

click on the coin and win \$20,000



The Map Editing FAQ for Duke Nukem 3D Version 2.0

This FAQ Is Authored and Copyright © 1999 By: Jonah Bishop
Please See Section 0.11 For Licensing Information

Sections in the table of contents below that appear in **boldface type**, have been either updated or added since the last version of this FAQ.

Section 0 - Map FAQ Information

- 0.1 Author Information
- 0.2 Contacting the Author
- 0.3 Included Files
- 0.4 Revision History**
- 0.5 Credits and Acknowledgements**
- 0.6 Where to Download this FAQ
- 0.7 Printing Information
- 0.8 The Online Version
- 0.9 Map FAQ Central
- 0.10 FAQ Errors and Reporting Them
- 0.11 Licensing Agreement
- 0.12 Where are all the questions?

Section 1 - Information About Maps

- 1.1 What is a map?
- 1.2 What can I do to maps?
- 1.3 Can I look at the game maps?
- 1.4 What do I use to make maps?
- 1.5 Can I make maps for other games?
- 1.6 How do I run user maps?
- 1.7 What exactly is the Build engine?

Section 2 - Things You Should Know Before Starting

- 2.1 Introduction
- 2.2 2D and 3D Modes
- 2.3 Hitags and Lotags
- 2.4 Placing Sprites
- 2.5 The Special Sprites
 - 2.5.1 Sector Effectors

- 2.5.2 Activator
- 2.5.3 Touchplate
- 2.5.4 ActivatorLocked
- 2.5.5 Music & SFX
- 2.5.6 Locator
- 2.5.7 Cyclor
- 2.5.8 Masterswitch
- 2.5.9 Respawn
- 2.5.10 GPSpeed

Section 3 - Your First Map

- 3.1 Installing and Setting up Build
- 3.2 Creating a Room
 - 3.2.1 Adding Textures
 - 3.2.2 Adding Sprites
 - 3.2.3 Adding a Nested Sector
 - 3.2.4 Raising and Lowering Sectors
 - 3.2.5 Parallaxing Skies
 - 3.2.6 Sloped Surfaces
 - 3.2.7 Creating a Circle
- 3.3 The Art of Shading
 - 3.3.1 Shading Techniques
- 3.4 Advanced Features
 - 3.4.1 Splitting Walls
 - 3.4.2 Resizing and Moving Sectors
 - 3.4.3 Joining Sectors
 - 3.4.4 Deleting Sectors
 - 3.4.5 Copying and Pasting Sectors
- 3.5 Completing this Map

Section 4 - Making All The Effects

4.0 Introduction

- 4.0.1 Introduction

4.1 Doors

- 4.1.1 Doom Type Doors
- 4.1.2 Adding Sounds to Doors
- 4.1.3 Reversed Doom Type Doors
- 4.1.4 Splitting Doom Doors
- 4.1.5 Swinging Doors
- 4.1.6 Swinging Glass Doors
- 4.1.7 Problems and Fixes with Swinging Doors

- 4.1.8 Sliding Star Trek Doors
- 4.1.9 Splitting Star Trek Doors
- 4.1.10 Teeth Doors
- 4.1.11 Another Way To Do Teeth Doors
- 4.1.12 Sliding Doors
- 4.1.13 See Through Sliding Doors
- 4.1.14 Rotating Subway Doors
- 4.1.15 Making Doors Auto Close
- 4.1.16 Making Locked Doors

(See sections 4.11.1 - 4.11.4 for switch operated doors)

4.2 Parallaxed Skies

- 4.2.1 Correctly Parallaxed Skies
- 4.2.2 Problems And Fixes With Parallaxing

4.3 Lighting Effects

- 4.3.1 Making Lights Blink
- 4.3.2 Making Lights Blink When Shot
- 4.3.3 Open Door & Room Lights Up
- 4.3.4 Light Switches
- 4.3.5 Colored Lights
- 4.3.6 Pulsating Lights

4.4 Moving Sectors

- 4.4.1 Rotating Sectors (Gears)
- 4.4.2 Using Rotating Sectors for Doors
- 4.4.3 Two-Way Trains
- 4.4.4 One Car Subways
- 4.4.5 Making Multi-Car Subways
- 4.4.6 Notes On Subways

4.5 General Sector Effects

- 4.5.1 Earthquakes
- 4.5.2 Rooms Over Rooms
- 4.5.3 Stretching Sectors
- 4.5.4 Conveyor Belts & Water Current
- 4.5.5 Making Curtains Draw Back
- 4.5.6 The Pool Table
- 4.5.7 Fish Tanks

4.6 Water

- 4.6.1 Submergible Water
- 4.6.2 Sloped Water Sectors

- 4.6.3 Swimming in Slime
- 4.6.4 Making Waves
- 4.6.5 The Rising Water Effect
- 4.6.6 Level Over Level Water Trick
- 4.6.7 Timed Underwater Airlocks (Complex)
- 4.6.8 Rising Water on Sloped Surfaces (Complex)

4.7 Walls and Wall Effects

- 4.7.1 Masked Walls
- 4.7.2 One-Way Walls
- 4.7.3 Forcefields and Controlling Them
- 4.7.4 Wall Information

4.8 Mirrors

- 4.8.1 Mirrors
- 4.8.2 Problems and Solutions with Mirrors
- 4.8.3 Making the Two Way Mirror

4.9 Teleporters

- 4.9.1 Teleporters
- 4.9.2 Sewer Holes & Silent Teleporters
- 4.9.3 One Way Teleporters (Version 1.4 or 1.5 Required)
- 4.9.4 Version 1.3D One Way Teleporters

4.10 Ending the Level

- 4.10.1 Nuke Buttons
- 4.10.2 Ending Sectors
- 4.10.3 Ending Sectors with Messages

4.11 Switch Effects

- 4.11.1 Switch Activated Doors
- 4.11.2 Combination Door Switches
- 4.11.3 Another Version of Combo. Switches (Version 1.4 or 1.5 Required)
- 4.11.4 Multi-Door Switches

4.12 C-9 Effects

- 4.12.1 C-9 Canisters (Visible)
- 4.12.2 Switch Activated Explosions
- 4.12.3 Touchplate Activated Explosions
- 4.12.4 Cracked Walls
- 4.12.5 Fixing The 'Texture Problem'
- 4.12.6 Making the Ceiling Blow Up (And Enemies Fall From the Sky!)

4.13 Sectors That Rise and Fall

- 4.13.1 Lifts

- 4.13.2 Switch Operated Lifts
- 4.13.3 Locked Lifts
- 4.13.4 Elevator Transports
- 4.13.5 Constructing Destructible Buildings
- 4.13.6 Crushers (Engine Pistons)
- 4.13.7 Making the Ceiling Fall Upon an Explosion
- 4.13.8 Rotating & Rising Sectors
- 4.13.9 Making Ceilings and Floors Drop Down
- 4.13.10 Incremental Rising and Falling Sectors (V. 1.4 or 1.5 Required)

4.14 Music and Sound Effects

- 4.14.1 Types of Sound Effects
- 4.14.2 Adding Your Own Music and SFX
- 4.14.3 Sector Activated Sounds
- 4.14.4 Making Phones Ring

4.15 Working with Sprites

- 4.15.1 Working With Sprites
- 4.15.2 Security Cameras
- 4.15.3 Vents
- 4.15.4 "3D" Bridges
- 4.15.5 Demo Cameras
- 4.15.6 Spawning Stuff
- 4.15.7 Active Tripbombs
- 4.15.8 Making the Mini-Boss
- 4.15.9 Exploding Helicopter & Car
- 4.15.10 Cranes

4.16 Multiplayer Level Making

- 4.16.1 Dukematch & Co-op Starting Positions
- 4.16.2 Making Items Appear Only In Multiplayer
- 4.16.3 Making Duke TAG Levels (Version 1.4 or 1.5 Required)

4.17 Miscellaneous

- 4.17.1 Glass
- 4.17.2 Secret Areas
- 4.17.3 Shooters
- 4.17.4 Turning Shooters Off
- 4.17.5 Difficulty Settings
- 4.17.6 Signs That Fall When Shot
- 4.17.7 Lightning and Thunder (Version 1.4 or 1.5 Required)
- 4.17.8 Making Rain
- 4.17.9 Forcing Duke To Start Weaponless

Section 5 - General Information

- 5.1 Palette List
- 5.2 Facts On Dimensions
- 5.3 Sector Effector Lotag Listing
- 5.4 Sector Lotag Listing
- 5.5 List of Official Levels
- 5.6 List of Cheats
- 5.7 List of Secret Messages in the Game

Section 6 - The Most Common Questions and Answers

- 6.1 I can't find Build on my CD - where is it?
- 6.2 I can't get Build to work. How do I do it?
- 6.3 How do I enter 3D mode?
- 6.4 3D Mode is screwed up! What do I do to fix it?
- 6.5 How can I play my own maps?
- 6.6 When I run my level, I get an error. What does it mean?
- 6.7 Why does the game crash when I run my level?
- 6.8 Build sucks. Is there another editor available?
- 6.9 Walking under a parallaxed sky kills me. Why is this?
- 6.10 My subway shoots rockets at me - Why?

Section 7 - Common Build and Runtime Errors

- 7.1 Found Lonely Sector Effector
- 7.2 Subway Found No Zero'd Sectors With Locators
- 7.3 Can't Make a Sector Out There

Section 8 - Utilities for Duke Nukem 3D

- 8.1 Wad2Map
- 8.2 Wad2Art
- 8.3 Kextract
- 8.4 Kgroup
- 8.5 Group File Studio

Section 9 - Finishing Up

- 9.1 Map Authoring Template
- 9.2 Uploading Your Level



Click on me

and win **\$20,000**



Section 0 - Map FAQ Information

[Previous Section](#)

[Table of Contents](#)

[Next Section](#)

0.1 Author Information

My name is Jonah Bishop and I spend most of my time doing something computer related (gaming is the largest activity). I enjoy editing of any kind, and have always had a great interest in Duke Nukem 3D and Build. At the time of this writing, I am a freshman at North Carolina State University. In addition to this FAQ, I maintain the Map Editing FAQ for Shadow Warrior, the Con Editing FAQ for Duke Nukem 3D, and Group File Studio.

0.2 Contacting the Author

Being a student brings the pressures of school and homework, and it unfortunately brings the fact that I don't have much time for answering tons of Build questions. If you do need to get in touch with me, however, just [click here](#). I can't promise a reply, but I will certainly do my best to help you in your building career.

0.3 Included Files

Files below that appear in **bold type** have either been added or updated since the last release of this FAQ.

- **MPFQDN20.ZIP** - The main FAQ zip file
- **MPFQDN20.TXT** - The FAQ itself (this file)
- **SECT4.TXT** - Section 4 (the how-tos) by itself

The other files that should be included in the zip file are as follows:

- **ADVANCED.MAP** - Contains more difficult tutorials, step-by-step
- AIRLOCK.MAP - Example map showing how to create underwater airlocks
- **ATOMIC.MAP** - Contains atomic edition specific how-to examples
- **CREDITS.TXT** - FAQ Credits and Acknowledgements
- **LICENSE.TXT** - License Agreement for this FAQ
- **MIRRORS.MAP** - Mirrors tutorial map
- **README.1ST** - Important information about this FAQ
- **RHISTORY.TXT** - FAQ revision history
- **ROTATEDR.MAP** - Example map showing how to create complex rotating doors
- **SLOPEWTR.MAP** - Example map showing how to create rising water on slopes
- **STHRUSLD.MAP** - Example map for see through sliding doors
- **STHRUSLD.TXT** - Information on see through sliding doors
- **SWLIFT.MAP** - Example map showing how to create switch operated lifts
- **TDOOR2.MAP** - Example map for second way to create teeth doors
- **TDOOR2.TXT** - Information on the second way to create teeth doors
- **TEMPLATE.TXT** - User map authoring template

- **TUTORIAL.MAP** - Contains all examples for all effects

0.4 Revision History

The revision history for this FAQ resides in its own file, which can be found in the zip file. The name for this file is rhistory.txt. [Click here](#) to view an online version of this file.

0.5 Credits and Acknowledgements

The credits and acknowledgements for this FAQ resides in its own file, which can be found in the zip file. The name for this file is credits.txt. [Click here](#) to view an online version of this file.

0.6 Where to Download this FAQ

The latest version of this Map Editing FAQ will always be located at either Dukeworld or Map FAQ Central. The URLs for these two sites are as follows:

- [Map FAQ Central](#)
- [Dukeworld](#)

0.7 Printing Information

If you want to print this FAQ, it is recommended to set your page margins to the following values (all units are in inches):

- Top : 1.00
- Bottom : 1.00
- Right : 1.00
- Left : 1.00

It is also recommended that you set the font for this entire FAQ to Courier New. This ensures that the FAQ looks the way it was intended. Consult your text editor's help to see how to change page margins and font (it's quite easy to do).

0.8 The Online Version

An online edition of this FAQ is available at both Map FAQ Central and Dukeworld. [Click here](#) to go to the Table of Contents.

0.9 Map FAQ Central

The Map Editing FAQ has an official home, and you can visit it at the following URL:

<http://mapfaq.3dportal.com>

Map FAQ Central is a large and ever growing site, with links to user created files, tons of cool sites on the web, and more Build information than you can shake a stick at! A list of the best ever levels is available, for both Shadow Warrior and Duke Nukem 3D. Be sure to tell your friends about it too!

0.10 FAQ Errors and Reporting Them

Sometimes I'll goof up and make a mistake while writing this FAQ, and other times I just get the information all wrong. If you find an error in this FAQ, or if you have a suggestion to make this FAQ better, please [click here](#).

You will definitely be credited for finding any error (no matter how small) and I thank you in advance for

letting me know about them.

0.11 Licensing Agreement

A special license has been issued with this FAQ, in order to prevent the misuse and misdistribution of this FAQ. I work hard on this FAQ (I love doing it to) and I don't want people stealing the work that I've put into this guide (you wouldn't want that to happen either, right?). So please take a moment to read through it, and follow the guidelines it sets. Thanks!

[Click here](#) to view this license.

0.12 Where are all the questions?

FAQ is supposed to stand for frequently asked questions, right? Yes, it is supposed to stand for that term. But where are all the questions in this FAQ? Well, over the course of the development of this guide, the questions have turned into headings for information (take a look at the table of contents to see). I have turned away from the basic question and answer format in favor of the current format in use. The *most* frequently asked questions, however, can be found in section 6 of this FAQ.

Copyright © 1999 by Jonah Bishop. All rights reserved.



Click on me

and win **\$20,000**



Duke Nukem 3D Map Editing FAQ Revision History

[Table of Contents](#)

[Back to Section 0](#)

Version 0.1

FAQ Created. Everything is new.

Version 0.2

A *lot* of stuff added & some errors corrected. Thanks to all the kind people that pointed them out! Added about 7 new things, many other updates. Added section 5: Finishing Up.

Version 0.3

Again, a *lot* of stuff added. Now over 50 (53 to be exact) how-tos, not including some of the things that were put into a different category. Did some reorganizing. Added a Questions and Answers section and a Info and Other Stuff section. Remade some of the maps to be step-by-step maps.

Version 1.0

The author changes. Many errors are corrected. Remodeled tutorial maps to be easier to understand (basically changed the textures to something better looking than the Brown-block texture). Added a section on Utilities for Duke, added more how-tos, and added a part about Build. The FAQ is now much easier to read (as far as the steps of some of the how-tos go).

Version 1.1

More corrections to errors found. Some sections added, including a list of keys for Build, both in 2D and 3D modes. Neat Tricks and Tips moved into Section 5. Reorganized section 3.

Version 1.2

Fixed several errors. A few sections added, on wall information, correctly parallaxed skies, working with sprites, etc. Changed some information to make it more up to date. Build key list is alphabetized and the sound list is revised a bit. Added a list of secrets to the ZIP file.

Version 1.3

HUGE reorganization. Made everything easier to find, and rewrote large chunks of the FAQ to make it easier to understand. Cleaned up a lot of sloppy how-tos, added a list of Editart keys, added information on Duke TAG, and added some brand new how-tos.

Version 1.31

Minor Fixes. Changed the Conveyor Belt How-To, and CONVEYOR.MAP to fix a bug. Removed Section 0.3.1. Several other changes in section 0.

Version 1.4

Fixed the Sliding Star Trek Doors Tutorial Map. The doors in the map did not work at all, but it has now been

corrected. Updated the URL to the HTML version of this FAQ. Fixed an error in the Making Phones Ring how-to.

Version 1.5

Basically the same revisions as version 1.6, but never formally released to the public. It was released to a few people, however.

Version 1.6

Added the Licensing Information. A few new how-tos were added, spelling errors were corrected, and a few things were revised. The tutorial maps were condensed into their own ZIP file within the main ZIP file.

Version 1.7

Changed all URLs pointing to Map FAQ Central due to the change. Changed a few copyright dates, updated license.txt, and other minor changes. Fixed some spelling errors.

Version 1.8

Added the how-to on making Duke start without any weapons.

Version 1.9

Fixed several errors in the FAQ. Added a new version of the Build key list to the ZIP file. Also added a Rich-Text Format version of the Build key list.

Version 2.0

Largest FAQ release yet. The complete FAQ was rewritten, fixing too many problems to mention.

Copyright © 1999 by Jonah Bishop. All rights reserved.



Duke Nukem 3D Map Editing FAQ Credits and Acknowledgements

[Table of Contents](#)

[Back to Section 0](#)

There are several people that made contributions and reported errors, so I would like to thank them for doing so. This list is in alphabetical order by last name (or nickname).

3D Realms

Made one of the very best games ever - And thanks for giving us an editor so quickly!

Chris Allcock

Let me know about the error in the turning shooters off section.

Aphelion

Helped with numerous things. Submitted the swinging glass doors how-to idea.

Andrew Ball

Sent in the information concerning the "found lonely sector effector" error message.

Chris Biberstein

Pointed out what a palette value of 5 does. Also sent in the information about making Duke start without weapon and ammo.

Mats Blixt

Made the incredible sound effects list and allowed me to put it in this FAQ.

Captain Massive

Found *numerous* errors in the FAQ, and made several suggestions.

Martin Doms

Suggested adding the circle feature to section 3 (which I did).

Todd Faris

Found the sound tag bug in the sector tag list.

Brett Gmoser

Started this FAQ in the first place!

Gavin Lambert

Found more errors in version 1.9 than can be listed. Thanks a bunch!

Levelord

Made some of the cool levels in Duke Nukem 3D, and let me know what the secret messages meant.

Kevin Marcoux

Pointed out an error with the [old] conveyor belt how-to and map.

Jaap Menist

Helped with swinging doors and countless other things. Wrote the section on the second way to do teeth-doors and see through sliding doors.

Michael

Found the [old] sliding star trek door error.

MqsTout

Pointed out some errors in the FAQ and gave me some fixes for them.

P.E. Nilsson

Made the dimensioning facts.

Ben Roffelsen

Creator of the "level over level water trick".

R.P.G.

Made a telephone sound suggestion and made a level over level suggestion.

SemiCharm

Found the "All Pointers Checked" key for the Build key listing.

Tinus

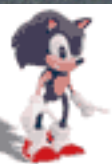
Submitted the process for entering negative shade values in the Atomic Edition of Build.

Gijs de Vries

Showed me how to make the underwater airlock and how to make water rise on sloped surfaces.

Tom Welsome

Gave me the right information on turning shooters off.



Section 1 - Information About Maps

[Previous Section](#)

[Table of Contents](#)

[Next Section](#)

1.1 What is a map?

A "map" is a general term used to describe the levels in Duke Nukem 3D (or just about any other game for that matter). Maps include all the level information (effect parameters, wall placement, sprite placement, etc.).

1.2 What can I do to maps?

Unlike Doom (which made use of wad a pwad file formats), maps do not contain the art textures, enemy code, sounds, or other such objects in the game. Maps are solely the layout of a level, the effects used in that level, etc. Enemy coding, sounds, and art textures can be found in the .grp and .con files.

1.3 Can I look at the game maps?

Yes. The file "kextract" is used to extract map files into your game directory, where they then can be viewed. Take a look at section 8 of this FAQ to see how to use this program.

You also can view the maps without extracting them. At the command prompt in DOS, type in the following:

`BUILD ExLyy.map`

The letter x (after the letter E) is the episode number to look at, and the letters yy (after the letter L) is the level number to look at. For example, to view episode 2, level 4, I would type: `BUILD E2L4.map`

1.4 What do I use to make maps?

The only editor available for creating your own maps is the Build program, which was written by 3D Realms. This editor was the editor used to create the levels found in Duke Nukem 3D, so you can do everything that the official level designers did in their levels. Build is only available on the game CD, and can be found under the \goodies\build directory on the CD.

1.5 Can I make maps for other games?

Yes. The Build engine that comes with Duke Nukem 3D, was made specifically for Duke Nukem 3D, but it can now be used to make levels for NAM.

1.6 How do I run user maps?

Running user maps that you create and/or download is as easy as using a short command line parameter. Type in either of the following lines in DOS:

`duke3d -map foo.map`

`duke3d -map foo`

In the lines above, foo is the name of the map that you wish to run. The .map extension is not needed, but can be used.

1.7 What exactly is the Build engine?

The Build engine is the graphics engine that games such as Duke Nukem 3D, Shadow Warrior, Blood, and NAM run on. Although it isn't truly 3-D, it does give a nice feel for the environment that has been presented, and it gives the player a never before seen level of interactivity in this environment.

Copyright © 1999 by Jonah Bishop. All rights reserved.

click on the coin and win \$20,000



Section 2 - Things You Should Know Before Starting

[Previous Section](#)

[Table of Contents](#)

[Next Section](#)

2.1 Introduction

There are quite a few things that you will need to know prior to making your first level. Read this section completely, and you will have everything I feel that you need to know to start making levels!

2.2 2D and 3D Modes

There are two different editing modes in Build. The 2D mode, which shows the layout of the map, sprites, and other goodies, and the 3D mode, which looks like the game itself. To switch between these two modes, press the [ENTER] key on the numberpad. The regular [ENTER] key does not switch modes, so this must be noted!

In 2D mode, you lay out the map itself. Here you draw walls, place sprites, change tags of walls, sprites, and floors, and save your work. In 3D mode, you change the textures of the walls, change how high or low the floor, ceiling, sprites, and other objects are, and you change the shade here as well.

2.3 Hitags And Lotags

This topic is complicated, yet simple at the same time. Tags can be given to walls, sprites, or sectors, and they tell the wall, sprite, or sector what to do. There are two numbers that any of these objects can be given. The first number is called a hitag. Hitags usually identify the object, so that other objects can be paired with it (I'll give an example in one moment). Lotags are generally a fixed number that tells the game exactly what to do.

Here is an example:

I make two sectors, and I want them to be teleporters (i.e. - I step into one sector, and teleport to the other sector). I place a sector effector sprite in both sectors, and give them tags to identify them. I first give *both* sprites a lotag of 7. This number is preset, and it tells the game that when the player enters this sector, he is to "teleport" to the other teleporting sector. I then give *both* sprites the *same* hitag. This number must be unique, as it gives an identity to both sprites. Here is what the program is probably thinking when I'm playing this level:

Hmmm. The player has just entered a sector with a special sector effector sprite in it (lotag of 7). Let me see if I can find another teleporting sector with the same identity (unique hitag), so that I can "teleport" him there. Hey - There's the other sector! WOO-HOO!

All the hitags do are link various effect sprites together. If you only gave one sprite a hitag, the program would not be able to find another, matching teleporting sector, and then would not be able to teleport you anywhere, because it can't find this sprite's "mate". Here are the key strokes that you need to do in order to place these tags (in 2D mode):

- Sector Hitag : Press [H] in the desired sector
- Sector Lotag : Press [T] in the desired sector
- Sprite Hitag : Press [ALT]+[H] on the desired sprite
- Sprite Lotag : Press [ALT]+[T] on the desired sprite

- Wall Hitag : Press [ALT]+[H] on the desired wall
- Wall Lotag : Press [ALT]+[T] on the desired wall

These same tag assignments can be made in 3D mode as well. Here are the key strokes for use in 3D mode:

- Sector Hitag : Press [']+[H] in the desired sector
- Sector Lotag : Press [']+[T] in the desired sector
- Sprite Hitag : Press [']+[H] on the desired sprite
- Sprite Lotag : Press [']+[T] on the desired sprite
- Wall Hitag : Press [']+[H] on the desired wall
- Wall Lotag : Press [']+[T] on the desired wall

NOTE: To set wall tags in 2D mode, it is advised to turn off the grid, so that you are not selecting sprites. Press [G] several times to turn the grid off.

2.4 Placing Sprites

NOTE: When I say that you should press the [V] key (or any other key for that matter) on a sprite, I mean that you should move the cursor over that sprite with the mouse, and then press the key.

Without sprites, levels in Duke Nukem 3D would be nothing. Sprites make up all the "eye candy" in the level and are the root of all effects in Duke 3D. To place a sprite in the level, press the [S] key in either 2D or 3D mode. It is advised to place them in 2D mode, as you have the grid as a guide for placement.

There are several different modes sprites can be in (3 of them to be exact). Sprites can lie flat on the floor, be "pasted" to the wall, or they can be in the "normal" mode (guns and other goodies are usually in this mode). Normal mode is the default setting, and to switch the current mode of a sprite, enter 3D mode, and press the [R] key on it. It will go from normal to being pasted on the wall (according to which direction the sprite is facing). Pressing [R] on it again will paste it to the floor, and again will make it normal.

NOTE: If you place a sprite on a wall in 3D mode (using the [S] key), the sprite will automatically be "pasted" flat to the wall that you placed it on.

NOTE: Only "square" sprites can be made to lie flat on the floor. Sprites that fit this condition are ones that have the same x-axis height as the y-axis height.

Sprites can face in different directions as well. In 2D mode, you will notice a "tail" coming out of the circle which represents the sprite. This tail points in the direction that the sprite is currently facing, and it also denotes the "front" side of the sprite. To change this, press the [,] or [.] keys on the sprite in 3D mode.

To change the texture of a sprite (changing what the sprite actually is), enter 3D mode and press [V] on it. This will bring up a screen of all the current sprites used in the level you currently are working on (This is useful as you start to need the same sprite over and over). Press [V] again, and you will proceed to the tile selection screen. Here, you can press [PGUP] and [PGDN] to scroll through the many textures used in Duke. When you have found a suitable texture, press [ENTER].

To raise or lower the sprite, press the [PGUP] or [PGDN] keys on the sprite in 3D mode. This changes the altitude of the sprite.

Here is an example of placing a sprite in a room:

I want to place a calendar on the wall in my room. In 2D mode, I would press the [S] key ON THE WALL where I want the sprite to be, and I would then enter 3D mode, pressing the [V] key on it twice. I would select the calendar sprite, raise it up to the height I want it on my wall (with the [PGUP] and [PGDN] keys) and then press [R] on it *once*. But now that I look at it, the sprite is turned to the side, or it is backwards!

You must remember to set the sprite direction with the [,] and [.] keys. The tail should face *away from the wall, so that it lies flat on the wall. Check out some of the levels that came with the game if you don't understand.

2.5 The Special Sprites

There are a few special sprites that need to be explained. The most important ones are as follows:

2.5.1 Sector Effectors

Sprite #1 - The sector effector sprite is the most used sprite in the entire game. It does just about what the name says: effects a sector. These sprite are used to create game effects and manipulate sector attributes. These sprites are the main tool used to create all effects in Duke Nukem 3D. Place the sprite in the sector to effect. The lotags for these are preset, and a list of the lotags for this sprite can be found in section 5.4.

2.5.2 Activators

Sprite #2 - Activators are used to activate a sector effector (SE) sprite when a switch or touchplate is triggered. They also can activate sector lotags.

2.5.3 Touchplates

Sprite #3 - The touchplate will activate an activator or a masterswitch sprite, when the player walks over the sector that contains it. A hitag of 0 means that the touchplate will activate every time the player walks on top of the sector. A hitag of greater than 0 means that the touchplate will only be activated that many times (whatever number you put on there). The only number I've ever seen used that is greater than 0, is 1 (to activate an effect only once). But use the value that you want to.

2.5.4 ActivatorLocked

Sprite #4 - This sprite locks the sectors lotag function, so that it cannot take effect until the player unlocks it with a switch or touchplate. This is used to make locked doors and other such effects.

2.5.5 Music&SFX

Sprite #5 - This sprite is used to create all the sound effects in the game. It can be used in 3 ways:

- 1. Activation Sound
- 2. Ambient Sound
- 3. Echo Effect

We will go into this sprite in much more detail later on.

2.5.6 Locator

Sprite #6 - These are used to define a path that moving sectors and certain enemies (the Pigcop Recon Cars) will follow. Both moving sectors and the Pigcop Recon Cars cannot be used in the same map. They are also used as the endpoints of a two-way train.

2.5.7 Cycler

Sprite #7 - This sprite is used to make the sector floor, walls, and ceiling pulsate in brightness. Section 4.20 goes into great detail about this sprite.

2.5.8 Masterswitch

Sprite #8 - This sprite is the exact same as the activator, but it has the ability to use a time delay. The hitag

is the time delay that the effect should wait to be carried out. I think that a value of 32 is one second.

2.5.9 Respawn

Sprite #9 - This sprite is used to spawn enemies or other items after activated by a touchplate or switch. An example of this is entering a room, and a bad guy spawns behind you.

2.5.10 GPSPEED

Sprite #10 - This sprite is used to define rates of effects. The lotag is the only thing that is needed here, and the value for it is how fast the effect should go (i.e. - smaller means slower). These cannot be used on all effects, but they can be used on some.

Copyright © 1999 by Jonah Bishop. All rights reserved.



Section 3 - Your First Map

[Previous Section](#)

[Table of Contents](#)

[Next Section](#)

3.1 Installing and Setting up Build

Build is located on your Duke Nukem 3D CD, under the /goodies/build directory (if this directory isn't on your CD, consult section 6 of this FAQ for more information on why this is). This program can be installed from the main game installer, so that should be the place to install from. Simply run the install program, choose *only* to install Build, and install it into your game directory (i.e. - the Build program and all associated files should be in the same directory as the game itself). Not installing this program into the game directory can cause some major problems, so this is where to install it.

After the program has been installed, enter into your game directory and run the BSETUP program. Within this program are several menus where you can change your settings for sound cards, video modes, and more. **YOU SHOULD ONLY CHANGE THE VIDEO MODE!** Changing other items in this program will (for some strange reason) cause this setup program to not save your settings. So change the video mode to the mode you want to run (make sure your system can run in that mode before setting it that way), and select "Save Settings & Exit". Build does not use music and sound effects at all, so setting these items isn't needed anyway.

Now, simply type "build" at the DOS prompt (without the quotation marks of course) and you will be launched into the program!

3.2 Creating a Room

Throughout this section, I will walk you through making your very first Duke Nukem 3D level. Build is an extremely fun editing program, and some cool stuff can be done in this editor, so let's get started.

NOTE: The user map TUTORIAL.MAP (which can be found in the zip directory that this FAQ came in), is an example map that I have created, and which includes everything covered in this section. If you want to see exactly what something looks like as you go along in this section, just take a look at this map in Build.

NOTE: Save early and *often*! I cannot stress this enough – I have lost tons of work before by forgetting to save my work. Pressing the [ESC] key in 2D mode allows you to save your work by pressing the [S] key (saves it as the current filename (the default is newboard.map)) or by pressing the [A] key (saves it as a new filename, which you specify).

The first thing that we'll do is create a square room. Press the [G] key several times, until the largest grid size in Build appears. Zoom out a little bit (by using the [Z] key). Move your mouse cursor near one of the vertices, and press the [SPACEBAR]. What you are doing is beginning to draw the walls of your level. Move your mouse cursor up 3 grid blocks and press the [SPACEBAR] again. You have just drawn your first wall! Move to the right 3 blocks, press [SPACE] again, move down 3 blocks, press [SPACE] again, and finally move the cursor back to the first vertex you drew, pressing [SPACE] one last time. Your room should now be 3 grid blocks by 3 grid blocks.

3.2.1 Adding Textures

Textures are added in 3D mode. Press the [ENTER] key on the number pad (it's found on the far right on your keyboard). You should now be in 3D mode, which looks exactly like the game. Cool, huh? The current texture, however, is quite ugly, and one that you (hopefully) do not want to use. So let's change the texture

of the walls, the ceiling, and the floor.

Move the mouse cursor over the floor. Press the [V] key once. This screen (which currently has that ugly texture on it) is a listing of all the textures that appear on either the floor or ceiling of your level (This screen becomes quite handy later on when you want to use a texture already used in your level). Now press the [V] key again. You should be looking at a large screen of sprites, textures, and other items. Press the [PGDN] key several times, or use the arrow keys to scroll around the screen. There are literally thousands of tiles in Duke Nukem 3D, so choose one that you want to use as your floor, and press the [ENTER] key (not the one on the numberpad). Your floor is now the texture you wanted it to be! Now do the exact same thing for the ceiling.

When you have that done, move the mouse cursor over one of the walls in your level. Press the [V] key once. The screen you are now looking at is much like the first one for the floor and ceiling, but this time it shows all the wall textures used in your level. Press [V] again, and select a good texture for the wall. Your texture appears, but it doesn't look like it's all there! You need to scale the texture by using the [4], [8], [6], and [2] keys on the numberpad. Experiment a little, and get it to look like you want. Pressing the [5] key, in addition to the keys listed above, will make the scale of the texture change in larger increments (making it faster to do).

You could do this for each wall, but that would take quite some time to do in a big level. So move the mouse cursor over the wall that you have just changed (after you have it like you want it) and press the [TAB] key (in version 1.3D of the game, you need to press this key twice - if you don't, the small white text that appears on the screen will stay there). Now move the mouse cursor over one of the other walls. Press the [ENTER] key (not the one on the numberpad). You have just copied and pasted the texture, its scale factor, and its shade value! This makes texturing a room much quicker, and it is quite a handy feature. Do this for each wall. (**NOTE:** Sometimes, the scale isn't pasted 100% correctly, so minor adjustment might be needed.)

3.2.2 Adding Sprites

Sprites are what levels in Duke Nukem 3D revolve around. Without sprites, this game would have no meaning at all. Sprites consist of enemies, weapons, ammo, sector effectors, and all kinds of eye candy. Placing sprites is quite easy and can be done in either 2D or 3D mode. Placement in 2D mode is *much* easier, because you have the grid to line up sprites with other objects.

Let's add a shotgun sprite to our level. In 2D mode, move the mouse cursor into one of the corners in the room. Press the [S] key. You have just placed a sprite. What you now see is a circle with a "tail" coming out of it. The tail indicates the direction of the sprite. This is not important in the case of ammunition, health, enemies, weapons, and some eye candy, but some objects that you might want to place on a wall need the angle to be just right. If the sprite you just placed is currently pink, it means that this sprite will block your movement in the game. If it is pink, move your mouse cursor over the sprite you just placed and press the [B] key. This toggles sprites between "blockable" and "not blockable". No sprites *need* to be blockable, but it comes in use when placing a table or chair sprite and you want it to block the movement of the player. Now enter into 3D mode. You should see some sort of sprite (probably that ugly texture again). Move the mouse cursor over the newly created sprite, and press the [V] key once. Like the textures, this first screen shows you what sprites are used in your level. Press the [V] key again. Select tile #28 (which is the shotgun sprite). You have now placed a shotgun in your level! You might want to size this sprite down a bit (using the [4], [8], [6], and [2] keys on the numberpad), but weapons, health, ammo, enemies, and inventory items all size themselves in the game.

Now let's add a poster type sprite to our level. In 2D mode, move your mouse cursor over one of the walls of your room (the cursor should be on the side of the wall that the player will see). Press the [S] key. Now change the angle of this sprite to face straight away from the wall. This is done using the [,] and the [.] keys (also known as the [<] and [>] keys). When you have the angle properly set, enter 3D mode. Change this sprite to #500. We want this sprite to stick to the wall, but it doesn't seem to be doing that. Move the mouse cursor over the sprite in 3D mode, and press the [R] key once. This makes the current sprite lie in a flat plane, perpendicular to the sprite's angle. So if the sprite is facing to the right (or east), the sprite would lie flat in the north-south plane (the sprite would still face east). This is kind of hard to explain, but play around

with it to see what I mean. Change the angle of the sprite after the [R] key has been pressed, and you will see how the flat position and the angle relate. Size your sprite to your liking.

Let's add one more sprite to our level for now. Place one sprite in 2D mode somewhere in the middle of your room. Change this sprite to #568. This is the magazine sprite. Back in 2D mode, change the angle of this sprite to be 45 degrees or something of that nature (press the [<] or [>] key twice to do this). Now enter 3D mode again. This time, move the mouse cursor over the sprite and press the [R] key *twice*. This command makes sprites lie flat on the floor. Pressing [R] again returns the sprite to its default setting. Size this sprite to your liking. NOTE: Not all sprites can be made to lie flat on the floor. If the sprite you want to make lie flat on the floor looks really weird (or messed up) when you try to place it flat on the floor, it simply cannot be made to be in that position. If I am not mistaken, the only sprites that can lie flat on the floor are square sprites (their x-axis size is equal to their y-axis size).

3.2.3 Adding a Nested Sector

Now we need to add what is called a "nested" sector. This term simply means that you have a sector within a sector. We will first create a small box sector in our room. Somewhere in your room, begin drawing a new sector. Make sure that the grid size is smaller than the largest setting, and draw a square sector again. If you enter 3D mode at this point, you will notice that the sector you just drew has appeared as new walls, taking a chunk out of your room. This is not what we want, so enter back into 2D mode. Move your mouse cursor inside this newly created sector, and press the [ALT]+[S] keys. Your sector walls should have just turned red! Now enter back into 3D mode again. Looks like normal huh? Well, it's not quite like it was before. You have just created a nested sector, which will now raise up from the floor.

3.2.4 Raising and Lowering Sectors

In 3D mode, move your cursor over to the general area of your newly created nested sector (point the cursor at the sector floor). Now press the [PGUP] key several times. You are now raising your sector out of the ground! Raise it to a decent height (but don't raise it all the way to the ceiling), and put some good textures on it (maybe the crate texture or something of that sort). Using the [PGUP] key raises either the floor or ceiling of a sector, and using the [PGDN] key lowers either the floor or ceiling of a sector. This is how you will set your sector heights in the game, and will most likely become one of the most used commands in Build. These two commands can also be used to raise and lower sprites. Remember that poster-type sprite we added to our level? Press the [PGUP] key on it several times. You have just raised the sprite up on the wall. Very useful and very easy, is it not?

3.2.5 Parallaxing Skies

In 3D mode, move the mouse cursor over the ceiling of the room sector. Now change the texture of this sector's ceiling to #89. This is the Los Angeles texture, and for it to look right, needs to be parallaxed. Press the [P] key. Do this for the ceiling of the nested sector we created as well. Several special cases with parallaxing (and several problems) are documented in the parallax sub-section of section 4.

3.2.6 Sloped Surfaces

Let's now create a sector off to one side of our room. To do this, we first need to insert some vertices. Pick the wall that you want to have the sector on, and move your mouse cursor over that wall. Press the [INSERT] key in the places on this wall where you want the vertices for our new sector to be (you should only need 2). Move the mouse cursor over one of these newly created vertices and press the [SPACEBAR], starting the drawing process. Now move the mouse cursor over the second vertex that you inserted, pressing [SPACE] again. The move out away from the room sector, drawing your sector as normal. When you finish drawing your sector, the part of the wall between the two vertices you inserted should change into a red line. Enter 3D mode and take a look at the newly created sector. Let's make the floor of this sector sloped.

By using the [and] keys, slope the floor of this sector downwards (or upwards, whichever you prefer – just experiment with it). Keep in mind that when sloping sectors, you need to set the "first wall" of the sector. The

"first wall" is the wall that the slope will pivot around (think of it as a hinge point). This can be set in either 2D or 3D mode by pressing the [ALT]+[F] key combination on the wall you want to be the first wall of the sector (I personally find it easier to do this in 3D mode). Experiment with setting this parameter, and see how it changes the tilting angle of the sector (try this on our newly created sector).

3.2.7 Creating a Circle

This is a feature that is (unfortunately) not used enough in level design, and it can be a little tricky to get just right.

Somewhere in your level (do this outside of your room sectors), draw a triangular sector. Move your cursor over one of the walls of this sector, and press the [C] key. Now move the mouse around. You will see the beginnings of a circle, but we need to manipulate this circle to get it the way we want it to be. Pressing the [-] and [+] keys (on the numberpad) decreases and increases the number of points on the circle, respectively. Pressing the [SPACEBAR] draws the circle, while pressing the [C] key cancels the drawing of the circle. Once you've got half of the circle drawn, move the vertex of the triangle (that you initially created) that is sticking out away from the half-circle, over one of the vertices that runs across the half-way point of the circle. This is quite hard to explain in words, so take a look at the tutorial map (I've placed several iterations of this circular construction on the map, so you can see how this is done step by step). Then create the other half of the circle.

3.3 The Art of Shading

One of the key elements in any level made is the shading technique. Shading is a delicate art that must be learned and used **everywhere** in the level. Leaving your level's walls, sprites, floors, and ceilings set with a shade value of 0 makes most level reviewers shun (or even skip) your level entirely. There are several shading techniques that you must learn to use, and they're not hard to learn at all! Once you learn these techniques, you will be able to make some very cool looking levels!

3.3.1 Shading Techniques

Before we learn about the different techniques needed to make shading what it needs to (and should) be, we first need to learn how to shade. All shading is done in 3D mode. Pressing the [+] key (on the numberpad) makes the object under the mouse pointer get brighter (the shade value decreases). Pressing the [-] key (on the numberpad) makes the object under the mouse pointer get darker (the shade value increases). For small shading amounts, this can be handy, but to make things dark, pressing these keys over and over can be a pain. So there's a shorter way to do this! Point at the object you want to shade and press the [=][S] key combination. Then enter the value you want the shade to be. The larger the number, the darker the shade. A value of about 50 is almost completely dark, but you can go all the way up to 128. Negative values are allowed (down to -128) and make things bright (but not noticeably lighter than a shade value of 0). Entering negative values in version 1.3D of the game is quite a pain. It appears that it can only be done by pressing the [+] key (on the numberpad) multiple times. In versions 1.4 and 1.5, you simply type the [-] key (not on the numberpad) before typing the value.

The first shading technique that you need to learn is that you should offset the shade of walls of a room. To do this, select two opposite directions on your map (i.e. - north and south) and make **all** walls in that direction one shade value (let's use a value of 10 for now). Now you should give **all** the walls in the opposite two directions (i.e. - east and west) a shade somewhat lighter or darker than the previous shading (let's use a value of 15 for now). This allows your eye to find the corner of the room much faster, and it gives a much more realistic look to any room (if you make joined walls the same shade, finding the corner can be quite a difficult thing to do). This technique should be used in **every** room and passageway in your level, and it should remain consistent (i.e. - if you made all east-west walls the lighter value, then all east-west walls should remain lighter in the future). If you have a angled wall (i.e. - it doesn't run perfectly north-south or east-west) use a value somewhere between your east-west and north-south values. It is a good idea to set this type of shading before setting the next type.

The second technique that you need to learn is that light travels in a straight line. The light from a ceiling

light should make straight shadows when passing past a corner of a hallway or other object. Remember to light your areas according to the positioning of your lights. This is the second type of shading you should set.

The third technique is ambient shading. Although light travels in a straight line, there is some ambience. Use multiple sectors for shading, making the sectors farther from the light source incrementally darker. Although this type of shading isn't needed, it definitely makes for some cool looking effects (gradual shading effects make some scenes look excellent – especially at a high resolution).

3.4 Advanced Features

Build has a few features that can be very useful to a learning Builder, and make the Building process *much* faster to complete.

3.4.1 Splitting Walls

Splitting walls is quite easy to do. In 2D mode, simply point at the place on the wall where you would like to insert a vertex and press the [INS] key. That's all you have to do! Easy, huh?

3.4.2 Resizing and Moving Sectors

Another nice feature of Build is the ability to resize and move sectors. To resize a sector, simply move each vertex to the place you wish them to be. To select multiple vertices to move at the same time, hold the [RIGHT SHIFT] key down, and move the box (you'll see it appear when you move the mouse) over the desired vertices. Now moving one of these vertices will also move the other selected vertices! To move a sector, hold down the [RIGHT ALT] key and, again, drag the mouse over the desired sector. Now, release the [RIGHT ALT] key and move the sector with the mouse. Press the [RIGHT ALT] key one last time to deselect the sector once you have successfully moved it.

3.4.3 Joining Sectors

Joining sectors is a useful feature when you don't want a sector to be split up into individual (red lined) sectors. In 2D mode, simply press the [J] key in the first sector you wish to join (this is the sector from which the properties (height, shade, etc.) will be taken). Then press the [J] key again on the second, adjacent sector. There you go - No more red line!

3.4.4 Deleting Sectors

Deleting sectors is also an easy thing to do. Simply move the mouse into the sector you wish to delete, and press the [CTRL]+[DEL] keys to delete the sector. Be warned that there is no undo feature in Build, so make sure you know *exactly* what you are doing before using this command. A quick save of the level before using this command wouldn't hurt.

3.4.5 Copying and Pasting Sectors

Copying sectors can make Building go very fast. You can copy sectors to the current level, or even to another level! To copy sectors, hold down the [RIGHT ALT] key in 2D mode and move the box (you'll see it appear when you move the mouse) over the desired sectors. Then release the [RIGHT ALT] key. Once the sectors are selected (they will flash green), click and hold the left mouse button (be careful not to move the selected sectors) and press the [INSERT] key. Keeping the left mouse button down, move this set of sectors (you will see that they have been copied) to the desired location in your level. Press the [RIGHT ALT] key again to deselect them.

To copy sectors from one level to another, simply select the desired sectors (as stated above) and press the [INSERT] key (again, just like before). Now, while the sectors are still selected (they should still be flashing green), load another level. The selected sectors should appear in your level, and you can move them to the desired location. Pressing the [RIGHT ALT] key will then deselect the sectors.

3.5 Completing This Map

You are now through the basic parts of editing. The next section goes into detail on making every effect known in Duke Nukem 3D (there are even several effects not used in the game itself that people have invented). To run your map, simply type the following at the command prompt:

```
duke3d -map [mapname].map
```

[mapname] is the name of the map you have just created. Run your map and see the fruits of your labor – Welcome to the world of Build!

Copyright © 1999 by Jonah Bishop. All rights reserved.



check out...

Gran Turismo 2

PlayStation

epworld.com



4.0 Introduction

[Previous Section](#)

[Table of Contents](#)

[Next Section](#)

4.0.1 Introduction

This section of the FAQ covers every known effect available to the map designer. Due to the number of effects available, this section is quite lengthy. Example .map files are used throughout this section, so take a look at any of the how-tos for the corresponding map name.

Copyright © 1999 by Jonah Bishop. All rights reserved.

4.1 Doors

[Previous Section](#)[Table of Contents](#)[Next Section](#)

4.1.1 Doom Type Doors

Map Name: TUTORIAL.MAP (ROOM 1-A)

This how-to explains how to make one of the most common doors found in the game: "Doom" doors. These types of doors rise from the floor into the ceiling, and are quite easy to make.

STEP 1: Draw your door sector in 2D mode. Move the pointer over the door sector, and give the door sector a lotag of 20 (press the [T] key when the mouse pointer is inside the door sector).

STEP 2: Enter into 3D mode and give the door frame (the walls of the door sector) some nice textures. Move the pointer over each wall and press the [O] (the letter, not the number zero) key. This will prevent the walls from "moving" with the door as it travels (don't press the [O] key and see what happens - it's very unrealistic).

STEP 3: Now enter into 3-D mode, and lower the ceiling of the door sector all the way to the floor. Then texture the door itself.

4.1.2 Adding Sounds to Doors

Map Name: TUTORIAL.MAP (ROOMS 1-A through 1-I)

To add sound effects to a door (the sound of the door opening and closing) follow these simple steps:

STEP 1: Place one music and sound effect sprite (#5) in the door sector.

STEP 2: Give the music and sound effects sprite a lotag value equal to the sound you wish to play. A listing of sound effect numbers can be found in the zip file that this FAQ came in. Values of 166 through 168 are good values, but use what you want (anything other than a door sound isn't very realistic however).

4.1.3 Reversed Doom Type Doors

Map Name: TUTORIAL.MAP (ROOM 1-B)

This how-to explains how to make the reversed version of the Doom Type door. This type of door lowers from the ceiling into the floor, and is quite easy to make.

STEP 1: Draw your door sector in 2D mode. Move the pointer over the door sector, and give the door sector a lotag of 21 (press the [T] key when the mouse pointer is inside the door sector).

STEP 2: Now enter into 3-D mode, and raise the floor of the door sector all the way to the ceiling. Then texture the door itself.

4.1.4 Splitting Doom Doors

Map Name: TUTORIAL.MAP (ROOM 1-C)

This type of door splits in the middle horizontally, with the top half rising to the ceiling and the lower half

lowering to the floor. These doors are constructed exactly as regular Doom-Type doors, with the following changes:

1. The sector lotag should be 22.
2. The floor and ceiling of the door sector should meet in the middle of the door itself.

NOTE: The height of the door should be an even value, to ensure that the top or bottom half does not need to travel farther than it should (which can cause some problems). The value you should use is the number of times that you must press the PGUP key to raise the ceiling from floor height to the height you want.

4.1.5 Swinging Doors

Map Name: TUTORIAL.MAP (ROOM 1-D)

Swinging doors are possibly the coolest and most useful doors in Duke Nukem 3D, but they can be a pain to construct. Follow these steps, however, and you should be able to make a working swinging door.

STEP 1: Create the room and the hallway that the door will separate. The ceiling height of the hallway should be the height of the door. The hallway and the room should be two separate sectors (there should be a red line between these two sectors in 2D mode).

STEP 2: Create the door sector on the side of the red line that the door will swing in to. This sector should be valid player space. In 3D mode, raise the floor of this sector to the height that you want the door to be (or you can lower the ceiling; whichever you prefer).

NOTE 1: The door sector should be made within another, valid sector.

STEP 3: Now move the door sector to where it should be. One side of the door should be in the exact position as the red line separating the hallway from the room (i.e. - two of the vertices of the door should sit exactly on top of the two vertices of the hallway and room junction).

NOTE 2: The door should be in the sector in which it will swing into. If it is not, some big problems can occur.

STEP 4: Give the door sector a sector lotag value of 23.

STEP 5: Place one sector effector sprite in the door. Give this sprite a lotag value of 11, and a unique hitag value. Now you need to move the sector effector at the pivot point for the door. Turn off the grid lock feature by pressing the [L] key in 2D mode. Move the sector effector sprite as close to the pivoting corner of the door as you can, but **do not** place this sprite on the corner itself. Zoom in as far as you can possibly go in 2D mode to see just where you are putting it.

STEP 6: Now you need to set the angle for the door. If you want the door to swing in a clockwise direction, point the sector effector up. If you want the door to swing in a counter-clockwise direction, point the sector effector down.

STEP 7: Place one Music&SFX sprite in the door. Give this sprite a lotag value of 165. You are now finished! Take a look at the tutorial map if you don't quite understand.

NOTE 3: Swinging doors will not work correctly in any of the following situations:

- If the floor or ceiling is sloped.
- If the door swings into another sector.
- If the door is not a valid player space sector.

4.1.6 Swinging Glass Doors

Map Name: TUTORIAL.MAP (ROOM 1-E)

Swinging glass doors look great and add a nice touch to virtually any level. They aren't easy, however, so follow these instructions very carefully.

STEP 1: Create a swinging door (see the swinging door how-to) but do not raise the door sector up from the floor.

STEP 2: At the end of the door farthest from the pivot point (or farthest from the sector effector with lotag 11 - same thing) create a small square sector inside of the door. This sector should ***NOT*** be valid player space (i.e. - the walls of this sector should be white).

STEP 3: Mask the outermost red line of the door itself, changing the mask texture to the glass texture. The [M] key in 3D mode creates a masked wall. Press the [B] and [H] keys on this mask. A point can be inserted on this line (before masking it) if desired, in order to prevent the glass texture from being as long as the door.

STEP 4: Lower the ceiling of the door sector a few [PGDN] units (2 works nicely) and raise the floor of the door sector a few [PGUP] units (again, 2 works nicely). You're done! Take a look at the tutorial map if you don't quite understand.

4.1.7 Problems and Fixes with Swinging Doors

Map Name: NONE

The following is a list of common problems and their solutions for working with swinging doors.

- PROBLEM 1: My door disappears when it opens! What do I do?
- SOLUTION 1: The problem here is that a corner of your door is actually going ***outside*** of your map (into nothingness) and is causing your door to disappear. Either move the sector effector sprite with the lotag of 11 closer to the vertex of the door, or angle the corner that travels outside the map. Also make sure that you are not building a door that swings in sectors of different height or sloped sectors.
- PROBLEM 2: My door swings in the wrong direction! How do I fix this?
- SOLUTION 2: You need to fix the direction that the sector-effector is pointing. To make the door swing clockwise, the sector effector needs to point upwards. To make it swing counter clockwise, you need to point the SE down.
- PROBLEM 3: When I make my door, I try to move it and it loses it's unique sector lotag/hitag.
- SOLUTION 3: The 'door' sector has to be created inside the sector where it will swing into. Do not create it and then move it into another sector or the it will lose it's unique lotag/hitag.
- PROBLEM 4: When I open my door, the walls move with it! How do I fix this?
- SOLUTION 4: When you create the 'door' sector, it should be a free standing sector. In other words the vertices should NOT be connected to any other vertices. AFTER the door sector has been made, you may move it (to the door hole) and now the vertices may overlap because they will not connect now. Be sure of this or the walls will be pulled along with the door!
- PROBLEM 5: When I open my swinging door, the door kills me! How can I stop this from happening?
- SOLUTION 5: Make sure that the sector effector sprite is actually INSIDE of the sector, and NOT on a line of the sector.
- PROBLEM 6: When I open one swinging door, a completely different door opens up. What's the deal?
- SOLUTION 6: The hitags for your sector effector sprites are the same. You must use UNIQUE hitags for doors that you want to open separately.

4.1.8 Sliding Star Trek Doors

Map Name: ADVANCED.MAP (ROOM 1-A)

Sliding Star Trek doors are difficult to make, but they are a cool addition to most any level. Because this type of door is so complex to make, a step by step construction has been placed in the tutorial map. Take a look at it to see what to do, and even copy the sectors if you want.

NOTE 1: The texture of the door will be "squished" when the door is opened. There is no way around this problem.

NOTE 2: The door sound will sound twice for some reason. Again, there is no way around this problem.

4.1.9 Splitting Star Trek Doors

Map Name: ADVANCED.MAP (ROOM 1-B)

This is a more complex version of the sliding star trek door, and it is by far one of the coolest doors in the game.

STEP 1: Create a sliding star trek door.

STEP 2: Raise the floor of this door about halfway up, and lower the ceiling about halfway down (the ceiling and floor should meet in the middle of the door).

STEP 3: Change the door sector lotag value from 9 to a value of 26. You're all done! Take a look at the tutorial example if you don't quite understand.

4.1.10 Teeth Doors

Map Name: TUTORIAL.MAP (ROOM 1-F)

This door never appears in the game (it has a few minor problems) but it is very cool anyways.

STEP 1: Create a sector for the door, and two square sectors inside the door sector (for the teeth of the door). All these sectors should be valid player space.

STEP 2: Lower the teeth sector floors down until you can no longer see the floor (even when looking down at an angle).

STEP 3: Give the door sector a sector lotag value of 29 and a unique sector hitag value.

STEP 4: Place one sector effect in each tooth sector. Give each sector effector sprite a sprite lotag value of 22. Also give each sector effector sprite a sprite hitag value equal to the door sector hitag value. You're done! Look at the tutorial map if you don't quite understand.

NOTE 1: This door must start in the open position for it to properly work.

4.1.11 Another Way To Do Teeth Doors

Map Name: TDOOR2.MAP

This is an effect created by Jaap Menist. The how-to text file can be found in the zip file that this FAQ came in (the how-to is very long). The name for this file is TDOOR2.TXT. This is a very difficult door to construct, but it ends in a very cool effect.

NOTE: Using this method, the door can start in the closed position.

4.1.12 Sliding Doors

Map Name: ADVANCED.MAP (ROOM 1-C)

Again, due to the extremely difficult task of explaining this how-to, I have provided a step by step example in the tutorial map, showing how to make this door. Just take a look at the example and it should make sense (it goes step by step in the building process).

4.1.13 See Through Sliding Doors

Map Name: STHRUSLD.MAP

This is an effect created by Jaap Menist. The how-to text file can be found in the zip file that this FAQ came in (the how-to is very long). The name for this file is STHRUSLD.TXT. This is a very difficult door to construct, but it ends in a very cool effect.

4.1.14 Rotating Subway Doors

Map Name: TUTORIAL.MAP (ROOM 1-G)

This type of door constantly revolves in a circle, much like the doors you find in an airport or subway station.

STEP 1: Create your door sector to be rotated. The sector should be valid player space. Raise the floor of the door up to the ceiling (or lower the ceiling to the floor).

STEP 2: Place one sector effector sprite in the middle of the door sector. Give this sprite a lotag value of 1. Give it a unique hitag value. If you want this door to rotate counter-clockwise, angle the sprite straight down. If you want the door to rotate in a clockwise manner, angle the sprite straight up.

STEP 3: Place one sector effector sprite somewhere else in the door sector. Give this sprite the same hitag value you gave the sprite in step 2. Leave the lotag value as 0. Change the angle of this sprite to point straight down. You are now finished - take a look at the tutorial map if you don't quite understand.

NOTE 1: Several versions of this door can be made. Look at the tutorial example to see the different versions in action.

4.1.15 Making Doors Auto Close

Map Name: TUTORIAL.MAP (ROOM 1-H)

To make any door auto-close, place one sector effector sprite in the door sector, and give it a lotag value of 10. The hitag value is the delay that the door will wait before it closes. A value of 32 is one second, 64 is two seconds, and so on.

4.1.16 Making Locked Doors

Map Name: TUTORIAL.MAP (ROOM 1-I)

To make a locked door (that must be opened with a keycard) follow these steps:

STEP 1: Place an access pad sprite (#130 or #170) somewhere in your map. Give it a hitag value of 212. Give it a unique lotag value.

STEP 2: Place an ActivatorLocked sprite (#4) in the door sector. Give this sprite the same lotag value you gave the access pad.

STEP 3: Place a key sprite (#60) somewhere in your level.

STEP 4: Change the palette of *both* the key and the access pad (use [ALT]+[P] in 3D mode) to the following values to make one of the 3 different kinds of keys and locks:

- 0 = Blue

- 21 = Red
- 23 = Yellow

Copyright © 1999 by Jonah Bishop. All rights reserved.

click on the coin and win \$20,000



4.2 Parallaxed Skies

[Previous Section](#)

[Table of Contents](#)

[Next Section](#)

4.2.1 Correctly Parallaxed Skies

Map Name: TUTORIAL.MAP (ROOM 2-A)

Parallaxed skies present the player with the feeling of being outside, but can be problematic if not constructed correctly. Follow these simple steps:

STEP 1: Change the texture of the ceiling or floor that you wish to parallax.

STEP 2: Move the mouse cursor over the area you want parallaxed and press the [P] key. To prevent the space textures from killing you when you walk under them, change the palette value of the parallaxed floor or ceiling (or both) to 3. There are 3 types of parallaxing. Press the [CTRL]+[P] key combination to cycle through the types of parallaxing. To make the walls look parallaxed (used in the space levels to make it look as if Duke looks into space), create several sectors along the sides of the main parallaxed sector, change the floor and ceiling texture to the same texture that the main sector has, and parallax both the floor and ceiling of these side sectors. Then lower the ceiling down to the floor. Take a look at the tutorial map if you do not quite understand.

4.2.2 Problems And Fixes With Parallaxing

Map Name: NONE

PROBLEM 1: My sky is screwed up! It shows one picture, and then to the side of that another, and another, etc. What's the deal?

SOLUTION 1: The textures that use more than one frame (such as the space texture) usually are the cause of this problem. For example, the space texture uses six textures; the first texture and five after that. When you use a texture such as this *anywhere* in your level, it becomes impossible to have one (and only one) texture parallaxed elsewhere in your map. So suppose I have the space texture somewhere in my level, and I want to parallax a texture that only has one frame. If I try to parallax this second texture, it will show the texture I'm using, the frame after that (a sprite let's say), the frame after that, etc. The solution for this is basically this: use the same texture everywhere. If you want to just use a one frame texture, you must change *all* the multiple frame sky textures to single frame sky textures.

PROBLEM 2: I'm not using multiple frame sky textures, and it still looks weird. What am I doing wrong?

SOLUTION 2: This problem occasionally occurs upon first parallaxing a sector, and can be resolved by saving your level, exiting from Build, and then reloading your level.

Copyright © 1999 by Jonah Bishop. All rights reserved.



4.3 Lighting Effects

[Previous Section](#)

[Table of Contents](#)

[Next Section](#)

4.3.1 Making Lights Blink

Map Name: TUTORIAL.MAP (ROOM 2-B)

Blinking lights can add a great mood and sense of environment in any user level. This effect makes lights blink randomly.

STEP 1: Create a light sector. This sector can either be in a wall, or a ceiling light sector, but it needs to be a sector. Take a look at the tutorial map to see examples of both types of lighting.

STEP 2: Place one sector effector sprite in each sector you want to light up. Give these sprites a lotag value of 4, and a unique hitag value. The hitag value is the flash randomizing number, so use any value you want.

STEP 3: Set the shade of the walls, floor, and ceiling near your light to the shade value you want it to be when the lights are off. Then, set the shade of the sector effector sprites to the shade value you want it to be when the lights are on. You set the shade value by pressing the [']+[S] key on the desired wall, ceiling, or floor in 3D mode. You are finished - Look at the tutorial map if you don't quite understand.

NOTE 1: You can make colored blinking lights as well. Simply give the sector effector sprites a palette value equal to the color that you want the light to be when flashing. Both the sectors and the sector effector sprites can have different palette values.

NOTE 2: This effect can *not* be used with the light switch effect.

NOTE 3: To prevent one side of a red line in 2D mode (a 2 sided wall) from being affected by the light effect, give the side of the wall you want to not be affected a wall hitag value of 1.

4.3.2 Making Lights Blink When Shot

Map Name: TUTORIAL.MAP (ROOM 2-C)

This effect allows you to cause a light to flash randomly once it is broken by the player (by the player shooting the light).

STEP 1: Create a light sector. This sector can either be in a wall, or a ceiling light sector, but it needs to be a sector. Take a look at the tutorial map to see examples of both types of lighting. If you are constructing a light in a wall, make sure that you lower the ceiling of the light sector to the floor, placing the light texture on the wall face that appears once you lower the sector ceiling to the floor.

NOTE 1: The light texture that you use, should be breakable (i.e. - when you shoot the light, the light texture should change to the corresponding broken light texture).

STEP 2: If you are creating a ceiling light, give the light sector a unique sector hitag value. If you are creating a wall light, give the red wall that separates the light sector from the wall sector a unique wall lotag value.

STEP 3: Place one sector effector sprite in each sector you want to light up. Give these sprites a lotag value of 3. If you are creating a wall light, give the sector effector sprite a hitag value equal to the wall's lotag

value. If you are creating a ceiling light, give the sector effector sprite a hitag value equal to the light sector's hitag value.

STEP 4: Set the shade of the walls, floor, and ceiling near your light to the shade value you want it to be when the lights are off. Then, set the shade of the sector effector sprites to the shade value you want it to be when the lights are on. You set the shade value by pressing the '['+[S] key on the desired wall, ceiling, or floor in 3D mode. You are finished - Look at the tutorial map if you don't quite understand.

NOTE 2: You can make colored blinking lights as well. Simply give the sector effector sprites a palette value equal to the color that you want the light to be when flashing. Both the sectors and the sector effector sprites can have different palette values.

NOTE 3: This effect *can* be used with the light switch effect.

NOTE 4: To prevent one side of a red line in 2D mode (a 2 sided wall) from being affected by the light effect, give the side of the wall you want to not be affected a wall hitag value of 1.

4.3.3 Open Door & Room Lights Up

Map Name: TUTORIAL.MAP (ROOM 2-D)

This effect allows rooms to light up when a door is opened (assuming that light would come in the door).

STEP 1: Construct the door and the sectors that you wish to light up.

STEP 2: Place one sector effector sprite in the door and in each sector that you want to light up.

STEP 3: Change the shade of each sector effector to the shade you want the sector it is located in to be when the door is open. Do this by moving the mouse over the sprite, and press the '['+[S] keys in 3D mode.

STEP 4: Change the shade of each wall, ceiling, and floor of the sectors you want to light up to the value you want them to be when the door is closed.

STEP 5: Give each sector effector sprite a lotag value of 8 and the same, unique hitag value. You are now finished - Take a look at the tutorial map if you don't quite understand.

4.3.4 Light Switches

Map Name: TUTORIAL.MAP (ROOM 2-E)

This effect allows you to create a light switch from which the player can toggle a light on or off.

STEP 1: Place a switch sprite (#712 is the standard light switch) somewhere in your level. Give this sprite a unique lotag value.

STEP 2: Place one sector effector sprite in each sector that will light up when the light is on. The shade of this sprite determines how bright the light will be in that sprite's particular sector when the light is on.

STEP 3: The shade of all walls, ceilings, and floors of the sectors to be lit up should be set to the shade they will be when the light is turned off.

STEP 4: Give each sector effector sprite a lotag value of 12. Give each sector effector sprite the same hitag value that you gave to the light switch. You are now finished - Take a look at the tutorial map if you don't quite understand.

4.3.5 Colored Lights

Map Name: TUTORIAL.MAP (ROOM 2-F)

To create colored lights, simply change the palette of the walls, floors, and ceilings you desire to be colored to one of the following values (palette is set in 3D mode by pressing the [ALT]+[P] keys when the mouse is over the desired object to change):

- 1 = Blue
- 2 = Red
- 7 = Yellow
- 8 = Green

4.3.6 Pulsating Lights

Map Name: TUTORIAL.MAP (ROOM 2-G)

Pulsating lights can be used to create some neat effects (moving lights to name one).

STEP 1: Place a Cycler sprite in the sector(s) that you want the light to pulsate in. Change the shade of this sprite to how bright you want the light to be when the light is at its brightest point.

STEP 2: Change the shade of the floor to how dark the sector should be when the light is at its darkest point.

STEP 3: To change the speed of the light pulsing, simply place one GPSSpeed sprite in the same sector as the Cycler sprite. A hitag value for the Cycler and GPSSpeed sprites are not needed (as was previously thought to be the case). Now give the GPSSpeed sprite a lotag value. This value is how fast or slow the light will pulsate (the higher the value, the faster the light will pulsate).

NOTE 1: Here is a great bit of information on the speed of Cycler sprites when combined with GPSSpeed sprites (thanks to Keith Tyson for this information). The cycle length (going from bright to dark to bright again) seems to be halved as the GPSSpeed lotag value is doubled. Strangely enough, this is the case up to a value of 1023, (a value of 1024 doesn't work). From 1025 to 2047, the above pattern is reversed: As the GPSSpeed lotag value gets higher, the cycle length increases. A value of 2048 doesn't work either. A simple conversion indicates that a lotag of 1 will produce roughly one cycle in 120 seconds (quite slow), while a lotag of 1023 will produce roughly 8 cycles per second (quite fast).

NOTE 2: Walls with a hitag value of 1 are not affected by the Cycler sprite.

NOTE 3: This effect can be used to create the "rotating" lights found in some of the episode 2 levels. Take a look at those levels to see how this is done.



Click on me

and win **\$20,000**



4.4 Moving Sectors

[Previous Section](#)

[Table of Contents](#)

[Next Section](#)

4.4.1 Rotating Sectors (Gears)

Map Name: TUTORIAL.MAP (ROOM 2-H)

This effect allows you to create a constantly rotating sector, such as the gears found in the registered levels.

STEP 1: Create the sector to be rotated.

STEP 2: Place one sector effector sprite where you want the pivot point for this sector to be (it is usually placed in the center). Give this sprite a lotag value of 1, and a unique hitag value. Now change the angle of this sprite to one of the following:

- Up - Sector rotates counter-clockwise
- Down - Sector rotates clockwise

NOTE 1: This pivot point sprite is not required to be inside the sector that will rotate.

STEP 3: Place one other sector effector sprite somewhere else in your gear sector. Give this sprite the same unique hitag value that you gave the first sector effector sprite. Leave the lotag value set as 0. Then set the angle of this sector effector sprite. The following angles have the following related functions:

- Up - All points of sector rotate around pivot point
- Down - Entire sector rotates around pivot point

STEP 4: If you want sprites and the player to rotate with this sector, you must enter 3D mode and press the [R] key on the floor of the sector. This will allow sprites and actors to move with the rotating sector.

4.4.2 Using Rotating Sectors for Doors

Map Name: ROTATEDR.MAP

Rotating sectors can be used in combination with switches to create a very interesting type of door. The how-to is fairly complicated, so a tutorial map (created by Jack Gutzler AKA Captain Massive) has been included with this package.

4.4.3 Two-Way Trains

Map Name: ADVANCED.MAP (ROOM 1-D)

Two-way trains are very cool and a great addition to any level. They are, however, quite complex to build.

STEP 1: Create the track for the train (it should be straight), the train itself, and two rooms at either end of the track.

NOTE 1: The train sector should not be split into multiple sectors. However, the train can include white-walled (non-valid player space) sectors and nested red-lined sectors (take a look at the tutorial map to see exactly what is possible).

NOTE 2: The train should be made on the opposite side of the track from which the player will enter first. And the train should not be made right in front of where the player will board it (place the train in the middle of the track for now).

STEP 2: Place three switches on the map: one in the train, and the other two in both rooms at either end of the track (i.e. - one in each room).

STEP 3: Give all three switch sprites the same, unique lotag value.

STEP 4: Place one activator sprite in the train and give it the same lotag value that you gave the switch sprites.

STEP 5: Place one sector effector sprite in the train and give it a lotag value of 30. Give this sprite a hitag value of 1 added to the value you gave the switch sprites (so if your switch sprite lotag was 10, the hitag value of this sector effector sprite would be 11). You now must change the angle value for this sprite. Point it in the direction of the track where the train will start from (if your train starts from the left hand side, point the sector effector sprite to the left).

STEP 6: Give the train sector a sector lotag value of 31.

STEP 7: Place two locator sprites (sprite #6) on your track, one where the train will begin and one where the train will stop. Think of these two sprites as reference sprites. The sector effector sprite that you placed in your train sector is a fixed point on your train. When the train moves from point to point, the fixed sector effector on the train will move over top of the locator sprites. So place these in an appropriate place. Give the locator closest to where the player will first board the train a lotag value of 1 and a hitag value of 1. Do not give the other locator sprite a lotag or hitag value (both should remain at the default value of 0).

STEP 8: In 3D mode, press the [R] key on the floor of your train. If your train has a ceiling, do the same thing.¹

OPTIONAL STEPS

STEP 9: Create door sectors on each end of the track where the player will board and exit the train. They should be regular Doom-type doors. Place one activator sprite in each door sector. Give the activator sprite in the door where the player will first board the train, a lotag value of two more than the switch lotag. For example, if the switch lotag (near the door) was 10, the lotag of this activator sprite would be 12.

STEP 10: Give the other activator sprite (in the other door sector) a lotag value of one more than the switch sprite lotag. For example, if the switch lotag was 10, the lotag of this second activator sprite would be 11.

STEP 11: Make sure that the door where the player will first board the train is in the closed position. The other door (at the other end of the track) should be open. If this is not set properly, the timing of the doors will be off.

4.4.4 One Car Subways

Map Name: ADVANCED.MAP (ROOM 1-E)

Subways are not difficult to master, but they do require patience and a great deal of work. But they can be done, and they can be a great addition to any level!

STEP 1: Make the track for the subway. For this example we will be making a simple rectangle room. Later on, when you get practice with the Locator sprites, you will be able to have a track with a more realistic look (i.e. - Walls on both sides of the track. Here we are just using an open room which the subway will wander around in). Set your textures to whatever you want.

STEP 2: Make a sector off of this room (or anywhere else for that matter) to be a station for the train. Make sure the sector is valid player space, and raise it up from the floor of the track sector in 3D mode (this sector

will be higher up than the track).

STEP 3: Make one other rectangular sector that will be the subway car itself. Do not make the car right next to the station, as a locator sprite will be needed for that. Raise the car up out of the ground, and make it valid player space. Choose appropriate textures.

STEP 4: In 2D mode, put the mouse cursor in the subway car sector and press the [H] key. You now will give that sector a hitag of 1. This number is an indicator for the cars, so if you had a second subway somewhere else on your map, the cars would have a hitag of 2.

STEP 5: Now insert a sprite in the subway car sector by pressing [S] in 2D mode. Now go into 3D mode and press [V] on the newly made sprite twice. Change the sprite into a sector-effector sprite. Place the sprite in the center of the car and change the angle of the sprite to be in the direction of the front of the car. The sprite's angle tells the subway in which direction it will go. Press [R] on the floor of the subway (and on the ceiling too if your subway has a ceiling. If it does not have a ceiling, leave it alone) in 3D mode. This makes the floor move with the train.

STEP 6: Go into 2D mode and press [ALT]+[T] on the sprite, and enter the value of 6. This gives the sector-effector sprite a lotag of 6 and it makes this car the "engine" of the subway. When you make a multiple car subway, this will be the lead car of the subway. It will pull all of the other cars behind it.

STEP 7: Now you must place the locator sprites all over the track. These sprites make the actual 'path' of the train. Place a locator sprite near the station and give it a hitag of 1, and leave the lotag at 0. The hitag of 1 tells the subway that it must stop at this point of the track and wait for five seconds.

STEP 8: Now put Locator sprites all around the track. Put the sprites anywhere in your track that you want your train to go. The train will go over all the Locator sprites you put down, so place them in the way you want your train to go.

STEP 9: You now must give the Locator sprites lotags that go in numeric order. For example, the Locator that the train will go to, AFTER visiting the station would have a lotag of 1. The next sprite will have a lotag of 2, the next a lotag of 3, and so on. This will continue until you get back to the station locator (the one with the hitag of 1). DO NOT GIVE THE STATION LOCATOR A LOTAG! IT MUST BE LEFT AT 0! Make sure that you make the track for the subway as accurate as possible, as the subway can sometimes disappear. You are all done at this point! If you have any problems, or are unsure about how something is made, look at 1CARSUB.MAP.

4.4.5 Making Multi-Car Subways

Map Name: ADVANCED.MAP (ROOM 1-F)

Multi car subways are just as easy as one car subways, but it requires some work and time.

STEP 1: Make the first car, or "engine" of the train by following the instructions in the above how-to.

STEP 2: Make several more car sectors behind the engine of the train, make them valid player space, and choose appropriate textures for the car. The train cars should *not* be connected, and there should be some space between the two. When the train turns around a corner, sometimes the train sectors can overlap, causing some strange bugs to appear. So don't put them too close. Also make sure that the cars are raised out of the ground to the appropriate height (whatever you want it to be).

STEP 3: Give the car sectors a hitag of 1 (not necessary, but it is if you have more than one subway on your map. If you had a second subway, the cars hitag would be 2, and so on).

STEP 4: Put a sector effector sprite in the cars of the train and give them a lotag of 14, which means that these cars will follow the "engine" of the train.

STEP 5: You're done! The path for the train should have already been made, as you constructed that in the

first how-to. Take a look at MULTISUB.MAP if you have any other problems. This is quite easy to do.

4.4.6 Notes on Subways

Map Name: NONE

Here is a short list of notes that might make your subway making problems disappear. Many problems can be solved with some of these notes, so read on!

1. The locators should all be in the same sector (the track). It is not advised to split up the track sector, but if you must do it make sure the locators are all in the same sector.
2. The pixelheight (height between floor and ceiling) of the train has to have a maximum of 104 in order to run you over. If the pixelheight is more than this the train will pass through you instead of killing you (weird eh?).
3. A train with cars behind it will travel slightly faster than a single train! So be aware that when you put 2 trains on the same track, after quite a while, they will have caught up with each other (I actually did this).

Copyright © 1999 by Jonah Bishop. All rights reserved.



4.5 General Sector Effects

[Previous Section](#)

[Table of Contents](#)

[Next Section](#)

4.5.1 Earthquakes

Map Name: TUTORIAL.MAP (ROOM 2-I)

Earthquakes are a neat effect, and easy to make.

STEP 1: Create the sectors that will be altered during the earthquake (if you want any). Make these sectors look as if the earthquake has already happened.

STEP 2: Place one sector effector sprite in each sector that will be affected by the earthquake. Give each sprite a lotag value of 2.

STEP 3: Place one Masterswitch sprite in each sector that will be affected by the earthquake. Give each of these sprites the same, unique lotag value.

STEP 4: Raise the sector effector sprites and the Masterswitch sprites to the height that you want the affected sectors to be before the earthquake. If you have a sloped sector, place the sprites on the sector's axis line.

STEP 5: Place one touchplate sprite in the sector that you want to trigger the earthquake with. Give it the same lotag value that you gave to the masterswitch sprites. You are now finished - Take a look at the tutorial map if you still don't understand.

NOTE 1: To make the earthquake spawn scraps of rock and metal, simply place sector effector sprites where you want the scraps to fly from. Give each sprite a lotag value of 33. Then raise the sprite to the height that you want the scraps to be spawned from (it doesn't matter what that height is).

4.5.2 Rooms Over Rooms

Map Name: TUTORIAL.MAP (ROOM 3-A)

Making rooms over rooms can be a complex job, but is quite simple to do. A few simple rules need to be properly followed when constructing this effect, and if done properly, this effect can create some cool architectural layouts.

The first rule is simply this: The ceiling of the lower level must be lower than the floor of the upper level. If this isn't the case, your level will show some strange anomalies associated with level over levels, and some undesired results will show up.

Another rule also must be followed when constructing your level over level sectors. You cannot "see" the lower level from the upper level, and you cannot "see" the upper level from the lower level. This means that you could not make a 2 level house and have a window on the bottom floor and a window directly above it on the top floor.

You should also remember that the vertices of the upper level should not connect to any of the walls on the lower level. The levels must be completely independent of each other. Take a look at the tutorial map to see just how it works.

It is possible to have many layers of sectors over sectors, just make sure you keep track of them in Build (it can become quite complex).

4.5.3 Stretching Sectors

Map Name: TUTORIAL.MAP (ROOM 3-B)

This effect will stretch a sector in the direction you desire.

STEP 1: Create the sector that will be stretched (it should be a valid player sector).

STEP 2: Give the sector a sector lotag value of 27.

STEP 3: Place 3 sprites in this sector: one activator, one sector effector, and one GPSSpeed sprite.

STEP 4: Place a trigger (touchplate or switch) somewhere in your level, and give it a unique lotag value.

STEP 5: Give the activator sprite a lotag value equal to the switch lotag value. Give the sector effector a hitag value equal to the switch lotag value. Also give this sector effector sprite a lotag value of 20. Point the sector effector sprite in the direction that you want to stretch the sector. Now give the GPSSpeed sprite a lotag value equal to the distance that you want the sector to stretch (a value of 1024 is equal to the largest grid square in Build).

4.5.4 Conveyor Belts & Water Current

Map Name: TUTORIAL.MAP (ROOM 3-C)

This effect is used to create conveyor belts and water currents.

STEP 1: Create the conveyor belt sector(s).

STEP 2: Place one sector effector sprite in each sector, and change the angle of the sprite to the direction that you want the movement to go. Give the sector effector sprites a sprite lotag value of 24.

NOTE 1: The floor texture may need to be flipped (with the [F] key) or placed in relative alignment (with the [R] key) if you are using an animated texture (so that the texture animates in the proper direction).

NOTE 2: To make the movement of the sectors go faster, simply place one gpspeed sprite in each sector, giving it a sprite lotag value of the speed you want it to travel (higher numbers mean faster speeds - 64 and 128 are common values).

NOTE 3: To create a water current, follow the same steps above, and simply give each sector a sector lotag value of 1.

NOTE 4: To create switch operated conveyor belts, place a switch in your level and give it a unique lotag value. Give the sector effector sprite (with the lotag value of 24) the same hitag value as the switch lotag value.

4.5.5 Making Curtains Draw Back

Map Name: TUTORIAL.MAP (ROOM 3-D)

This effect makes curtains draw back when a switch is flipped (much like in the movie theater in episode 1, level 1).

STEP 1: Create your curtain sector. It should be valid player space. For best visual effects, it is advised to lower the ceiling of the sector almost, but not all the way, down to the floor (look at the tutorial map to see how it should look). Give the curtain sector a sector lotag value of 27.

STEP 2: Place a switch sprite somewhere in your level, and give it a unique lotag value.

STEP 3: Place one sector effector sprite, one GPSPEED sprite, and one activator sprite in the curtain sector.

STEP 3: Give the sector effector sprite a lotag value of 20, and a hitag value equal to the switch lotag value.

STEP 4: Give the activator sprite a lotag value equal to the switch lotag value.

STEP 5: Give the GPSPEED sprite a lotag value equal to the distance that the curtain sector will travel. A value of 1024 is equal to the largest grid square in Build.

STEP 6: If the ceiling of your curtain sector does not touch the floor of the sector, make sure you set the floor texture to relative alignment (by pressing the [R] key in 3D mode on the floor).

NOTE 1: Make sure that your curtain sector does not travel into the wall it is joined to, or a hall of mirrors (HOM) effect will occur.

4.5.6 The Pool Table

Map Name: NONE

The pool table (as found in episode 1, level 2) is a great effect to add to almost any level. It is simply 2 rectangular sectors, raised to a realistic height. The pocket sprite (#903) is placed on the table in the places that the pockets are found, and the ball sprites (#901 and #902) are also placed on the table. No special tags or other values are needed (just make sure to set the palette of the pool balls to different values, so the balls don't all look the same).

4.5.7 Fish Tanks

Map Name: NONE

The fish tank is quite easy to make. Simply create the tank itself (which is comprised of sectors and masked walls (with the glass texture set on them)), and give the water sector a sector lotag of 2. Then place the appropriate sprites (sharks, underwater plants, etc.) in the tank. NOTE: Sharks can bite the player through the glass.



3 check out...

Quake 3 Arena

epworld.com



4.6 Water

[Previous Section](#)

[Table of Contents](#)

[Next Section](#)

4.6.1 Submergible Water

Map Name: TUTORIAL.MAP (ROOM 3-E)

Submergible water is a neat effect, and easy to make.

STEP 1: Create your "source sector" (i.e. - the above water sector). Give this sector a sector lotag value of 1 (by pressing the [T] key inside the sector in 2D mode and entering in the value).

STEP 2: Create a "destination sector" (i.e. - the below water sector). Give this sector a sector lotag value of 2.

NOTE 1: The destination sector should be *exactly* the same size and shape as the source sector. If it is not, the effect will not work properly.

STEP 3: Place one sector effector sprite in the source sector. Also, place one sector effector sprite in the destination sector. The sprite in the destination sector must be in the *exact* same place as it is in the source sector.

STEP 4: Give both sprites a lotag value of 7. Then give both sprites the same, unique hitag value. You're done - Take a look at the tutorial map if you don't quite understand.

4.6.2 Sloped Water Surfaces

Map Name: TUTORIAL.MAP (ROOM 3-F)

To make it possible to submerge into an underwater area with a sloped floor, simply place the sector effector sprite in the underwater sector on the lowest point (the axis line) of that sector.

4.6.3 Swimming in Slime

Map Name: TUTORIAL.MAP (ROOM 3-G)

To create an underwater area resembling slime, simply give the sector effector sprite in the underwater sector a palette value of 8. This is done by pressing the [ALT]+[P] keys on the sprite in 3D mode. Red water (like lava) can also be created by giving the SE sprite in the underwater sector a palette value of 1.

4.6.4 Making Waves

Map Name: TUTORIAL.MAP (ROOM 3-H)

To have waves work properly, you must follow a preset rule. This rule isn't too complex, but it must be followed.

RULE 1: The wave sector(s) must be four-sided. If they are not, the wave will not work correctly.

STEP 1: Create several four-sided sectors (side by side - they should share a red line with their adjacent sector).

STEP 2: Place one sector effector sprite in each sector that you created. Give each sprite a lotag value of 29. Then give each sprite any hitag value between 0 and 2047. This hitag value is the starting height of the wave.

STEP 3: Place one gpspeed sprite in each sector, and give it any lotag value between 0 and 2047. This value is the height of the wave after it starts.

NOTE 1: If the waves are too high, Duke will not be able to stay afloat (provided that the sector is submergible).

NOTE 2: Wave sectors can be made into submergible sectors by following the standard guidelines for submergible water.

4.6.5 The Rising Water Effect

Map Name: TUTORIAL.MAP (ROOM 3-I)

This interesting effect simulates water rushing into an area and slowly rising (as it would realistically do).

STEP 1: Create a submergible water area (one above water and one below water sector). See the submergible water how-to for the instructions on the construction of this effect.

STEP 2: Create a sector that the player will pass through, and place one touchplate sprite in this sector. Give this sprite a hitag value of 1 and a unique lotag value.

STEP 3: For the "source sector" (the above-the-water sector), place the following sprites: 1 sector effector, 1 masterswitch, and 1 gpspeed.

NOTE 1: The gpspeed sprite is purely optional, but is used in slowing the rising effect of the water.

STEP 4: Give the sector effector sprite a lotag of 31, and a hitag value equal to the touchplate lotag value. The angle of this sprite should also face upwards.

STEP 5: Give the masterswitch sprite a lotag value equal to the touchplate lotag value.

STEP 6: Give the GPSPEED sprite any lotag value you want (the higher the number, the faster the water rises; the lower the number, the slower the water rises).

STEP 7: The "above-the-water" sector should look as if the water has not yet risen. Raise the sector effector sprite (with the lotag of 31) to the height that you want the water to rise to (the water will rise to the bottom of the sector effector sprite).

STEP 8: Place the following sprites in the "under-the-water" sector: 1 sector effector sprite, 1 masterswitch sprite, and 1 gpspeed sprite.

NOTE 2: Again, the gpspeed sprite is purely optional, but is used in slowing the rising effect of the water. The rate for this sprite should also be the exact same as in the "above-the-water" sector.

STEP 9: Make sure that the underwater sector looks as if the water has already risen.

NOTE 3: It is best if the ceiling of this sector is exactly as high as the rising height of the floor in the "above-the-water" sector.

STEP 10: Give the sector effector sprite you placed in step 8 a lotag value of 32, and leave the hitag value as 0. The angle of this sprite should be down, and this sprite should be raised to the height that you want the water in this sector to *start* at.

STEP 11: Give the masterswitch sprite placed in step 8 the same lotag value as the touchplate lotag value.

STEP 12: Give the GPSPEED sprite any lotag value you want (the higher the number, the faster the water rises; the lower the number, the slower the water rises).

STEP 13: Enter 3D mode while in the "under-the-water" sector and press the [O] (the letter o, not the number zero) key on all the walls in this sector. If you do not do this, the effect will look rather unrealistic. You are now finished - take a look at the tutorial map if you do not understand.

4.6.6 Level Over Level Water Trick

Map Name: TUTORIAL.MAP (ROOM 4-A)

This is an extremely cool effect, and was originally designed by Ben Roffelsen. This effect consists of two individual how-tos, joined together in an interesting way. Using level over level techniques and submergible water, a "true submergible water effect" can be constructed (i.e. - when you look at the map in the game, and you submerge into the water, you will not 'teleport' to another sector, but will appear as if you are still in the same area (which you are)). Read through this entire how-to before constructing this effect. Some things you might want to consider are towards the end of this how-to.

STEP 1: Construct the submergible water effect (one above-water sector and one below-water sector). These two sectors should be the exact same size and shape.

NOTE 1: The floor of the above-water sector should be higher than the ceiling of the below-water sector. If this rule is not followed, this effect will not work.

STEP 2: Select the above-water sector (using the right [ALT] key and dragging with the mouse pointer to form the selection box). Turn off grid lock by pressing the [G] key. Move the selected sector over the below-water sector. Make sure that the vertices of the above-water sector are as close as they can be without being directly on top of the vertices of the below-water sector.

NOTE 2: Turning the grid lock off is required. If the vertices of the above-water sector are exactly where the vertices for the below- water sector are, the sectors will join, and the whole construction will be ruined (you'll have to start over).

STEP 3: Now create a passage from the lower level (where the below-water sector is located) to the upper level (where the above-water sector is located). Take a look at the tutorial map to see how I did it.

NOTE 3: You may want to create the passage before moving the above-water sector. This may make it easier to connect the passage with another part of your level.

You are now complete. Take a look at the tutorial map to see exactly how this effect was created.

4.6.7 Timed Underwater Airlocks (Complex)

Map Name: AIRLOCK.MAP

Due to the complexity of this effect, I have only provided this effect as a tutorial map. The map was created by Gijs de Vries, and uses a two-way train for timing. The map shouldn't be too hard to pull apart.

4.6.8 Rising Water on Sloped Surfaces (Complex)

Map Name: SLOPEWTR.MAP

Due to the complexity of this effect, I have only provided this effect as a tutorial map. The map was created by Gijs de Vries. This is an advanced effect, so take your time in pulling it apart.



Click on me

and win **\$20,000**



4.7 Walls And Wall Effects

[Previous Section](#)

[Table of Contents](#)

[Next Section](#)

4.7.1 Masked Walls

Map Name: NONE

To make a masked wall, point the cursor at the general location that you want the mask to be (the mask must be placed on an existing red line somewhere in the level), and press the [M] key. It is advised that you point at the floor and not at the ceiling when placing the masked wall. Now you can change the texture, shade, shift, and most anything else about the mask. Just remember that a masked wall has **two** sides, and both sides must be edited.

4.7.2 One-Way Walls

Map Name: NONE

One-way walls are just as easy to create as masked walls. Do the exact same thing as you do in the masked walls how-to, this time pressing the [1] key. This time your mask will only have 1 side to edit, not 2. The other side will be invisible, so make sure you have the invisible side on the side you want it to be. The side on which you press the [1] key will be the visible side.

4.7.3 Forcefields and Controlling Them

Map Name: TUTORIAL.MAP (ROOM 4-B)

Forcefields are an effect that has many uses, but are easy to make. To construct a forcefield, follow these steps:

STEP 1: Create a masked wall where you want the forcefield to be.

NOTE 1: This mask should not be on a wall where the player can see the wall itself. Doing so causes the wall texture to shift, which makes the effect quite ugly. Create one buffer sector on either side of the forcefield to prevent the player from seeing this problem. Take a look at forcefield #1 in the tutorial map to see this error in action. Then look at forcefields #2 and #3 to see the "buffer sectors" that make the effect non-visible.

STEP 2: Change the texture of this mask to one of the following:

- BIGFORCE - Tile #230
- W_FORCEFIELD - Tile #663

NOTE 2: The BIGFORCE tile is used to make forcefields that are invisible during the game, and which do not hurt the player when they player touches them. The W_FORCEFIELD tile is used to make forcefields that flicker (you can see them), and the player does get hurt when they touch the forcefield. Forcefields made with the W_FORCEFIELD texture cannot be made invisible. In the tutorial map, forcefields #1, #2, and #4 are made with the W_FORCEFIELD texture, while forcefield #3 is made with the BIGFORCE texture.

STEP 3: While in 3D mode, press the [B] and [H] keys on the masked wall. This makes the masked wall block you and other characters from walking through the wall, and it makes bullets "hit" this wall.

STEP 4: Now point at the mask in 3D mode and press the [M] key. The mask will disappear, but all the attributes are left the same. You must do this, so that the forcefields will appear in the game.

Making the Forcefield Switch Operated

NOTE 1: You should only make forcefields using the W_FORCEFIELD texture operated by a switch. That way you can actually see if the forcefield is on or off.

STEP 1: Give the masked wall a unique lotag.

STEP 2: Place a switch somewhere in your level and give it the exact same lotag value as the masked wall.

4.7.4 Wall Information

Although they seem basic enough, walls have several hidden features that make them a little more complex than generally thought. The first thing that should be mentioned is two sided walls. Two sided walls appear as red lines in 2D mode, and they have two completely separate sides with completely unrelated attributes. If you set the shade of one side of a two sided wall to 10, the other side will still be zero (or whatever the default for that wall was). This also means that two sided walls have four tags: 2 hitags and 2 lotags (1 each for either side). Below is a list of wall tags and what they can do.

Hitag = 1 ; Lotag = 0

A hitag of one is a pretty complex subject. When you give a wall a hitag of 1, the wall is EXCLUDED from a certain function or effect. I've only seen this tag used in lighting effects such as flickering lights, flickering lights when shot, and light switches. If you have a sector that is brightened by a light, and you want to exclude a wall in the sector from being bright, give it a hitag of 1. This is particularly effective in using the "open door, room lights up" effect. Use this to force the door to stay dark while the room lights up. This is a quite useful tag.

Hitag = 0 ; Lotag = X

Giving a wall a lotag value of X (any number you want), in addition to any "doortile" texture, will transform a door into a "switch". An example of this tag can be seen in Episode 1, Level 2, at the entrance to the book store. The revolving door uses a certain sector-effector sprite and a sector tag to make it revolve. This sector effector and sector tag are run by way of an activator. Let's assume that the lotag of the activator is 12. You would give the walls of the door a lotag of 12, so when the player presses space on the door, the sector effector sprite function is called, and the effect takes place. This can be used in many different ways.

So what is a "doortile" texture? It's simply a texture with the name doortile appended to it. The available doortile textures are as follows:

#150 through #159, #395, #447 through #449, #717, #781, #1102, #1144, #1169, #1178, #1179, and (only in versions 1.4 and 1.5) #4391.



3 check out...

Quake 3 Arena

sp world.com



4.8 Mirrors

[Previous Section](#)

[Table of Contents](#)

[Next Section](#)

4.8.1 Mirrors

Map Name: MIRRORS.MAP (ROOM #1)

Mirrors are basically a one-way wall that uses the mirror texture, which is texture #560. Some funny stuff exists with mirrors, but when made properly, they can be a great addition to any level.

STEP 1: Create a one-way wall where you want the mirror to be placed. Remember that one-way walls can only be made on red lines in 2D mode. They are created by pressing the [1] key in 3D mode where you want the mirror to go. Change the texture of this one-way wall to texture #560. If you have created this one-way wall correctly, you will not be able to see through the mirror texture (you will only see a pink background).

NOTE 1: This mask should not be on a wall where the player can see the wall itself. Doing so causes the wall texture to shift when the mirror is broken by an explosive weapon, which makes the effect look quite ugly. Create one buffer sector on either side of the mirror to prevent the player from seeing this problem. Take a look at the mirror in the tutorial map to see to see the "buffer sectors" that make the effect non-visible.

STEP 2: Create a sector behind the mirror (on the opposite side from which the player will view the mirror), and make it *quite* large.

STEP 3: In 3D mode, press the [B] and [H] keys when the mouse pointer is over the mirror. This makes the mirror breakable and it can be hit by Duke's weapons.

STEP 4: If you want Duke to say "Damn, I'm looking good" when you press the space bar while looking into the weapon, give the side of the mirror that the player will look into a wall lotag of 252. You can do this in 3D mode, by pressing the [=]+[T] key on the mirror texture, and entering the value. Or you can do it in 2D mode by moving the mouse pointer to the side of the mirror the player will be on, and pressing the [ALT]+[T] key (make sure the grid is turned off). You're now done - Take a look at the tutorial map if you don't quite understand.

4.8.2 Problems and Solutions with Mirrors

Map Name: NONE

- PROBLEM : My mirror doesn't show bullet holes when shot.
- SOLUTION : Make sure you are pressing the [1] key in 3D mode (to make a one-way mask) instead of the [M] key. Also make sure that the hitscan flag (press [H] on the mirror in 3D mode) is set.

4.8.3 Making the Two Way Mirror

Map Name: MIRRORS.MAP (ROOM #2)

This effect allows a 2 way mirror to be made (i.e. - you can go behind the mirror and look through it to the other side).

STEP 1: Create a mirror as described in the mirror how-to.

STEP 2: Create two very thin sectors that run parallel with the mirror (one on one side of the mirror, and one on the other side of the mirror). Make sure that these sectors are very thin, because they will contain the "Hall of Mirrors" effect, and you want this to be as unnoticeable as possible.

STEP 3: Create some way to access the area behind the mirror. Now you are finished with the construction. Take a look at the tutorial map if you don't quite understand.

Copyright © 1999 by Jonah Bishop. All rights reserved.



check out...

Unreal Tournament



world.com



4.9 Teleporters

[Previous Section](#)

[Table of Contents](#)

[Next Section](#)

4.9.1 Teleporters

Map Name: TUTORIAL.MAP (ROOM 4-C)

Teleporters are one of the most popular effects to construct, and are quite easy.

STEP 1: Create a sector that the player will start from. This will be known as the "source" sector.

STEP 2: Create another sector where the player will exit from. This will be known as the "destination" sector.

STEP 3: Place one ST1 sprite in both the source and destination sectors. The angle of these sprites is the angle in which the player will face upon teleporting to that particular sector. Give both sector effector sprites the same, unique hitag value, and a lotag value of 7.

NOTE 1: The source and destination sectors do not have to be the exact same size for this effect to work properly.

4.9.2 Sewer Holes & Silent Teleporters

Map Name: TUTORIAL.MAP (ROOM 4-D)

This effect is the exact same as the regular teleporting effect, with the only exception being that no sounds or animations are made upon teleporting from the source to the destination sectors. We will construct a square sewer hole for this example.

STEP 1: Create your source sector and lower the floor down to the point that you cannot see it when looking down (I'm assuming that you are starting on the upper level).

STEP 2: Create your destination sector and raise the ceiling up to the point that you cannot see it when looking up.

NOTE 1: The source and destination sectors should be **exactly** alike in order for this effect to look its best.

STEP 3: Place one sector effector in both the source and destination sectors. Give both sprites the same, unique hitag, and give them both a lotag value of 7. Lower the sector effector in the source sector down to the bottom of the hole (use [CTRL]+[PGDN] to do this quickly). Raise the sector effector in the destination sector to the top of the hole (this time use [CTRL]+[PGUP]).

NOTE 2: The angle of the sector effectors does not need to be changed. The player's angle upon entry of the source sector is the same when he exits the destination sector.

STEP 4: Raise the sector effector in the source sector up off of the ground. A few [PGUP] units will do fine (5 or 6 seem to work).

STEP 5: Lower the sector effector in the destination sector down from the ceiling. A few [PGDN] units will do fine (5 or 6 seem to work). You're finished! Take a look at the tutorial map if you don't quite understand.

4.9.3 One Way Teleporters (Version 1.4 or 1.5 Required)

Map Name: ATOMIC.MAP (ROOM #1)

One way teleporters are a new effect available in versions 1.4 and 1.5 of Duke Nukem 3D, and they are quite useful. Follow these steps to make this effect work properly:

STEP 1: Place one sector effector sprite in the sector that the player will enter *first* (the teleporting sector). Give this sprite a lotag value of 7 and a unique hitag value.

STEP 2: Place one sector effector sprite in the sector that the player will teleport to. Give this sprite a lotag value of 23, and a hitag value equal to the sector effector hitag in step 1. The angle of the sprite is the direction the player will face upon being teleported. If you do not want the flash or teleporting sound effects to play when teleported to this second sector effector, change the palette of the sprite to anything above 0. If you do want them to play, leave the palette as 0.

4.9.4 Version 1.3D One Way Teleporters

Map Name: TUTORIAL.MAP (ROOM 4-E)

This is a way to fool the game into making a one way teleporter in version 1.3d of the game.

STEP 1: Create your source teleporting sector. Place one sector effector sprite in this sector, giving it a lotag value of 7 and a unique hitag value.

STEP 2: Place one sector effector sprite where you want the player to exit. Give it the same unique hitag as the sector effector sprite in step 1, and give it a lotag value of 7. Then, lower this sprite into the ground until you can no longer see it. Take a look at the tutorial map if you do not understand.

Copyright © 1999 by Jonah Bishop. All rights reserved.



4.10 Ending The Level

[Previous Section](#)

[Table of Contents](#)

[Next Section](#)

4.10.1 Nuke Buttons

Map Name: NONE

The Nuke Button sprite (#142) is the sprite that you hit to end the current level. Simply place it on a wall and give it a lotag value of 32767. To link a Nuke Button sprite to a secret level, change the palette value of the sprite to 14, and change the lotag value to the level number you want to go to next (if the secret level is level number 9, you would make the lotag value equal to 9).

NOTE: Nuke buttons can also use lotag values of 65535 and 65534 to end the current level.

4.10.2 Ending Sectors

Map Name: NONE

This allows the player to end a level by walking onto a sector (you don't have to press the Nuke Button). Simply give the sector you want the player to end the level on a sector lotag value of 65535 (by pressing [T] on the sector in 2D mode).

4.10.3 Ending Sectors with Messages

Map Name: NONE

Remember in Episode 1, Level 3 where Duke gets trapped by the aliens and the message "We're gonna fry your ass Nukem!" appears? To create the same effect in your level, simply give the ending sector a lotag value of 65534. The same message as above will appear on screen. Also, give the sector a hitag value of the sound number that you wish to play (when the player walks on this sector).

Copyright © 1999 by Jonah Bishop. All rights reserved.

4.11 Switch Effects

[Previous Section](#)[Table of Contents](#)[Next Section](#)

4.11.1 Doors With Switches

Map Name: TUTORIAL.MAP (ROOM 4-F)

To make a door open by toggling a switch, follow these steps:

STEP 1: Make the door itself.

STEP 2: Place a switch sprite somewhere in your level, and give it a unique lotag value.

STEP 3: Place one activator sprite (#2) in the door sector and give it the same lotag value as the switch. If you only want the switch to work a fixed number of times, give the activator sprite a hitag value equal to the number of times you want the switch to work.

NOTE 1: It is strongly advised *not* to make this type of door an auto-closing door.

4.11.2 Combination Door Switches

Map Name: TUTORIAL.MAP (ROOM 4-G)

To make a door open only when a certain combination of switches is set, follow these steps:

STEP 1: Make the door itself.

STEP 2: Place one ActivatorLocked sprite in the door sector. Give it a unique lotag value.

STEP 3: Place a number of "dipswitch" sprites (sprites #162, #164, #166, and #168) somewhere in your level. Give each sprite the same lotag value you gave the ActivatorLocked sprite. Now give the sprites that need to be "on" in the game a hitag of 1. When these switches are turned to the "on position", the door will then open.

4.11.3 Another Version of Combo. Switches (Version 1.4 or 1.5 Required)

Map Name: ATOMIC.MAP (ROOM #2)

Another version of Combination switches is available in versions 1.4 and 1.5 of Duke Nukem 3D. The only thing that is different in this version is the sprite that you need to use for the switches. The new sprite is a red shooting target (#4359). Just substitute this sprite for the "dipswitch" sprites used in section 4.54. Then, set the angle of the sprite to the direction that you want the target to "lay down" (the angle should be opposite of the player's viewpoint in the game). Also, make sure that the sprites are flat (i.e. - press the [R] key on each sprite once).

NOTE: These sprites should *not* be on a wall, but should be sitting in a sector somewhere (where they can fall over). You can put them on a wall if you want, but it would look rather unrealistic.

4.11.4 Multi-Door Switches

Map Name: TUTORIAL.MAP (ROOM 4-H)

This is a useful effect for limiting which rooms a player can enter into at the same time. Just follow these simple steps:

STEP 1: Make 3 doors and give them appropriate textures and sounds. Make sure that you ***DO NOT*** place auto-closing sector effector sprites in any of these doors! This can cause some major problems.

STEP 2: Place the multi-switch sprite (#146) somewhere in your level, and give it a unique lotag value.

STEP 3: Place ***two*** activator sprites in each door sector (you should have six activator sprites in this effect altogether). You now must give these activator sprites the right tags. The following list should help you in tagging these doors properly.

NOTE 1: The letter X in the table below is the unique value you gave the multi-switch sprite in step 2.

NOTE 2: The first number below is the hitag value, and the second number is the lotag value.

	Door 1	Door 2	Door 3
Activator 1	(0,X)	(0,X+1)	(0,X+2)
Activator 2	(0,X+1)	(0,X+2)	(0,X+3)

Copyright © 1999 by Jonah Bishop. All rights reserved.



Click on me

and win **\$20,000**



4.12 C-9 Effects

[Previous Section](#)

[Table of Contents](#)

[Next Section](#)

4.12.1 C-9 Canisters (Visible)

Map Name: TUTORIAL.MAP (ROOM 5-A)

C-9 is what creates the majority of choreographed explosions in Duke Nukem 3D. To create an explosive can, simply place a sprite where you want the can to be. Change it to either the C-9 sprite (#1247), the OOFILTER sprite (#1079), or the exploding barrel sprite (#1238). If you want to place explosive underwater mines, place the BOUNCEMINE sprite (#940). If you own version 1.4 or 1.5 of Duke Nukem 3D, you can use the GUNPOWDERBARREL (#4360) as well.

4.12.2 Switch Operated Explosions

Map Name: NONE

This effect allows you to choreograph explosions and set them off via a switch.

STEP 1: Place a switch sprite somewhere in your level. This will be where you set off the explosion from. Give the switch sprite a unique lotag value.

STEP 2: Place one masterswitch sprite in *every* sector that you want an explosion to take place in. Give every single masterswitch sprite the same lotag value that you gave the switch. If you want to create a delay between flipping the switch and the first explosion, give the masterswitch a hitag value of the delay you wish to have.

STEP 3: Place C-9 sprites (#1247) everywhere you want an explosion to take place. You can raise the sprites to where you want, creating some nicely choreographed explosions.

NOTE 1: Placing a large number of C-9 sprites can cause your computer to slow drastically if you are running on a slower processor.

STEP 4: Shrink each C-9 sprite as far as you can width wise. This can be done by using the [5]+[4] key combination on the *numberpad*. When the sprites shrink no farther, you know that you are as small as you can get.

STEP 5: Give every C-9 sprite a hitag value equal to the lotag value that you gave the switch in step 1.

STEP 6: Now give each C-9 sprite any lotag value that you want. The lotag value for each C-9 sprite is the delay that the C-9 will wait before exploding.

NOTE 2: A lotag value of 0 on the C-9 sprites will cause these sprites to *not* explode. Make sure that the lotag value is greater than 0.

NOTE 3: A lotag value of 96 is roughly one second. Experiment to find out what is best for your situation.

4.12.3 Touchplate Activated Explosions

Map Name: TUTORIAL.MAP (ROOM 5-A)

To create touchplate activated explosions, follow the exact same procedure in the above section (switch operated explosions). The only change here is that you replace the switch sprite with a touchplate sprite.

4.12.4 Cracked Walls

Map Name: TUTORIAL.MAP (ROOM 5-B, HOLE #1)

This effect is what allows the player to launch some sort of explosive at a crack in a wall, generally forming a hole in the wall in which the player can walk through.

STEP 1: Create your "hole" sectors where you want the hole to be. They all should be valid player space. You can use as many sectors as you want, but the more sectors you use, the more work you will have to put into this construction. Make this hole look like you want it to look *after* the explosion has occurred. You can tilt the floor and ceiling as much as you want as well.

NOTE 1: The floor and ceiling should fit perfectly when lowered together (i.e. - the slope amount for each should be the *exact* same). This is not required, but it certainly makes the effect look better.

STEP 2: Place a crack sprite (#546, #547, #548, or #549) on the wall in front of the hole sectors. Give the crack sprite a unique hitag value.

NOTE 2: It is recommended to make the crack sprite slightly transparent. This can be done by pressing the [T] key on the sprite while in 3D mode.

STEP 3: Place one sector effector sprite in each hole sector. Give each sprite the same hitag value that you gave the crack sprite. Now give each a lotag value of 13.

STEP 4: Place C-9 sprites throughout your hole if you want to have explosions when the hole is blown open. If you do not wish to have any explosions take place, skip this step. Shrink each C-9 sprite as far as you can width wise (with the [4]+[5] key combination on the *numberpad*). When the sprites shrink no farther, you know that you are as small as you can get. Give each C-9 sprite the same hitag value that you gave the crack sprite. Now give each C-9 sprite a lotag value equal to the delay that you want the C-9 to have before exploding.

STEP 5: You are now completely finished! Take a look at the tutorial map if you do not fully understand.

4.12.5 Fixing The 'Texture Problem'

Map Name: TUTORIAL.MAP (ROOM 5-B, HOLE #2)

This is a common problem that occurs when making cracked walls and it can make a level look very ugly. The problem is that, when a cracked wall is blown up, the textures above and below the hole misalign. To fix this problem, simply press the [O] (the letter o, not the number zero) key on the misaligned walls in 3D mode. The problem will correct itself, and will be correct in the game. The tutorial map shows the effect with this problem, and without this problem. Note that no aligning is needed on the textures at all.

4.12.6 Making the Ceiling Blow Up (And Enemies Fall From the Sky!)

Map Name: TUTORIAL.MAP (ROOM 5-C)

First seen in Episode 1, Level 2, this effect is one of the coolest, and one of the most complex effects there is. When constructed correctly, this effect can look *quite* nice.

STEP 1: Construct the sector that will explode. Make it look as if the explosion has already taken place. The ceiling for this sector should have a parallaxed sky, and this texture *MUST* appear in at least one other place on the map. If it does not, this effect will not work correctly! If you are not using parallax in your level, simply parallax the ceiling of the sector you will make in step 3.

STEP 2: Place one Touchplate sprite in the sector that you want to trigger the explosion from. Do **not** place this touchplate in the sector that will explode. Give this touchplate sprite a hitag value of 1 and a unique lotag value.

STEP 3: Create a small sector in an area of your map that will never be accessed by the player (i.e. - this sector should be by itself in the corner of your map or somewhere of the sort). In this sector, place one C-9 sprite. Give this C-9 sprite a hitag value equal to the lotag value of the touchplate, and leave the lotag set as 0. Also place one masterswitch sprite in this sector. Give this masterswitch sprite a lotag value equal to the touchplate lotag value, and leave the hitag value set at 0.

STEP 4: Place 2 sector effector sprites in the sector that will explode. Give one of these sprites a lotag value of 33, and leave the hitag set at 0. Then give the other sprite a lotag value of 13 and a hitag equal to the touchplate lotag value. Raise the sprite with the lotag of 33 as far up as it goes (press [CTRL]+[PGUP] to do this). Raise the sprite with the lotag of 13 to the height of the ceiling of the surrounding sector. This part can be tricky, as you must make the sector effector with lotag of 13 **exactly** as high as the ceiling of the surrounding sector, or it won't look right.

NOTE 1: The angle for the sector effector sprite with the lotag of 13 should be straight down. If it is not, this effect will not work properly.

STEP 5: Place a few respawn sprites (#9) in the sector that will explode, giving each a lotag value equal to the touchplate lotag. Give each a hitag value equal to the sprite number that you wish to spawn. Raise each sprite up to the ceiling level.

STEP 6: Place one more sector effector sprite in the sector that will explode and give it a hitag equal to the touchplate lotag. Also give this sprite a lotag of 8. This simulates the effect of light coming in through the hole, so make sure that the surrounding sector shade and the shade of the exploding sector is darker than 0. Leave the shade of this sprite set at 0 (it isn't required to be left at zero - just as long as the shade is brighter than the surroundings).

STEP 7: Place some C-9 sprites in the sector that will explode. Give them a hitag equal to the touchplate lotag, and a lotag equal to the delay you wish them to wait before exploding. Raise these to any height that you want. Be sure to shrink them all the way width wise with the [4]+[5] key combination on the numberpad. You are now finished! Take a look at the tutorial map if you do not fully understand.

Copyright © 1999 by Jonah Bishop. All rights reserved.



Click on me

and win **\$20,000**



4.13 Sectors That Rise And Fall

[Previous Section](#)

[Table of Contents](#)

[Next Section](#)

4.13.1 Lifts

Map Name: TUTORIAL.MAP (ROOM 5-D)

Lifts are what make up most kinds of elevators, and are pretty useful when it comes to level designing.

STEP 1: Create your lift sector.

NOTE 1: The lift sector must have an adjacent sector that is higher than the lift in its lowest position. So when the lift is all the way down, there must be an adjacent sector whose floor is higher than the lift itself. If this is not the case, the lift will not work properly.

STEP 2: Give the lift sector one of the following sector lotag values (by pressing the [T] key while the cursor is over the sector in 2D mode):

- 16 - Starts in raised position & comes down. Stationary ceiling.
- 17 - Starts in lowered position & goes up. Stationary ceiling.
- 18 - Starts in raised position & comes down. Ceiling and floor move together.
- 19 - Starts in lowered position & goes up. Ceiling and floor move together.

STEP 3: For proper elevator sound effects, place one Music&SFX sprite (#5) in the lift sector. Give this sprite a lotag value of 71 and a hitag value of 73.

NOTE 2: Lifts can be slowed down by adding a GPSPEED (#10) sprite to the lift sector. The lotag value for this sprite is the speed, with higher values being faster.

NOTE 3: Lifts who have a white wall as one side of the lift, and who use sector tags 18 or 19, will cause the texture on the white wall to move with the lift. There is no way around this.

4.13.2 Switch Operated Lifts

Map Name: SWLIFT.MAP

Lifts can be operated by switches only, which can be quite useful in some cases. It is recommended to take a look at the tutorial map (thanks to Jack Gutzler AKA Captain Massive for this map) to see how things work reliably.

STEP 1: Construct your lift.

STEP 2: Place one activator sprite (#2) in the lift sector. Give this sprite a unique lotag value.

STEP 3: Place one switch somewhere near the elevator. Give this switch sprite the same exact lotag value that you gave the activator.

NOTE 1: Only a few switch sprites seem to work with this effect. Take a look at the tutorial map to see which ones work.

4.13.3 Locked Lifts

Map Name: TUTORIAL.MAP (ROOM 5-E)

Lifts can also be locked, creating some very cool possibilities for puzzles and tricks.

STEP 1: Construct your lift.

STEP 2: Place one activatorlocked sprite (#4) in your lift sector. Give this sprite a unique lotag value.

STEP 3: Place a keycard lock sprite somewhere in your level, and give it the same lotag value that you gave the activatorlocked sprite.

STEP 4: Set the palette of the keycard lock to your desired color.

4.13.4 Elevator Transports

Map Name: TUTORIAL.MAP (ROOM 5-F)

Elevator transports are a neat way to simulate true level over level, but they can be a pain to get just right. This effect simply warps you from one lift sector to another, making it appear that a true elevator has been used.

STEP 1: Construct your elevator sectors. Both sectors must be the **exact** size, shape, and the ceiling must be the **exact** distance from the floor in each sector. Also make sure that your texturing and shading is the same (to make this effect look realistic).

STEP 2: Place one sector effector sprite in each elevator sector, and place each sprite in the **exact** same spot of each elevator (the sprite in the first elevator should be in the same position in relation to the elevator sector as the sprite in the second elevator). Give both sprites a lotag value of 17, and give both sprites the same, unique hitag.

STEP 3: Change the shade value of the sector effector sprite in the starting elevator (the one that the player will enter first) to any value darker than the other sector effector sprite in the ending elevator. 64 is a good value.

STEP 4: Give both elevator sectors a sector lotag value of 15 (by pressing the [T] key in the sector in 2D mode). Now, give the sector that the player will end on a hitag value of 1. You are now finished. Take a look at the tutorial map if you still need some help.

4.13.5 Constructing Destructible Buildings

Map Name: TUTORIAL.MAP (ROOM 5-G)

This effect is used to create destructible buildings, much like the one in episode 1, level 2.

STEP 1: Create the building sectors (that will blow up) and make them look as if the building has already been destroyed. Place one sector effector sprite, one masterswitch sprite, and one gpspeed sprite in each sector that will be affected.

STEP 2: Give the gpspeed a lotag value of how fast you want the building to lower during its destruction (a value of 64 is very common). Give the sector effector sprite a lotag value of 13, and a unique hitag value. Then give the masterswitch sprite a lotag value equal to the sector effector hitag.

STEP 3: Place a switch or touchplate somewhere in your level. Give it a lotag value equal to the masterswitch lotag.

STEP 4: Place some C-9 sprites around your building, giving each a hitag value equal to the masterswitch lotag value. Then set a lotag value for each C-9 sprite (this value is the delay before the C-9 sprite will

explode). Remember to make the C-9 sprites as skinny as you can in the x direction (use the [4] key on the keypad).

NOTE 1: A masterswitch sprite should be in each sector that a C-9 sprite exists. If this is not the case, the C-9 sprites will not explode.

4.13.6 Crushers (Engine Pistons)

Map Name: TUTORIAL.MAP (ROOM 5-H)

This effect creates an engine-piston like sector (the ceiling constantly rises and falls to crush the player).

STEP 1: Create the crushing sector. The ceiling should be in the uppermost position that it will rise to.

STEP 2: Place one sector effector sprite in the crushing sector, and give it a lotag value of 25. The height of the sector effector determines where the crusher will start.

NOTE 3: If you want to slow down this effect, simply place one gpspeed sprite in the crushing sector and give it a lotag value (the speed that you want the crusher to go at).

4.13.7 Making the Ceiling Fall Upon an Explosion

Map Name: TUTORIAL.MAP (ROOM 5-I)

This effect is not used in the game, and there aren't too many good uses for this effect (it's simply an effect left over from very early versions of Duke Nukem 3D). This effect causes the ceiling to fall down when an explosion occurs in the sector.

STEP 1: Create the sector you want to be affected.

STEP 2: Place one sector effector sprite in the sector you just created. Give it a lotag value of 19.

4.13.8 Rotating & Rising Sectors

Map Name: TUTORIAL.MAP (ROOM 6-A)

This effect is used to make some very cool doors and other effects. When activated, this effect will cause the floor of a sector to rise up and rotate 90 degrees at the same time.

STEP 1: Create the sector to affect. This sector should be at the lowest point that the sector will ever be. Give this sector a sector lotag value of 30.

STEP 2: Place one activator sprite and one sector effector sprite in the sector. Give the activator sprite a unique lotag value. Give the sector effector sprite a hitag value equal to the activator sprite lotag value. Leave the lotag value of this sprite set at 0. Raise the sector effector sprite to the height you want the sector to rise to.

STEP 3: Place a switch or touchplate somewhere in your level. Give it a lotag value equal to the activator lotag.

STEP 4: Place one sector effector sprite where you want the rotation point of the affected sector to be (this point can be outside of the sector if you want it to be). Give this sector effector a lotag value of 1, and a hitag value equal to the activator lotag value.

4.13.9 Making Ceilings and Floors Drop Down

Map Name: TUTORIAL.MAP (ROOM 6-B)

This effect causes ceilings and floors to drop upon activation.

STEP 1: Create the sector to be affected. Give this sector a lotag value value of 28.

STEP 2: Place 3 sprites in the sector: one sector effector, one activator, and one gpspeed sprite. If you want to drop the floor, proceed to the floor heading. If you want to drop the ceiling, proceed to the ceiling heading.

Dropping Floors

STEP 3: Make the angle of the sector effector sprite point straight down. Give this sprite a lotag value of 21.

STEP 4: Give the activator sprite a unique lotag value. Give the gpspeed sprite a lotag value equal to the rate that you want the floor to fall at.

STEP 5: Make sure that the floor of this sector is at the lowest point it will be in the game. Then raise the sector effector sprite to the height that you want this floor to raise up to.

STEP 6: Place a switch or touchplate sprite somewhere in your level. Give this sprite a lotag value equal to the activator sprite lotag.

Dropping Ceilings

STEP 3: Make the angle of the sector effector sprite point straight up. Give this sprite a lotag value of 21.

STEP 4: Give the activator sprite a unique lotag value. Give the gpspeed sprite a lotag value equal to the rate that you want the ceiling to fall at.

STEP 5: Make sure that the ceiling of this sector is at the highest point it will be in the game. Then lower the sector effector sprite to the height that you want this ceiling to lower to.

STEP 6: Place a switch or touchplate sprite somewhere in your level. Give this sprite a lotag value equal to the activator sprite lotag.

4.13.10 Incremental Rising and Falling Sectors (Version 1.4 or 1.5 Required)

Map Name: ATOMIC (ROOM #3)

This effect is used most often in Duke-Tag levels for showing the team score incrementally. What this effect does, however, is raises or lowers a sector incrementally by flipping a switch several times (or toggling some other trigger in the level).

STEP 1: Create the sector that will rise or fall when activated, and choose appropriate textures.

STEP 2: Place one sector effector sprite in this sector. Give this sprite a lotag of 18 and a hitag equal to the amount of units moved per activation (one unit is equal to 1 [PGUP] or [PGDN] in Build). Now, set the angle for which part of the sector you want to move. Up makes the floor rise or fall, while down makes the ceiling rise or fall.

STEP 3: Place a switch (or other trigger) somewhere in your level, and give it a unique lotag.

STEP 4: Place one activator sprite in the rising (or falling) sector and give it the exact same lotag as the switch.

STEP 5: Now, you must set the palette of the sector effector sprite you made in step 2. A value of 0 starts the sector at the height you set it at in Build, and it rises or falls to the sector effector's height. Anything other than 0 starts the sector at the height of the sector effector and lowers or raises it to the adjacent sector's height. Set the height of the sector effector accordingly to the palette value (you'll have to play around with it to get it just right - this step can be quite tricky).



check out...

Unreal Tournament



epworld.com



4.14 Music And Sound Effects

[Previous Section](#)

[Table of Contents](#)

[Next Section](#)

4.14.1 Types of Sound Effects

Map Name: TUTORIAL.MAP (ROOMS 6-C, 6-D, and 6-E)

There are 4 types of sound effects that are available for use in level making. The first 3 types are covered in this section, while the fourth is explained in section 4.72. The 3 effects covered here all use the Music&SFX sprite (sprite #5).

Activation Sound (ROOM 6-C)

When used in a sector with a sector lotag, this sound will play when the sector's lotag function is activated (i.e. - A door opening).

- Lotag : Number of the starting (or main) sound you want to play.
- Hitag : Number of the ending (or secondary) sound you want to play.
- Placement : In any sector with a sector lotag.

Ambient Sound (ROOM 6-D)

When used in a sector without a sector lotag, an ambient sound will be played when the player walks near.

- Lotag : Number of the sound you want to play
- Hitag : Maximum distance that the sound can be heard. A hitag value of 1024 equals one of the largest grid squares in Build.
- Placement : Any sector without a sector lotag.

Echo Effect (ROOM 6-E)

When used in a sector without sector lotag, all sounds will have an echo effect.

- Lotag : 1000 + the amount of echo (from 0 to 255, with 0 being the least amount of echo)
- Hitag : Maximum distance that the sound can be heard. A hitag value of 1024 equals one of the largest grid squares in Build.
- Placement : Any sector without a sector lotag.

4.14.2 Adding Your Own Music and SFX

Map Name: NONE

Custom sounds and music can be added to Duke Nukem 3D, but there are several ways to go about it. All music is required to be in .mid (MIDI) format, and all sound effects are required to be in either .VOC (VOICE) or .WAV (WAVE) formats.

Renaming Custom Sounds

To add custom sounds or music, simply rename the custom music or sound effect to the exact same filename of an existing music or sound effect file. For example: If I wanted to add a sound of Homer

Simpson saying "Woo-Hoo!" to the game, I would first find a sound to write over (let's use Duke saying "Where is it?"). The filename for this sound is whrsit05.voc. I would rename my custom file from woohoo.voc to whrsit05.voc, and I would place this new sound in the main game directory. The sound **must** be in the main game directory, or this will not work.

Editing CON Files

Custom sounds and music can also be added by editing the USER.CON file. Please see the Con Editing FAQ for instructions on doing this.

4.14.3 Sector Activated Sounds

Map Name: TUTORIAL.MAP (ROOM 6-F)

This sound technique allows you to trigger a sound when Duke walks into a certain sector. An example of this would be from Episode 1, Level 3 where Duke enters the revolving room and hears : "Too late Nukem. We're in control now!".

To make a sector activated sound, simply give the sector you want to be the "trigger" sector, a lotag of 10+++. +++ is the sound number that will be played. The sound number for Duke saying "Damn!" is 214, so the sector would have a lotag of 10214 if you wanted to play this sound. Note that these types of sounds only play once.

4.14.4 Making Phones Ring

Map Name: TUTORIAL.MAP (ROOM 6-G)

Making telephones ring can be a trick, but it's pretty easy to do.

STEP 1: Make your telephone sector. This effect works best (in my opinion) when the telephone is a sector and not a sprite.

STEP 2: Place a sprite on the front of the telephone. Change this sprite into a switch (#132 works just fine) and resize the switch to match the size of the front of the phone. In 3D mode, press the [1] key while the mouse pointer is over this switch sprite. Then rotate the switch sprite (with the [<] and [>] keys) so that you cannot see the sprite (the angle should face the back of the phone).

NOTE 1: Dipswitch sprites don't seem to work, so use the SLOTDOOR type switches.

STEP 3: Give the switch a unique lotag value. The hitag value of this switch is the sound number that you want to play upon activating the phone (by pressing the space bar). The sound for a busy phone signal is a hitag value of 174, but you can use any sound you wish to use. Take a look at the tutorial map if you do not fully understand.



3 check out...

Quake 3 Arena 



4.15 Working With Sprites

[Previous Section](#)

[Table of Contents](#)

[Next Section](#)

4.15.1 Working With Sprites

Map Name: NONE

Sprites are the most used aspect (other than walls and sectors) in Duke Nukem 3D, and are very easy to master. To place a sprite in your map, press the [S] key (in either 2D or 3D mode). To change the sprite graphic, move the mouse over the desired sprite and press the [V] key twice. Find the desired sprite in the list, select it, and press the [ENTER] key. Sprites are used to make water fountains, cracks in walls, viewscreens, and more.

4.15.2 Security Cameras

Map Name: TUTORIAL.MAP (ROOM 6-H)

To make a security camera and corresponding viewscreen, follow these steps:

STEP 1: Place one sprite where you want the viewscreen to be. You can use either sprite #502 or sprite #499. Give the screen a unique hitag value.

STEP 2: Place several sprites where you want cameras to be. The camera sprite is #621. Give all the camera sprites that you want associated with the viewscreen created in step 1, a lotag value equal to the screen's hitag value.

NOTE 1: The angle of the camera sprite is the angle that the camera will point.

NOTE 2: To make cameras pan right and left, give them a hitag. Set the camera's angle to the center of the arc in which you want it to pan. Then set the hitag value equal to the build units of angle measure. A value of 256 will make the camera rotate 45 degrees to both the left and right of the sprite's angle.

NOTE 3: To change how far the camera looks either up or down, change the shade value of the camera itself (by pressing the [']+[S] keys on the camera sprite in 3D mode). A negative (bright) value will make the camera look down, and a positive (dark) value will make the camera look up. The brighter (smaller) or darker (larger) the value, the more the camera will either look down or up.

4.15.3 Vents

Map Name: TUTORIAL.MAP (ROOM 6-I)

To make a vent and vent cover, simply follow these steps:

STEP 1: Create the vent area sectors. Make this area accessible from a room in your level.

STEP 2: Place a sprite at the entrance of your vent area, and change it to sprite #595. In 3D mode, resize this sprite to properly fit the vent area entrance, and press the [B] and [H] keys on the sprite (if it already appears as a thick pink sprite in 2D mode, you need not press these keys). Press the [R] key on the sprite one time to make it sit "flat" on the wall.

4.15.4 "3D" Bridges

Map Name: TUTORIAL.MAP (ROOM 7-A)

A bridge is nothing but a group of sprites with the blocking and hitscan flags turned on. To make a bridge, simply place several sprites across the place you wish to span, size them to the desired length and width, and press the [B] and [H] keys on them in 3D mode (the sprites should be thick and pink in 2D mode).

To make a bridge fall down when shot, give all portions of the bridge (i.e. - all the sprites that make up the bridge) a unique hitag value.

4.15.5 Demo Cameras

Map Name: NONE

To make a camera that will play back demos after they are recorded, simply place one sector effector sprite where you want the camera. Give this sprite a lotag of 27, and a hitag value equal to how far the camera can "see". Experiment with this value to get it just right.

4.15.6 Spawning Stuff

Map Name: TUTORIAL.MAP (ROOM 7-B)

There are basically 3 ways to spawn something in the game.

The "CANWITHSOMETHING" Sprite

This sprite is a trash can that will spawn something when shot. Simply place a sprite where you want the trashcan to be, change it to the correct tile number (#1232), and give it a lotag value equal to the reference number of the sprite that you want to spawn. For example, if you wanted to spawn shotgun ammo, you would give the CANWITHSOMETHING sprite a lotag value of 49 (because the shotgun ammo is located at tile #49).

Spawn After Killing an Actor

This effect allows you to spawn something when an actor is killed by the player. A good example of this is the stripper. If you shoot her, you might want some enemies to pop out. To do this, place the actor (that will trigger the spawn) where you want it to be (for our example we will use the stripper). Place a respawn sprite (#9) where you want each monsters to appear. Give the actor sprite a unique hitag. Give the respawn sprite a lotag value equal to the dancer's hitag. Now give the respawn sprite a hitag value equal to the reference number of the sprite you wish to spawn. For example, if you wanted to spawn shotgun ammo, you would give the respawn sprite a hitag value of 49 (because the shotgun ammo is located at tile #49).

Trigger Spawns

You can spawn items by either toggling a switch, or by triggering a touchplate. To create this, place the switch or touchplate where you want it to be, and give it a unique lotag value. You can give it a hitag value if you want - the hitag value is how many times the respawn will work. Now place a respawn sprite where you want another sprite to spawn. Give it the same lotag value as the switch or touchplate. Give the respawn sprite a hitag value equal to the reference number of the sprite you want to spawn. For example, if you wanted to spawn shotgun ammo, you would give the respawn sprite a hitag value of 49 (because the shotgun ammo is located at tile #49).

4.15.7 Active Tripbombs

Map Name: NONE

To place an active tripbomb, simply place a sprite on the wall where you want the bomb to be, and change the sprite to #2566. Resize the sprite to a realistic size. Keep in mind that the bomb does not arm itself until it "sees" the player, so place it in a convenient spot.

4.15.8 Making the Mini-Boss

Map Name: NONE

The mini-boss is simply the episode end boss, but when killed, his death doesn't start the end sequence. Plus, he's a lot easier to beat in "mini-boss" status. To change an end boss to a mini-boss, simply place the boss sprite where you want it, and give it a palette value of 21. To spawn a mini-boss, place a respawn sprite where you want it to spawn, give it a hitag value of 2630, a lotag value equal to the lotag of the switch or touchplate that toggles it, and a palette value of 21.

NOTE #1: The second and third episode bosses cannot be made into mini-bosses without hacking the con files. 3D Realms mistakenly gave them only 1 health point when their palette value is anything but 0. If you do make them mini-bosses, realize that they will be extremely easy to kill.

4.15.9 Exploding Helicopter & Car

Map Name: NONE

To create the exploding helicopter or car, simply place a sprite where you want it to originate from (#1346 for the helicopter and #2491 for the space car). Set the angle of the sprite to the direction you want the car to travel in, and raise the sprite up as high as you want in 3D mode. It should be noted that you must leave enough room for the car or helicopter to travel through - if you don't, it will slide down the first wall it hits, making a very unrealistic effect.

4.15.10 Cranes

Map Name: TUTORIAL.MAP (ROOM 7-C)

Cranes allow the player (and other sprites) to be picked up and carried from one place to another by a crane sprite.

STEP 1: Make a sector that the crane will travel down. Place one sprite in this sector, and change it to sprite #1222. Raise this sprite all the way up to the ceiling. This sprite is the crane itself and **should not** be resized - leave it the default size.

STEP 2: Make one smaller sector at the end of the sector you just created. This is where the player will be picked up. Place one sprite in this sector, and change it to sprite #1221. Make sure that this sprite stays at ground level (don't raise it off the ground). You're finished! Take a look at the tutorial map if you don't understand.

NOTE 1: If you have more than one crane in your level, you need to give both the sprite created in step 1, and the sprite created in step 2, the same unique hitag. Do so for each crane you make in your level.



check out...

Unreal Tournament



world.com



4.16 Multiplayer Level Making

[Previous Section](#)

[Table of Contents](#)

[Next Section](#)

4.16.1 Dukematch & Co-op Starting Positions

Map Name: NONE

To create co-op and Dukematch starting points, place the initial frame of the Duke Nukem sprite (#1405) where you want the start point to be. This is all you need to do to make a Dukematch starting point. Co-op start positions should all be located at the beginning of the level (but they are not required to be), and should have a lotag of 1. A minimum of 7 start positions is required for 8 player co-operative and Dukematch games. The player 1 start position (brown arrow in 2D mode) does count as a start position.

4.16.2 Making Items Appear Only In Multiplayer

Map Name: NONE

To make a sprite appear only in Dukematch or Co-op games, give the desired sprite a palette value of 1. This is effective for making weapons and other sprites you want available in multiplayer, not available to the player in single player mode. Switches to certain doors are one possibility for this effect (this possibility appears in most of the levels made by 3D Realms).

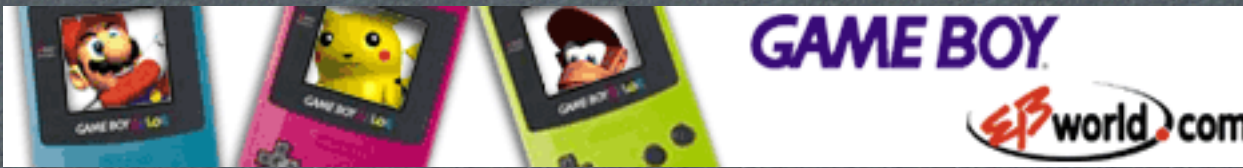
4.16.3 Making Duke TAG Levels (Version 1.4 or 1.5 Required)

Map Name: NONE

All the specs for making Duke TAG levels can be found in the file "buildtag.txt". This file can be found in the Build directory on your game CD.

The effects used for Duke TAG levels are incremental rising and falling sectors (for the scoreboards) and one-way teleporters for cross-level transportation.

Copyright © 1999 by Jonah Bishop. All rights reserved.



4.17 Miscellaneous

[Previous Section](#)

[Table of Contents](#)

[Next Section](#)

4.17.1 Glass

Map Name: TUTORIAL.MAP (ROOM 7-D)

Glass is basically a masked wall that has the transparency flag turned on. You can use any texture you want to, although this how-to will guide you through making "breakable" glass.

STEP 1: Create a masked wall where you want the glass window to be.

NOTE 1: This mask should not be on a wall where the player can see the wall itself, and where the wall has two properties (one for the top part and one for the bottom). Window 1 in the tutorial map shows this effect - notice how the wall changes when shot. You should create "buffer sectors" to make this problem non-visible. Windows #2-#4 all have these "buffer sectors", so look at them if you don't understand what I'm talking about.

STEP 2: Change the texture of this mask to the breakable glass texture, which is texture #503.

STEP 3: While in 3D mode, press the [B] and [H] keys on the masked wall. This makes the masked wall block you and other characters from walking through the wall, and it makes bullets "hit" this wall.

STEP 4: Now point at the mask in 3D mode and press the [T] key. The mask will now become transparent. Press the [T] key again. The glass will be even more transparent. There are 3 levels of transparency, and they are as follows:

- First - No transparency (try a regular wall texture to see)
- Second - Slight transparency
- Third - Most transparency

Set the transparency level to the setting that you desire. In the tutorial map, window 2 shows the first level of transparency, window 3 shows the second level of transparency, and windows 1 and 4 show the third level of transparency.

4.17.2 Secret Areas

Map Name: NONE

To make a secret area, give the desired sector a lotag of 32767.

4.17.3 Shooters

Map Name: TUTORIAL.MAP (ROOM 7-E)

This section covers how to make "shooters". A shooter is basically something that repeatedly shoots an object of your choice. A good example of a shooter is the one located in episode 1, level 4. When you unlock the red keypad, the pad lowers, revealing a shooter that shoots "shrinkers" (the projectiles that make you shrink in size).

STEP 1: Create a sector from which the shooter will shoot from.

STEP 2: In this newly created sector, place one sector effector sprite, one gpspeed sprite, and one masterswitch sprite.

STEP 3: Place a switch somewhere in your level and give it a unique lotag.

STEP 4: Give the masterswitch sprite the same lotag value as the switch. Also give the masterswitch sprite a hitag of 50.

STEP 5: Place one sector effector sprite where you want the shots to originate from. Change the angle of this sprite to the direction you want the shooter to shoot. Give this sprite a lotag of 36.

STEP 6: Now you must decide what you want to shoot. Do this by giving the gpspeed sprite a lotag of one of the following values:

- 1360 = Octabrain Blasts
- 1625 = Alien Lasers
- 1636 = Lizard Spit
- 1641 = Freezer Blasts (Auto-aims at player)
- 1650 = Mortars (Like the first boss throws)
- 2605 = Rockets (Auto-aims at player)
- 2556 = Shrinkers

4.17.4 Turning Shooters Off

Map Name: NONE

The following steps explain how to construct a shooter that can be turned off after it has been triggered.

STEP 1: Create a sector from which the shooter will shoot from.

STEP 2: In this newly created sector, place one sector effector sprite, one gpspeed sprite, and one activator sprite.

STEP 3: Place a dipswitch sprite somewhere in your level and give it a unique lotag. It **must** be a dipswitch sprite, or this effect will not work correctly.

STEP 4: Give the activator sprite the same lotag value as the dipswitch. Also give the activator sprite a hitag of 50.

STEP 5: Place one sector effector sprite where you want the shots to originate from. Change the angle of this sprite to the direction you want the shooter to shoot. Give this sprite a lotag of 36.

STEP 6: Now you must decide what you want to shoot. Do this by giving the gpspeed sprite a lotag of one of the following values:

- 1360 = Octabrain Blasts
- 1625 = Alien Lasers
- 1636 = Lizard Spit
- 1641 = Freezer Blasts (Auto-aims at player)
- 1650 = Mortars (Like the first boss throws)
- 2605 = Rockets (Auto-aims at player)
- 2556 = Shrinkers

4.17.5 Difficulty Settings

Map Name: NONE

To set the difficulty setting for a monster, simply give the desired monster one of the following values:

- 2 = Skill Level 2
- 3 = Skill Levels 3 and 4 (Skill 4 just makes the monster respawn)

4.17.6 Making Signs Fall When Shot

Map Name: NONE

To make a sprite sign (or other sprite) fall to the ground when hit by an explosion, simply give the sprite any hitag other than 0.

4.17.7 Lightning and Thunder (Version 1.4 or 1.5 Required)

Map Name: ATOMIC.MAP (ROOM #4)

To make the lightning bolt engine work, follow these simple steps:

STEP 1: Place several lightning sprites (#4890) in an outside area, and give each sprite *the same* unique hitag. The sky should also be cloudy (not required, but makes it look realistic).

STEP 2: Place a sector effector in each sector that contains a lightning bolt sprite. Give each sector effector a lotag of 28, and a hitag equal to the lightning bolt sprite's hitag value.

4.17.8 Making Rain

Map Name: NONE

To make a raining effect, place dripping sprites (#660) where you want them to be, and raise them to the ceiling of the sector they are in. If you own version 1.4 or 1.5, place a music and sound effects sprite somewhere in the raining sectors, give it a lotag of 358, and a hitag of the distance the sound should be heard.

NOTE: Giving the rain drops a palette number of 13 is recommended for both versions of the game.

4.17.9 Forcing Duke To Start Weaponless

Map Name: NONE

To make Duke start a level without any weapons or ammunition, simply place the HURTRAIL (#859) or other damaging floor tile under the starting position.



Click on me

and win **\$20,000**



Section 5 - General Information

[Previous Section](#)

[Table of Contents](#)

[Next Section](#)

5.1 Palette List

The following is a listing of all the palette numbers and their corresponding uses. Palettes can effect 3 different 'layers' of a tile. The easiest way to understand these 3 different 'layers' is by using the stripper sprite. The 3 'layers' are: Background (her skin), Foreground (her clothes), and Minor Foreground (money, accessories, etc).

- 0 - Normal
- 1 - Makes a sprite Multiplayer only. Also makes all wall, ceiling, and floor textures change to blue values.
- 2 - All 3 layers are red. Makes all wall, ceiling, and floor textures change to red values as well.
- 3 - Changes nothing on sprites. If used on a sector effector with a lotag value of 7, the water you jump into will turn into slime (a green tint). Using this value also prevents the player from being killed upon entering a sector with the BIGORBIT texture on either the floor, ceiling, or both.
- 4 - All 3 layers are solid black.
- 5 - All red colors (on all 3 layers) become brown.
- 6 - Night Vision Green (Used by the game to designate night vision. I wouldn't recommend using this in a regular level).
- 7 - All 3 layers are yellow-green. Makes all wall, ceiling, and floor textures change to yellow values.
- 8 - All 3 layers are light green.
- 9 - Normal
- 10 - Foreground is red.
- 11 - Foreground is light green.
- 12 - Foreground is light gray.
- 13 - Foreground is dark gray.
- 14 - Foreground is dark green.
- 15 - Foreground is brown.
- 16 - Foreground is dark blue.
- 17 - Background is blue, Foreground is green, Minor Foreground is blue.
- 18 - Exact same as 11.
- 19 - Background is red, Foreground is gray, Minor Foreground is red.
- 20 - Background is blue, Foreground is gray, Minor Foreground is blue.
- 21 - Foreground is red. This palette can be used to turn a regular boss into a mini-boss (the level doesn't end when the boss is killed). It also can be used to make an assault trooper an assault captain.
- 22 - Foreground is light gray.
- 23 - Foreground is yellow.

- 24 - Some shades of gray and white change to corresponding red shades.
- 25 - Minor Foreground is red.

5.2 Facts On Dimensions

This section is some useful dimensions to use in Building levels. The units of measurement used here are the number of times you should press the [PGUP] or [PGDN] keys.

Measurement	Standing	Ducking
Minimum sector height that can be entered	11*	6
Maximum sector height that can be walked on	3	0
Maximum height of floor to shoot above	8(9)**	3(4)**
Minimum height of ceiling to shoot below	10(9)**	5(4)**
Minimum height of ceiling to shoot below with RPG	11	6
Maximum height of floor to jump onto	20	0
Minimum height of wall to place tripbomb	10	4

NOTE:

- * Duke can also enter sectors with a height of 10, but he will automatically duck.
- ** The second number is provided for when only some of the shots actually get past the floor or ceiling.

5.3 Sector Effector Lotag Listing

- 0: Rotating sector
- 1: Center point for above
- 2: Earthquake
- 3: Random light effect after light is shot out
- 4: Random light effect
- 5: NONE
- 6: Subway Engine (Pulls other cars)
- 7: Transporter
- 8: Door opens - room lights up
- 9: Ceiling light lowers - room lights up
- 10: Door auto close (set Hitag to delay)
- 11: Swinging Door
- 12: Light switch
- 13: Used to make holes in walls
- 14: Subway car
- 15: Sliding door (ST 25)
- 16: NONE
- 17: Elevator transporter (ST 15)
- 18: Incremental sector rise/fall (PPAK only)
- 19: Lower ceiling when explosion hits sector
- 20: Stretch sector (ST 27)
- 21: Lower floor or ceiling (ST 28)

- 22: Prong for teeth doors
- 23: One way teleporter exit (PPAK only)
- 24: Conveyor belt or water current
- 25: Engine Piston (moves up and down)
- 26: NONE
- 27: Demo Camera Location
- 28: Lightning Bolt Engine (PPAK only)
- 29: Used to make waves
- 30: 2 way train (ST 31)
- 31: Used to raise or lower the floor
- 32: Used to raise or lower the ceiling
- 33: Earthquake Debris
- 36: Used for Shooters

5.4 Sector Lotag Listing

- 1: Above water
- 2: Under water
- 9: Sliding Star Trek door
- 15: Elevator transport (SE 17)
- 16: Elevator platform down
- 17: Elevator platform up
- 18: Elevator down
- 19: Elevator up
- 20: DOOM door
- 21: Opposite of DOOM door
- 22: Split from middle door
- 23: Swinging door
- 25: Sliding door
- 26: Splitting Star Trek door
- 27: Stretch sector (SE 21)
- 28: Drop floor (SE 21)
- 29: Teeth door prong (SE 22)
- 30: Rotate-rise sector
- 31: 2-way train
- 10+++: Play sound when player walks on sector (+++ is sound #)
- 32767: Secret room
- 65535: End level

5.5 List of Official Levels

The following is a list of the all the official levels found in Duke Nukem 3D, and the level's repsective designer.

- AB - Allen H Blum III

- LL - Richard Bailey Gray (The Levelord)
- RP - Randall Pitchford II
- KS - Keith Schuler
- GB - George Broussard

Episode 1:

- 1. AB - Hollywood Holocaust
- 2. AB - Red Light District
- 3. AB - Death Row
- 4. AB - Toxic Dump
- 5. LL - The Abyss
- 6. AB - Launch Facility
- 7. AB - Faces of Death (Dukematch Only)
- 8. AB - Dukematch Level 1 (Dukematch Only)

Episode 2:

- 1. AB - Moonbase
- 2. LL - Incubator
- 3. LL - Warp Factor
- 4. AB - Fusion Station
- 5. AB - Occupied Territory
- 6. AB - Tiberius Station
- 7. AB - Lunar Reactor
- 8. AB - Dark Side
- 9. AB - Overlord
- 10. LL - Spin Cycle
- 11. LL - Lunatic Fringe

Episode 3:

- 1. LL - Raw Meat
- 2. LL - Bank Roll
- 3. LL - Flood Zone
- 4. LL - L.A. Rumble
- 5. LL - Movie Set
- 6. LL - Rabid Transit
- 7. LL - Fahrenheit
- 8. LL - Hotel Hell
- 9. AB - Stadium
- 10. LL - Tier Drops
- 11. AB - Freeway

Episode 4:

- 1. RP - It's Impossible

- 2. AB - Duke Burger
- 3. GB - Shop N Bag
- 4. AB - Babe Land
- 5. RP - Pigsty
- 6. AB - Going Postal
- 7. KS - XXX Stacy
- 8. KS - Critical Mass
- 9. AB - Derelict
- 10. RP - The Queen
- 11. RP - Area 51

Other:

- AB - _se.map (Sector Effector Demo Map)
- AB - _st.map (Sector Tag Demo Map)
- AB - _zoo.map (Zoo Map)

5.6 List of Cheats

The following is a listing of all the cheat codes available in the game:

- DNCORNHOLIO - Toggles god mode
- DNKROZ - Also toggles god mode
- DNSTUFF - Gives you all weapons, inventory, and keys
- DNITEMS - Gives you all keys, inventory, and full armor
- DNINVENTORY - Gives you all inventory
- DNWEAPONS - Gives you all weapons
- DNKEYS - Gives you all keys
- DNHYPER - Gives you steroids
- DNSCOTTY#### - Warps to a certain level. The first # sign is the episode number, while the second and third # signs are the level number. DNSCOTTY203 would take you to episode 2, level 3.
- DNMONSTERS - Toggles monsters
- DNCASHMAN - When the spacebar is pressed, money is thrown out
- DNVIEW - Same as hitting F7 (also displays message saying "You're too good to be cheating!")
- DNRATE - Shows the Frames per second (FPS)
- DNDEBUG - Shows some debugging information
- DNSKILL# - Changes the skill level to whatever you input for #
- DNBETA - Displays the message "Pirates Suck!"
- DNTODD - Displays the message "Register Cosmo"
- DNALLEN - Displays the message "Buy Major Stryker"
- DNCOORDS - Displays your coordinates
- DNSHOWMAP - Displays the entire map
- DNCLIP - Toggles Clipping Mode (if you walk into empty space, you die). In versions 1.4/1.5, walking into empty space will not kill you.
- DNUNLOCK - Unlocks all doors and opens those that are not locked.

5.7 List of Secret Messages in the Game

This section has been included as an "extra" (it has nothing to do with editing, but is rather interesting information). The following messages are hidden in their respective levels. Can you find all of them?

Episode 1

Level 1

867-5309

Stryker Metronet.com

Level 3

Behind the babe

Level 4

How did you get here?

Level 5

Go with the flow

Dopefish Lives

You're not supposed to be here - Levelord

SUYT (Acronym for "Show Us Your Tits")

Episode 2

Level 1

U.S.S. Framerate (Hint: Can only be seen in Build)

Level 2

1138 (Can be found in 2 places)

Level 3

Blum Bed Co.

Level 7

Bite Me!

Level 8

Jump in the monolith

Episode 3

Level 1

Push her button

Push Me

No one should be here - Levelord

Level 3

The crack below

Annal Oothrick (See NOTE #1 below)

Level 4

Under the Knife

You're not supposed to be here... either - Levelord

Level 5

Push the register not the snacks

Jump at the Earth

Level 6

Telltale Footprints

Level 7

Blast the bottles

Level 8

Check the Vintage

Behind the Fountain

Duke was here 2017 (See NOTE #2 below)

Episode 4

Level 1

Are you small or are you cheating?

Looking for secrets? - DUV

C'mon, Quit Cheating - DUV

Level 3

Run your ass off

Special Notes

NOTE #1 : Here is the meaning of the message "Annal Ooothrick" straight from Levelord himself: "...not sure of the spelling, but that is a tribute to the movie Excalibur. I believe it's Gaelic for dragon's breath and was a tie-in to the steam being emitted at the message."

NOTE #2 : This message can be found in the game, but is unreadable due to the fact that all the letters are jumbled together (they were never raised up to where they should have been). Can you find this message?

Copyright © 1999 by Jonah Bishop. All rights reserved.



NBALive
for PSX **2000**



ep world .com



Section 6 - The Most Common Questions And Answers

[Previous Section](#)

[Table of Contents](#)

[Next Section](#)

This section is a set of the most commonly asked questions, along with the answers to these questions. Browse through this list, and if you find your question, read the answer! :-)

6.1 I can't find Build on my CD - where is it?

The Build program is located in the \goodies\build directory. If your CD does not have a \goodies\build directory, read this note written by Joe Siegler of 3D Realms:

If you bought Duke Nukem 3D, and it does not have a \goodies\build directory, please be aware that this was a misprint at the distributor level. You need to contact the distributor, GT Interactive, for information on how to get a replacement. Apogee/3D Realms will only replace CD's bought DIRECTLY FROM US. You can contact GT at support@gtinteractive.com. Alternately, you can call 1-732-248-1622 Extension 800 for the Missing Build Directory issue only.

6.2 I can't get Build to work. How do I do it?

Make sure that Build is in the *same* directory as the game itself. Next, make sure that the read only properties of all the files in this main directory are turned off. You can do this by entering DOS (or a DOS window from Windows 95) and typing the following line while in the Duke Nukem 3D

directory:

```
attrib -r *.*
```

6.3 How do I enter 3D mode?

Press the [ENTER] key on your numeric keypad. This key is located on the far right of the keyboard, under the [+] key (on most keyboards at least).

6.4 3D Mode is screwed up! What do I do to fix it?

Read the answer to question 6.2 first. If that doesn't work, run the program BSETUP, changing *only* the video mode. Then select "Save Properties and Exit" from the menu.

6.5 How can I play my own maps?

Type in either of the following lines in DOS:

```
duke3d -map foo.map
```

```
duke3d -map foo
```

In the line above, foo is the name of your map. The .map extension is not needed, but can be used.

6.6 When I run my level, I get an error. What does it mean?

Take a look at section 7 for the meanings to various Build errors.

6.7 Why does the game crash when I run my level?

Make sure that you have placed a start position in your level (by pressing the [SCROLL LOCK] key in 2D mode). If you already have a start position, see if an error is being generated upon the game crashing. If an error does exist, check section 7 of this FAQ for more information on the error.

6.8 Build sucks. Is there another editor available?

No, Build is the only editor available. It's not that hard to learn, so stick with it. A level editor other than Build is available for Macintosh users, but I don't use a Mac, so I can't give you much information on it anyway.

6.9 Walking under a parallaxed sky kills me. Why is this?

First, the texture you are using is the BIGORBIT1 texture. How do I know this you ask? Because it's the only texture that can kill you when parallaxed. To fix this problem, simply change the palette of this texture to 3 by using the [ALT]+[P] key combination in 3D mode. Walking under it will not kill you.

6.10 My subway shoots rockets at me - Why?

Subway sectors that are under a parallaxed sky will shoot rockets (much like the orbiting ship in Episode 2, Level 1). To fix this problem, simply change the palette of the parallaxed texture to 3 by using the [ALT]+[P] key combination in 3D mode (or you can un-parallax the texture).

Copyright © 1999 by Jonah Bishop. All rights reserved.



Section 7 - Common Build and Runtime Errors

[Previous Section](#)

[Table of Contents](#)

[Next Section](#)

7.1 Found Lonely Sector Effector

This error occurs when a sector effector sprite is found with no hitag and lotag value (i.e. - both the hitag and lotag is 0). To fix this, copy down the coordinates that the error gives you and find those coordinates in your map in Build. Accordingly delete or tag the sector effector in question to correct this error.

7.2 Subway Found No Zero'd Sectors With Locators

This is a very complex error and will require some explaining. This error usually occurs in two-way trains however, I have seen it in regular subways built in certain maps. For this explanation, we will look at the problem as if it were occurring in a two-way train.

The problem is this: the sector effector in the two-way train is not in the correct sector. So which sector should it be in? Well, you cannot split a two-way train sector into multiple sectors. And you cannot nest sectors in a two-way train (there is a "complex" way of nesting sectors in a train - see the two-way train how-to in this FAQ for more information). In other words, the train itself should be one sector, and the player should be in this sector at *all* times while riding the train. So if you can't nest or split sectors, how did the level designers at 3D Realms make the window sectors on the two-way trains found in Episode 2? Well, that's a complex method of using non-valid sectors and nesting sectors between the two. Sound complicated? It is. But it can be done, and done very well. Check the two-way train how-to to see exactly how to do this.

7.3 Can't Make a Sector Out There

This problem sometimes occurs when trying to create a sector inside of another sector (and using the [ALT]+[S] keys to make it valid player space). The solution to this problem is simple. Delete the sectors in question, and *always* build from the innermost sector outwards. Building from the outside to the inside usually ends up in this error.

This error also appears when trying to make a valid player space sector outside of the map. The solution for this case is simply this: Don't try to create valid player space sectors (red lined sectors) outside of the map.



Click on me

and win **\$20,000**



Section 8 - Utilities for Duke Nukem 3D

[Previous Section](#)

[Table of Contents](#)

[Next Section](#)

8.1 Wad2Map

This little utility, found in the \goodies\wad2map directory on your game CD, will convert Doom maps into Duke Nukem 3D maps. The enemies, textures, tagging, etc. are changed, but when I used this utility, I had to redo almost everything by hand. It can save some time in the Building process, however.

- Usage: WAD2MAP <DOOM.WAD path> <level to change> DUKE.TXT
- Usage Example: WAD2MAP c:\games\doom\doom.wad E1M1 DUKE.TXT

The <DOOM.WAD path> is the full path to the DOOM.WAD file on your computer, the level to change is the level that you want to change over, and the script file to use is always duke.txt (you can make your own if you want, however). The script file duke.txt can be found in the \goodies\wad2map directory on your game CD.

8.2 Wad2Art

This utility, found in the \goodies\wad2map directory on your game CD, converts the original Doom art to Duke Nukem 3D art. All of the original Doom art tiles are placed into one, large tiles000.art file.

NOTE: Before using this utility, backup the names.h file and the palette.dat file, because wad2art will overwrite them.

- Usage: WAD2ART <DOOM.WAD path>
- Usage Example: WAD2ART c:\games\doom\doom.wad

The <DOOM.WAD path> is the full path to the DOOM.WAD file on your computer. Three files will be created in your game directory (or the directory in which wad2art is located):

- Tiles000.art --- A large file containing all the Doom tiles
- Names.h --- The names of all the Doom tiles
- Palette.dat --- The original Doom palette

8.3 Kextract

Kextract is a little program that was written by Ken Silverman (author of the Build engine). This program can extract files from any valid .grp file, including the duke3d.grp file (which contains all the maps, art files, sounds, music, and other files for the game itself).

NOTE 1: This program *must* be in the same directory as the .grp file you wish to extract files from.

NOTE 2: Extracting all files from the duke3d.grp file will take up a large amount of disk space, and will create over 2000 files in your directory. This is possible to do, but use caution when doing this.

- Usage: kextract [grouped file] [files to extract]

[grouped file] is the name of the file that you wish to extract from. The duke3d.grp file is the file that contains all the game art, levels, sounds, etc.

[files to extract] is simply the files that you wish to extract. Just sequentially list the files you wish to extract, placing a space between each file. Wildcard symbols (*) can be used to extract all of one type of a file (i.e. - *.art).

8.4 Kgroup

Kgroup is another little program written by Ken Silverman (author of the Build engine), and is used to compile multiple files into one single group file (.grp) that you specify.

- Usage: kgroup [grouped file] [files to group]

[grouped file] is the name of the file that you wish to group to. Do not use duke3d.grp here, unless you want to reinstall the game, or unless you have a copy of the .grp file in your directory.

[files to group] is simply that: the files you wish to group. Just list the files you wish to group, placing a space between each file. Wildcard symbols (*) can be used to group all of one type of a file (i.e. - *.art).

8.5 Group File Studio

Group File Studio is a utility for Windows 95 and Windows 98 that makes working with group files much simpler than ever before. At the time of this writing, the available features of this program are:

- Create new group files
- Insert files into an existing group file
- Permanently remove files from a group file
- Extract a single file, a file type (i.e - *.map), or all files from a group file
- Rename embedded files
- Play embedded sound (*.voc) files directly from the group file
- Auto-Play option for automatically playing sound files
- Play embedded music (*.mid) files directly from the group file
- View the attributes of the embedded art tiles (located in .art files)
- View art tiles from the embedded art files
- Specify a default names.h file for showing the associated names when viewing the artwork
- Specify a custom palette.dat file to use for viewing art tiles
- Specify a default directory to extract files to
- HTML based help

If you find kextract and kgroup to be difficult and cumbersome to use, give Group File Studio a try. You can download this utility at Map FAQ Central or Dukeworld.

Section 9 - Finishing Up

[Previous Section](#)[Table of Contents](#)[Next Section](#)

9.1 Map Authoring Template

The map authoring template is a small text file that tells others about your map; what the story line is (if any), whether there is new art, music, SFX, or effects (by editing the .CON files), and other useful information. One has been included in the zip file that this FAQ comes in, under the name template.txt. Fill this out before sending your level to any site for uploading - it's very useful!

9.2 Uploading Your Level

After your level is complete (and mostly bug free), you need to upload it so that people can play it! Upload your level to the Dukeworld user level archives. The URL to Dukeworld is:

<http://www.dukeworld.com>

Information on uploading files can be found there. Have fun!

Copyright © 1999 by Jonah Bishop. All rights reserved.