of JCL_CONTROL.

```
$JOB JCL_CONTROL JOR = ABORT;  
$SEND 'JCL_CONTROL ACTIVATED';  
$ENDJOB;
```





H.6 Termination of the Queued Writer

The Queued Writer is terminated like any writer by the `TERMINATE_OUTPUT_WRITER (TOW, TO)` GCL command. The format is:

```
{TERMINATE_OUTPUT_WRITER|TOW|TO}  ON={wnnn|*}  
  [$site:library]  
  [SELCLASS=oclist]  
  [STATION=station]  
  [STRONG={0|1}]
```

If `STRONG` is not specified, the Queued Writer is stopped when the current output ends.

If `STRONG` is specified, the current output is stopped, the output is set to the "WAIT" state, and may only be reprocessed from the beginning (the question "RESTART FROM?" is not asked).

If "oclist" is used, the writer stops working only on the specified output classes.





Glossary

A

Active (Member, Server)

In the Complex Management Service of the TDS High Availability product, the term ACTIVE SERVER designates the SERVER supporting the functional implementation of a SERVICE and ACTIVE MEMBER designates the MEMBER supporting this ACTIVE SERVER.

ANTESP or ASPA (Ante Service Processor)

An early version of the service processor on the DPS 7000/2x0/3x0.

ARM (Automatic Resource Management)

A GCOS 7 system administration and regulation tool.

Automatic Initialization Mode

Two different parameters exist:

1. The Hardware *Automatic Initialization mode* (called ASPA AUTO in DPS 7000/2x0 and /3x0, SCC AUTO in the DPS 7000/An and AUSP AUTO in the DPS 7000/4xx/5x0/7xx/8x0/Dx0/Mx0.

This is the automatic mode mentioned in Section 2. It is taken into account at the next complete hardware initialization.

2. The GCOS 7 automatic initialization mode, that indicates whether or not a *Reload* should require the intervention of the operator during the IL01 OPTIONS question. This is the automatic mode mentioned in Section 3.

AVR (Automatic Volume Recognition)

Whenever a device is put into the READY state, the system tries to identify the volume. This operation is called Automatic Volume Recognition.



B

Backing Store (BKST)

Due to limitations in main memory size, backing store files are used to contain permanent or temporary information and to temporarily store executing programs not being processed by the system.

Backup (Member, Server)

In the Complex Management Service of the TDS High Availability product, the term BACKUP SERVER designates the SERVER which implements the rescue function of a SERVICE and BACKUP MEMBER designates the MEMBER supporting this BACKUP SERVER.

BFAS (Basic File Access System)

The file access method used for sequential files organized with the Variable Block Organization (VBO).

BIOS (Basic Input/Output System)

A set of program, typically in firmware, that enables a DPS 7000/XTA CPU to communicate with peripheral devices before WINDOWS™ loading

Bi-System

Operating mode on a multiprocessor on which two operating systems can run. Peripherals accessible from the two systems are Shareable even though not described as such in the IRT.

BSR (Basic System Release)

The BSR contains the firmware source components.

C

CDA 7 (Cached Disk Array 7)

A disk unit for DPS 7000/MTxx redundant models.

CDP (Customer Dedicated Processor)

On DPS 7000/MTxx, a CDP can be any co-processor that can be assigned to a function, to enhance its performance. Physically, CDPs can be EPUs, FPUs and GPUs.

See domain, function (privileged), EPU, FPU, GPU.

CMSC (Complex Management Service)

The administration service implementing the TDS high availability features for release GCOS 7-V6 onwards.

CMSR (Complex Management Server)

The server implementing the administrative function on a member of the Complex Management Service.

CNP7 (Communications Network Processor 7)

The network processor that is fully integrated into the cabinet of the DPS 7000, Model A computer



Console or Operator Station Console

The workstation from which the operation of a computer system can be monitored and controlled.

Complex

Term used to designate the set of members implementing the TDS high availability features for release GCOS 7-V6 onwards.

Coupled Systems

Two installations with common peripheral resources.

CRCXGEN (Create Complex Generation)

Description and generation processor for a complex.

Crash

GCOS 7 operating system session abnormal termination.

Crash Reasons

Crash attribute that describes previous operating system session crash originator: System Exception, Call to Panic, Channel Exception, IOP Exception, System Restart, Hardware Failure.

D

Device Identifier (dvnrm)

The device identifier is made up of the 2 letters for the device type and the 2 characters for number (e.g. PR01).

Device Type (dvtyp)

Each device belongs to a device type identified by two letters (e.g., PR = printer). Each device of a device type is identified by a device number (2 digits or letters).

Device Reconfiguration

If the operator mounts a removable volume (typically a cartridge or a magnetic tape) on a device other than the device requested by the system, AVR reads the volume name and directs the job to this new device. This is called, Device Reconfiguration.

Dimension

A set of rules pertaining to memory, CPU and multiprogramming used by the Automatic Resource Manager (ARM)

**Domain**

On DPS 7000/MTxx, a domain represents a kind of CPU resource. A domain exists if the corresponding hardware configuration exists (general purpose CPU, or CDP co-processor). There are four domains:

- Standard
- CDP1
- CDP2
- CDP3

See CDP, function, IPU, EPU, FPU, GPU.

Double Failure

In the Complex Management Service of the TDS High Availability product, the term DOUBLE FAILURE indicates the situation where the two COMPLEX MEMBERS have successively crashed without the CMSRs being able to resynchronize themselves.

DPCV

The standard load module that converts a BKST type system dump output file into a UFAS indexed file ready for access by the DUMP ANALYZER service.

Dump

The operator can either print, display, or store for later printing, the dump (GCOS 7 address space image) of the previous session.

E**EPU (Extended Processor Unit (type 1))**

On DPS 7000/MTxx, a processor for a all privileged functions, corresponding to CDP1.
See CDP, domain, privileged function

F**FBO/VBO (Fixed/Variable Block Organization)**

These are methods for data organization on disks. The fixed mode, which organizes the data in blocks of fixed length, is tending to gradually take the place of the variable mode.

FDA 7 (Fiber Disk Array 7)

A disk subsystem for DPS 7000 models, allows user's disks to be secured with RAID1 mechanism.

FEP (Front End Processor)

Communications processor that manages connections to data networks.

FEPS (Front End Processor Server)

The GCOS 7 communications server that manages a Datanet interface.



FF1 (Full Form 1)

This is a logical page for printing used with the SYS.URCINIT printer control facility.

File Salvaging

Following a crash or any other abnormal termination of the previous session, action is taken to establish the state of files open at the time. Decisions are made about whether these files can be restored, kept or lost.

Form

This refers to a paper page in printer control.

FORMHT

Form height in line numbers.

FPU (Extended Processor Unit (type 2))

On DPS 7000/MTxx, a processor for a privileged function specified by the administrator, corresponding to CDP2.

See CDP, domain, privileged function.

Function (privileged function)

On DPS 7000/MTxx, a function is a set of applications that can be privileged or not. A privileged function can be assigned to a domain. The following functions are privileged:

- SYSTEM
- ORACLE
- OPEN 7
- DW 7
- IQS

See CDP, domain.

G

GAC (General Access Control)

This GCOS 7 Facility enables several users in any of the three processings, Batch, IOF (IQS) and TDS, to access the same resources concurrently.

Galileo

An external peripheral controller which is used to connect on DPS 7000 platforms the Powersave, PowerExchange and Opensave offers.

The GCOS 7 external device type is equivalent to a CT/M6 device.GCL (GCOS 7 Command Language)

The current set of CGOS7 commands providing a user-friendly interface with menus and prompts for the operator.

GCOS 7 (General Comprehensive Operating System 7)

The operating system of the DPS 7000 family of computers.

**GPL (GCOS 7 Programming Language)**

A programming language for systems programmers who develop advanced applications similar to GCOS 7 basic software.

GPU (Extended Processor Unit (type 3))

On DPS 7000/MTxx, a processor for another privileged function specified by the administrator, corresponding to CDP3.

See CDP, domain, FPU, privileged function.

H**HA (High Availability)**

This is a term used in the Complex Management Service of the TDS High Availability product. A SERVICE is said to be in HA mode if it has been mapped on two MEMBERS at COMPLEX generation.

HOF (Head Of Form)

This is the logical starting point for printing.

HRD (High Reliability Disk)

Disk accessed through a sub-system ensuring high reliability.

I**Initialization**

Consists of several phases, which can be divided into two categories:

1. Loading operating systems necessary to proper system operation (GCOS 7, the main operating system, is loaded after SPOS the service operating system. There is also a variety of other modules such as SPI, SIP, or MID, that we are not concerned with here).
2. Testing the correct functioning of the system.

Installation

The largest possible system than can be built on one site. i.e. the total resources than can be controlled from one Service Processor.

IOF (Interactive Operation Facility)

The GCOS 7 versatile time-sharing system that can operate in batch, remote batch and transactional contexts.

IPU (Instruction Processor Unit)

On DPS 7000/MTxx, a general-purpose (or standard) processor. It provides the STANDARD domain resources.

See CDP, domain, FPU, privileged function.



IRT (Installation Resource Table)

A file describing the installation configuration. It is loaded at system initialization time and is used to build the SRST.

On DPS 7000/XTA platforms, the GCOS 7 resources are managed by the V7000 software. IRT is no more used.

ISL (Initial System Load)

The last phase of system initialization. ISL loads the GCOS 7 operating system into main memory.

J

JAS (Journalization Advanced Service)

The GCOS 7 service that guarantees the integrity of a set of user files. In the Complex Management Service of the TDS High Availability product, a JAS SERVICE is SWITCHABLE when there are no MIXED JOBS or non-HA TDS jobs running.

JCG (Job Class Group)

A 'mother' job class associated with one or several 'daughter' job classes, used by ARM.

Job

Work that is processed in a GCOS 7 system. A job is introduced, scheduled, executed and output. It is identified by a run occurrence number (ron) and has a job-class.

Job Checkpoints

In case of error in the processing of a job, one can restart the processing, not completely from the beginning, but from the latest in a series of steps in the process, known as checkpoints.

Job Class

This is a composite object larger than a job. A job-class is modified, executed, held, given priorities, etc.

Job Input Queue

After a job has been introduced to the system and is not HELD, it is placed in a scheduling queue to be selected for execution. Its place in this schedule depends on its scheduling priority, or by default, the priority of its class and its FIFO (First In First Out) position in that class.

Job Outputs

After job execution, the resulting job output is printed or stored on magnetic media.

L

LEVEL (I and II)

In the Complex Management Service of the TDS High Availability product, the term LEVEL refers to different levels of availability offered by the CMSC.

LOS (Level of Simultaneity)

The number of simultaneous data transfers that can be performed on a set of peripheral devices.



M

MAIN Operator

An operator registered in the site catalog with the MAIN attribute.

MAM (Message Access Method) Message Queues

These are queues for communications messages, stacked in order to establish or retrace the events concerning a job run over the network. At Restart, various actions are possible regarding these queues, for example, elimination, or storage for rollback.

Member

A GCOS 7 host system that is integrated into a Complex of the GCOS 7 High Availability facility.

Millennium Application Test Facility

On DPS 7000/MTxx, the ability to test an application with a private date within one GCOS 7.

Millennium Bi-System Test Facility

On DPS 7000/MTxx, the ability to run two versions of GCOS 7 with two different dates for production and test purposes.

MIRROR disks

A pair of disks that have the same name, the same device attributes, the MIRROR attribute, the same volume label, plus the mirror extension, MIRLAB. They are ordered hierarchically Primary and Secondary. The user sees this pair as one logical disk.

Mixed Job

In the Complex Management Service of the TDS High Availability product, the term MIXED JOB designates a batch or an IOF job which uses simultaneously an HA JAS and a non-HA JAS.

MSP (Mass Storage Processor)

Subsystem controller.

MSPD

ORU name for a SCSI drive and streamer controller.

Multiprogramming

Execution of two or more unrelated programs on the same computer.

O

OCL (Operator Command Language)

The older GCOS 7 command set that is still operational, but without the menus and other user-friendly features of the newer GCL.

OPR (Output Priority)

Used to queue outputs waiting to be processed by the writer service job.

**P****POP (Programmed Operator)**

A facility of the DPS 7000 5x0/7xx/8x0/Dx0/Mx0 computers that runs various services without operator intervention, for example: automatic initialization.

PMOS (Programmed Operator Support)

A program, that can be installed locally or at a remote site, which executes many of the routine actions of the main GCOS 7 operator.

Power-on (or Power-up)

The action of applying power to the system. When the *Hardware Initialization* mode is in automatic - the normal situation on all DPS 7000 systems - this is followed by *initialization*, without any operator intervention required.

R**Redundant (System)**

The term REDUNDANT is used in this manual for systems having major components that are in duplicate to prevent failure of the system in case of breakdown of a component.

Reference Level

In printing, this is the paper position when the paper fold coincides with a mark located on the tractors.

Reload

The action of performing again the *Initial Storage Load* after a *System Termination*.

Remote Operator

A correspondent intervening on the system by use of the network.

Restore

If the CONFIG job has been run or a new technical status installed then a RESTORE is mandatory. Equally if segments of the SYS.BKST or SYS.LIB files have been corrupted due to a physical Input/Output error, then this file must be restored from the SYS.SYSTEM file using the FORMAT option.

RLN (Remote Line Handler).

A service of the AUSP Service Processor.

RON (Run Occurrence Number)

GCOS 7 gives a ron identification number to each job in the system. By this number, an operator can identify a job once introduced.

RMOS (Remote Multiplexed Operator Support)

An interactive processor under IOF that enables the monitoring of applications at different sites from a single terminal without having to log-on/log-off.

**Restart**

Cold
Warm
Clean

S**SDA 7 (Small Disk Array 7)**

A RAID1 disk unit which provides 20 GB to 40 GB per cabinet. It can be:

- connected to Bull DPS 7000/MTxx non-redundant systems,
- connected to Bull open systems,
- shared between Bull DPS 7000/MTxx non-redundant systems and Bull open systems.

Segment

In printing, this designates repetitive series of characters on a belt.

Service Disk (on DPS 7000/An)

A disk containing the Service Processor files (for loading SPOS, handling errors, etc.). It is processed by GCOS 7 as a BKST disk and displayed as SRV by the DHW command.

Server

This is a term used in the Complex Management Service of the TDS High Availability product. A SERVER is the result of the projection of a SERVICE on a MEMBER. All the SERVERS implementing the same SERVICE have the same name, which is the name of the corresponding SERVICE.

Service

This is a term used in the Complex Management Service of the TDS High Availability product. A SERVICE is the implementation of a functional component. Its name is declared by the SERVICE administrator. Today, there are three types of SERVICE: JAS, TDS and ORACLE.

SM (Shareable Modules)

These are system modules that can be accessed in the processing of different jobs at the same time.

SMD (Server Management Dialog)

Administrative protocol between the servers of a Complex and the Complex Management Service.

SPI (Service Processor Assy Language Interpreter)

A utility used in the Service Processor of the DPS 7000/2X0/3X0 computers.

Space Line

In printing, this is the minimum space between two printed lines.

SPOS (Service Processor Operating System)

The operating system used by the Service Processor of the DPS 7000/2X0/3X0 computers.



SPR (Scheduling Priority)

Used with job classes to control the order in which jobs are queued and selected for scheduling.

SRST

System Resource and Status Table. At initialization, the System Initialization Program (SIP) updates the **IRT** image, according to the actual state of the configuration, to produce the SRST for that system. The SRST is visible to the Operating System.

On DPS 7000/XTA platforms, the SRST is managed by V7000 software. SIP and IRT are not used.

SSI

Storage Server Interface. This is a CLX process located:

- on the ACSLS server for the DPS 7000/XTA platforms
- or on a local OPEN7 subsystem for all other DPS 7000 platforms.

Stop Level

VFU channel entry/hole.

Substitution Character

Character to be printed instead of a combination that is not present on the belt.

SW (Switchable)

In the Complex Management Service of the TDS High Availability product, a SERVICE is said to be SWITCHABLE if it can be switched from one system to the other.

SYC (System Control)

A Software service which is used to control the DPS 7000 platforms: it allows for example the system initialization, system reload, system restart commands.

System Disk

The disk containing the SYS.BKST (or BKST0) file. This disk is automatically resident, although the RSDT attribute is not displayed by the DHW command.

System Operator Panel(s) (SOP)

The panel(s) used by the operator to control and monitor CSS initialization and operation. They are illustrated and described in Section 2 of the present manual.

System Reset (or Re-initialization)

The action of completely repeating *initialization*, as if a power-off/power-on command had been entered.

System Termination

A system termination can occur in two ways:

Normal termination is requested by the operator using the TSYs command.

Abnormal termination occurs when the system crashes, following an error that cannot be overcome by the GCOS 7 operating system.



T

TA (Twin Architecture)

A DPS 7000 series (DPS 7000/TAxxx) which is the first DPS 7000 series to promote the Twin Architecture (GCOS 7 and Open world on the same server).

Takeover

In the Complex Management Service of the TDS High Availability product, the term TAKEOVER refers to the operation where a SERVER is switched from the BACKUP state to the ACTIVE state.

Terminal

A device connected to the network and accessed by telecommunications software.

U

UFAS (Unified File Access System)

The UFAS Extended access method is versatile in its operation and it supports files with different organizations including data base files. UFAS Extended is the most used access method with GCOS 7.

Used-Service

In the Complex Management Service of the TDS High Availability product, the term USED-SERVICE refers to a SERVICE that is used by another SERVICE.

V

V7000 (Virtual DPS 7000)

A Firmware and software product enabling GCOS 7 programs to run on an Intel™ architecture based server.

This product is the base of the DPS 7000/XTA platforms.

VBO/FBO (Variable/Fixed Block Organization)

These are methods for data organization on disks. The fixed mode, which organizes the data in blocks of fixed length, is tending to gradually take the place of the variable mode.

VFU (Vertical Format Unit)

In printing, this is a set of channels containing predefined stop levels for automatic page printing.

VMM (Virtual Memory Manager)

The GCOS 7 component that manages main memory and backing store to provide the virtual address space visibility.

Volume (file)

Disks, tape reels, and tape cartridges are referred to as volumes. A Volume contains one or more files. Each file is, for the user, a set of information.



Volume Name

Disks, cartridges, and magnetic tape volumes are identified by their Volume Name (volnm). The term volnm is used throughout this manual for convenience, but the volnm may in fact be alphabetic, alphanumeric or numeric.

VOS (V7000 Operation Supervisor)

A V7000 service used on a DPS 7000/XTA platform to manage the server activity.

W

WSP (Wide SCSI Processor)

A controller for wide SCSI disk or cartridge subsystem connections.

X

XPR (Execution Priority)

Controls CPU sharing between executing steps during multiprogramming.

XTA (eXtended Twin Architecture)

A DPS 7000 series which is a major evolution of the DPS 7000/TA platform.

This DPS 7000/XTA platform runs GCOS 7 programs on an Intel™ architecture based server whereas the previous DPS 7000 platforms run GCOS 7 programs on a Bull proprietary CPU chipset .





Index

#

##FW ILxx 3-12

\$

\$JOB 5-22

6

61 rule G-1

A

Access Path (dual) 10-19

Access Path (more than two -) 10-19

Access Path (single) 10-19

AFS (Auxiliary Function Service) 1-5, 1-6, 1-8

After Journal 5-37, 5-38

ALLOCATED 6-4

AMB (Memory Bus) 1-4, 1-5, 1-7

Anomalies B-5

Ante Service Processor (ASPA) 2-15

Any Media Mode 6-12

ARM 1-23, 5-1, 5-53, 7-74, 7-120

 Basic ARM 5-55

 commands 5-59

 Full ARM 5-55, 7-76

 Messages 5-61

ASB (System Bus) 1-4, 1-5, 1-7

ASPA (Ante Service Processor) 2-15

assignment rules(between domain and function) 5-64

asynchronous line 1-27

Attended mode 3-13

Auto Mode, cancelling 3-11

Automatic Reload 3-10

Automatic Resource Management 5-1

Automatic Resource Management (ARM) 1-23

Automatic Resource Manager 5-53

Automatic Restart 3-10

Automatic Volume Recognition (AVR) 1-26

AVR (Automatic Volume Recognition) 1-26, 6-3

 native volumes 6-3

 non-standard 6-3

B

Backing Store 1-22, 3-18, 3-37, 4-8

Backup console 2-20

Basic ARM 5-55

Before Journal 5-35, 5-37

BES 5-62

BFAS access method 3-38

Biprocessor 7-92

BKST 3-18

BKST disk 6-7

BKST file 5-32

BPRU D-28

C

cache 8-81

Cache 7-91, 8-109



- Cache Option 3-46
- Cache Size 3-46
- Cache-Option 3-47
- Cache-Size 3-48
- Cancel Request 2-5
- cartridge tape 6-2, 6-8
- Cartridge Tape Library 6-2, 6-12, 6-15, 6-17, 7-59, 7-180, 8-17, 8-98, 8-137
- catalog 7-15
- CD DOC v
- CDP (Customer Dedicated Processor) 5-62
- Checking
 - status of MFT facility 7-111
- Checkpoints 4-8
- Class group 5-25, 7-34, 7-47, 7-51
- Class load 5-25
- Classes (of devices) F-1
- CLC 7-59, 7-180, 8-17, 8-98, 8-137
- CLC (Cartridge Library Client) 6-11
- clean restart 3-2
- Clock and Maintenance Unit 1-4
- CLS 7-59, 7-180, 8-17, 8-98, 8-137
- CLX (Cartridge Library Unix) 6-12
- CMSR 5-13, 7-132, 8-100, 8-113, 8-134, 8-139, 8-159
- CMU (Clock and Maintenance Unit) 1-4, 1-5, 1-7
- CNP7 4-21
- cold restart 3-2
- Complex 8-100, 8-113, 8-123, 8-133, 8-139
- COMPLEX 7-36, 7-63, 7-109, 7-131
- COMPLEX MGMT C-1
- COMPLXE 8-150
- CONFIG 7-127
- Console Log 4-9
- Control Screen 2-4
- CR (Cancel Request) 2-5
- CRASH COMMENTS 3-24
- CSMR 8-150
- CSS 2-24, 2-25, 10-4, 10-32
- CSS Reconfiguration 10-6

- D**
- data enclosure 5-20
- Datanets 4-21
- DATE 3-20
- date in two digits G-1
- DDP directive 7-83
- Deadlock 8-143
- device 6-2, 7-26
- device class 6-2
- device identifier 6-2
- Device Incidents 6-23
- device sharing (level) 10-21
- device sharing (static, dynamic) 10-21
- device state 6-4
- Device Trace 6-23, 6-27
- device type 6-2
- Device Types and Classes F-1
- Directive, DISPLAY_MFT_FACILITIES 7-111
- DISK 3-21
- Disk Read/Write Head T&D 5-13
- dispatching priorities 8-22
- Dispatching Priority 5-10
- DISPLAY 3-22
- DISPLAY_DUMP directive 7-83
- DISPLAY_HARDWARE (DHW) 10-12, 10-27
- DISPLAY_JAS 5-39
- DISPLAY_MFT_FACILITIES directive 7-111
- DISPLAY_STORAGE_FACILITIES dir 7-134
- Displaying status of MFT facility 7-111
- Distributed Job Processing 5-41, 7-80
- DJP 5-13, 7-80
- DJP facility 5-41
- domain (and function) 5-63
- DP_SAVE B-5
- DPAN 3-25
- DPS 7000/2x0 2-1, 2-9
- DPS 7000/3x0 2-1, 2-9
- DPS 7000/400/Cx0 1-6
- DPS 7000/4xx/Cx0 2-1
- DPS 7000/5x0/7xx/8x0/Dx0/Mx0 1-4, 2-25
- DPS 7000/5x0/7xx/8x0/Dx0/Mx0/MTxx 2-1
- DPS 7000/A 1-3, 2-1, 2-2
- DPS700/4x0/Cx0 2-24
- DSF directive 7-134
- Dump B-5



Binary B-5
DUMP 3-23, 7-83
DUMP_CD 3-11
DV43 question 3-58
dvm A-1
Dynamic Configuration 10-2, 10-30
Dynamic Save 7-85, 8-106, 8-143

E

EJR 7-163
ELECTRONIC DOCUMENTATION v
environment 4-6, 8-79
EPRU D-32
EPU 5-62
EPxx 10-6
Error Logging 6-23, 6-25
EVERY
 arguments of - E-5
 example of - arguments E-6
 purpose E-5
Execution Level 5-56, 8-27
Execution Level Class 5-57
execution priorities 8-46
Execution Priority 5-10

F

FECM 5-13
FEPS 4-21, 5-13
file transfer 7-148, 7-157
filter 4-13, 7-39, 7-45, 8-33
filter set 4-6, 4-16, 4-20, 7-42
FORCE
 WHEN option E-4
FORCE_JOB 7-185
FORCE_OUTPUT 7-187
Form (paper) D-31
FPRU D-32
FPU 5-62
FPxx 10-6
Full ARM 5-55
function (and domain) 5-63

G

GCL (GCOS 7 Command Language) 5-2
GCL Cache 3-46
GCL commands 7-1, 9-3
GCL global variable that stores
 status of MFT 7-111
GCL menus 4-3
generic message 4-14
GMT 7-71
GMT time 3-52
GPU 5-62
GPxx 10-6
GTP 5-13, 5-41
GTWRITER 3-37

H

H_NOCTX domain 7-111
HA 5-13, 7-109, 7-131, 8-100, 8-113, 8-123,
 8-133, 8-139, 8-150, 8-158
Hardware Reconfiguration 7-92, 8-35
HELD 6-4
HELP 3-28
help mechanism 7-12
High Availability 3-27, iv
HOLD state 5-22
HPS 5-55
HRP 5-62

I

IDLE 5-4
IL01 OPTIONS? 1-21
INIT 3-28
Initial Storage Load 1-21
Initial Storage Load (ISL) 3-2
initialization 1-1, 2-1
initialization options 8-58
Input Reader 5-1, 5-16
Input Stream Reader 5-13
Input/Output Cache 8-147
Input/Output Sub-system 1-7
Input/Output Units 1-4
installation (and system) 10-2



Interactive Operation Facility 5-13
 IOF 4-4, 5-2, 7-99, 8-41
 IOSER 5-13
 IOSS 1-7, 2-24, 2-25
 IOU (Input/Output Units) 1-4, 1-5, 1-7
 IPRU D-29
 IPU (Instruction Processor Unit) 5-62
 IPxx 10-6
 IRT (Installation Resource Table) 10-16
 ISL (Initial Storage Load) 3-2
 ISL (Initial System Load) 7-123
 ISL banner 3-22

J

JAS 5-37, 7-101, 8-159
 JAS SERVICES 8-159
 JCL 5-2, 5-13, 5-21
 JCL Translator 5-22
 JCL_CONTROL H-6
 jclass A-1
 jnm A-1
 job accounting 4-1
 job classes 7-30
 Job Control Language 5-2
 Job Execution 5-1, 5-26
 Job Scheduling 5-1
 JOB SCHEDULING 5-24
 Job States 5-2
 Job Termination 5-1, 5-40
 JOB_OUT H-2
 jobset 7-148, 7-163
 Journalization Advanced Service 7-101
 Journalization Advanced Service (JAS) 5-37
 JPPC 5-13
 JRU 5-13
 JTRA 5-13, 5-22

K

KPRU D-29

L

LAEH 5-13

LIB 3-29
 library 7-173
 load module 8-22
 Load Module 5-2, 5-32, 7-30, 7-47
 Local Console 1-3
 LOS (Level of Simultaneity) 10-18

M

Main Distributor Board 2-9
 main operator 1-24, 4-3
 MAINTAIN_VOLUME utility 3-41
 Maintenance Panels 6-24
 Maintenance Station Handler 2-20
 MAM 3-31
 MAM (Message Access Method) 3-15
 MDTIME 3-53
 MEMBER 7-109, 8-123, 8-133, 8-139, 8-150, 8-158
 Memory Bus 1-4
 memory dimensions 8-22
 MENU 9-27
 messages 4-12
 MFT facility
 displaying status of 7-111
 MIRLAB 8-172
 MIRLOG 7-43, 8-170
 Mirror Disks 7-43, 7-113, 8-48, 8-90, 8-170, 8-172, 8-174, iv
 MIRROR_FILE C-3
 MIRROR_MODE C-3
 MIRROR_SHARED C-3
 MIRROR_VOLUME C-3
 MODIFY_HARDWARE (MDHW) 10-12, 10-27
 mono- system and bi-system) 10-2
 MOUNT 6-4
 MPA (Maintenance Processor Adaptor) 1-2
 MPL 5-56
 MSCI (Maintenance Serial Channel Interface) 1-2
 MSH (Maintenance Station Handler) 2-20
 multiprogramming 8-46
 Multiprogramming 1-23, 5-1, 5-6
 Multiprogramming Level 5-24, 5-56
 multiprogramming levels 5-53



MUxx 10-6

N

Network Commands 7-2
Non-removable disks 6-2
non-removable volume 6-3
non-standard 6-6, 6-8
Normal termination 3-60
NSTD 6-6, 6-8

O

OLTD 1-2
OMH (Operator Message Handler) 3-7
On-line Test and Diagnostics 6-23
On-line Tests and Diagnostics 1-2
Operator Message Handler 3-7
OPR 5-12, 5-46
OUT 6-4
outindex H-2
output 8-51, 8-65
Output 5-42
output class 8-20
output classes 8-22
Output Commands 5-45
Output File 5-51
Output Priorities 5-46
output priority 8-20
Output Priority 5-10
Output States 5-44
Output Writer 3-60, 5-1, 5-41, 5-50, 6-20
 automatic JCL H-6
 JCL_CONTROL H-6
 subfile name H-5
OUTPUT_WRITER 8-114, 8-152

P

paper form 6-19, 6-20
Pass Through 9-16, 9-29
Peripheral Processor Incidents 6-23
peripheral processors 6-24
peripheral subsystems 1-2
Permanent Backing Store 1-22

Permanent Virtual Memory Files 4-8
Physical Connection 10-16
POP 2-18
PoWer Control 2-20
POWER OFF 2-23
Power supply sub-system (AFS) 1-8
power timer 8-56
POWER_TIMER 8-155
PPRU D-32
PREMOUNTED 6-4
Print Belt 5-51
Print belts 6-18
Print Test 6-21
Printer Belt D-22
printer incidents 6-22
printers 6-18
PRINTJOR B-24
priority 5-10
Priority
 Execution 5-12
 Output 5-12
PRLOG 6-25, B-3
Programmed Operator Facility 2-18
PSI 1-2
PSP 5-62
PSS 10-4, 10-33
PSS (Peripheral Subsystem) 10-14
PVMF 3-32
PWC (PoWer Control) 2-20

Q

Queue Monitor 5-13
Queued Writer H-1
 termination H-7

R

RAEH 5-13
RBF 5-13
reader 8-40
READY 6-4
RECOV 5-13
RECOVER 6-4
Redundancy 10-22



- reel tape 6-2, 6-8
 - REFILL 5-13
 - REFORMAT 3-34
 - RELOAD_OPTIONS 3-11
 - Remote Batch Facility 5-13
 - Remote Line Handler 2-20
 - Remote Maintenance Service 7-126
 - Remote Maintenance System 8-156
 - Remote Multiplexed Operator Support 4-4, 9-1
 - Remote service console (RSC) 2-20
 - remote terminals 4-1
 - Remote Workstation 2-20
 - removable disk 6-2
 - removable volume 6-3
 - repeat 5-30
 - REPEAT 3-34
 - request 7-26
 - RESIDENT 3-35
 - restart 3-2, 3-9
 - Clean 1-21
 - Cold 1-21
 - Warm 1-21
 - Restart 3-31
 - RESTART 3-37
 - (CLEAN) 3-42
 - restart options 3-8
 - RESTART(COLD) 3-41
 - RESTART(WARM) 3-37
 - Restore 1-21
 - RESTORE) 3-43
 - RL (Reload) 2-16, 2-22
 - RLN (Remote Line Handler) 2-20
 - RMOS 4-4, 9-1, 9-27
 - RMOS (Remote Multiplexed Operator Support) 1-27
 - RMOS session 9-28
 - RMS 8-121, 8-156
 - Rollback 5-37
 - Rollforward 5-37
 - ron 5-2, 5-20, A-1
 - RRESTART
 - arguments of - E-7
 - example of - arguments E-8
 - purpose E-7
 - RSC (Remote Service Console) 2-20
 - RSMT disk 6-6
 - Run Occurrence Number 5-2, 5-27
- ## S
- SADMOPT 7-127
 - SARF D-41
 - scheduling 8-46
 - Scheduling
 - Priority 5-10
 - SCU (System Controller Unit) 1-4, 1-5, 1-7
 - SECOPT 7-127
 - Server shutdown 8-143
 - SERVER2 5-13
 - SERVICE 8-123, 8-158
 - Service commands
 - Reload 2-16, 2-22
 - Restart 2-17, 2-22
 - service disk 6-7
 - Service jobs 5-15, 5-34
 - Service Processor; 2-18
 - Sharable modules 4-8
 - SHARED 3-44
 - SHARED disks 6-7
 - SHUTDOWN 3-61
 - SITE.CATALOG 7-127
 - SOP 2-18, 2-24, 2-25, 6-24
 - SOP (System Operator Panel) 1-3
 - SOP Switch 2-9
 - specific message 4-14
 - SPOOL 3-46
 - SPR 5-11
 - SPV (SuPerVisor) 2-20
 - SR (Restart) 2-17, 2-22
 - SRV disk 6-7
 - SSF D-41
 - STANDBY 6-4, 6-15
 - START_OUTPUT H-3
 - state of a device 10-23
 - Static Configuration 10-1, 10-30
 - statistics 4-1
 - step 5-2, 5-26
 - Step Execution 5-27
 - Storage Load 3-58
 - Sub-systems, Input/Output 1-7
 - SuPerVisor 2-20



suspended job 5-34
SVLOGF B-23
SWITCH_MIRROR 8-132
SWMIR 8-132
synchronous terminal 1-27
Syntax Notation v
SYS.BKST 1-22, 3-18, 3-43
SYS.ERLOG 6-25, 8-96, 8-103, 8-107
SYS.LIB 1-22
SYS.LOGC 8-140
SYS.OUT 5-33, 5-52
SYS.PVMF 1-22
SYS.SPDUMP file B-5
SYS.TVMF 1-22
SYS.URCINIT D-1
SYSOUT File D-19
SYSQM 3-49
 (BACKUP) 3-49
 (CLEAN) 3-49
 (WARM) 3-49
system (and installation) 10-2
system (mono- and bi-) 10-2
System Administrator 5-54
System anomalies B-5
System Backing Store 1-22
System Bus 1-4
System Controller Unit 1-4
System Crash B-5
SYSTEM disk 6-6, 6-24
system load 7-107
System Operator Panel (SOP) 1-3
system performance 4-1
System Ready 3-58
system shutdown 8-162
System termination 3-63
System Trace 5-13

T

TAKEOVER 8-133
tape 6-2
TDS 5-13, 5-37
TDS Commands 7-3
Temporary Backing Store 1-22

Terminal Writer 7-27, 7-143, 7-145, 7-152,
8-8, 8-65, 8-67, 8-130, 8-131, 8-
166, 8-168
TERMINATE_OUTPUT_WRITER H-7
Time Deviation 7-71
TIMEDEV 3-13, 3-21, 3-50, 3-53
Timer 7-119
TNS 4-21, 5-13
trace 7-141, 8-63, 8-127, 8-164
TRC 3-51
TRCCL 5-13
TVMF 3-51
Types (of devices) F-1

U

UD (Use Device) 2-5
UFAS access method 3-38
Unit Record Controller D-1
URINIT D-1
 Commands D-4
 Statement D-3
 Utility D-3
Use Device 2-5
User Jobs 5-1
usernm A-1

V

VCAM 5-13, 9-3, 9-37
Virtual Communications Access Method 9-3
VMF 4-8
VMF backing store 3-32, 3-51
VMM 5-56
VOLCHECK utility 3-41
volume 6-2
volume name 6-2
VPRU D-32

W

warm restart 3-2
Warm Restart 3-31
WHEN
 absolute addressing E-1



example of - E-2
example of - arguments with EVERY E-6
fast options E-3
FORCE option of - E-4
purpose of - E-1
relative addressing E-3
windowing method (61 rule) G-1
WORK tape 6-8
WRITE PERMIT 6-8, 6-16, 8-102
WRITE PROTECT 6-16

X

X-HRP 5-62
XL 5-56, 8-27
XL Class 8-77
XLC 5-57
XPR 5-12

Y

year 2000 G-1

