DoomCAD

The state of the art Doom Level Creation System

DoomCAD is a powerful Drag-N-Drop style editor for id Software's Tour de Force game, DOOM. DoomCAD allows you to create new levels for DOOM, and preview them in **THREE DIMENSIONS!**

This software is in NO WAY connected to id Software. It is intended for the personal creation of Doom levels ONLY. You have been warned! id Software cannot and will not provide any support for this product, nor will they provide support to DOOM itself if it is using data files created by this product.

What you need to know first

Doom levels are a complicated set of structures. It would be nice if the DoomCAD user didn't need to know ANYTHING about the underlying structure of a Doom level, but this is unfortunately not possible. It is STRONGLY recommended that the user acquaint himself with the various structures before trying to create a new level.

There are many ways to do this, the best being Matt Fell's fabulous work, "The Unoffical DOOM WAD file specs". The dedication that Matt has put into this work is staggering, especially in light of the fact that he has NO plans for using the specs to create a level editor himself! This exhaustive treatise has become the Doom level editor programmers' bible, having a byte-for-byte description of each structure in the WAD file.

As the Doom level creator, you nned not concern yourself with a byte-by-byte explanation of DOOM.WAD, but you WILL need to know what each structure is, and how they relate. How do I add a new monster? A new weapon? How do I create a teleport pad? A staircase? A crushing ceiling?
By acquainting yourself with the Wad file 'logo', we can speak to each other using the same language. Questions you ask can be more easily answered by other WAD file creators, or the authors of the level editors themselves.

The MINIMUM structures needed to know what makes a Doom level tick, and how they interact, are in the following sections. More information will be given for each structure when talking about editing that structure.

Things

A Thing is ANY object in the game that can be interacted with, as well as various things used as 'scenery'. The Things can be broken into the following categories:

Monsters
Player and Deathmatch Start Positions
Weapons and Ammo
"Scenery" (torches, barrels, various bloody corpses.
Goodies (RadSuits, Armor, Light goggles)
KeyCards and Skull Keys

A thing can be defined to appear on only certain difficulty levels. In addition, the monsters can be set to facing one of 8 directions at the game's start.

Verticies

(or VERTEXES)

A Vertex is a point in space where walls meet. It has an x and a y coordinate, but not a z, or 'up' coordinate. Doom looks like it is a truly '3D' game, but if you study it carefully, the levels are still really a modified '2D'. Nothing in the game is ever 'above' or 'below' anything else. That is, by looking at a plan view of a level, ALL areas of the map are visible, nothing is ever obstructed by being under something else.

LineDefs

A LineDef connects two verticies together. This line can then be a wall in the game, althougth not all LineDefs are walls. LineDefs can also be 'transparent', or not visible to the player. A LineDef can have either one or two sides. It will only have one side if the player could never be on the second side (For example, a wall that defines a square column, the 'inside' portion of the wall need not be defined because the player can never be inside the column). In addition, a LineDef has DIRECTION: it goes FROM one vertex TO the second vertex. Because the LineDef has direction, we can define which side of the LineDef is the 'right' side and which is the 'left' side (imagine standing on the 'from' vertex looking at the 'to' vertex, along the line, the right and left sides become apparent.)

This brings us to golden rule #1 of Doom Levels: if a LineDef has only one side, **IT MUST BE THE RIGHT SIDE.**

SideDefs

There is one SideDef defined for each visible side of each LineDef. That is, there are either one or two SideDefs defined for each LineDef. Again, if a LineDef has only one side (and therefore one SideDef), it must be the RIGHT side. We call the right side SideDef "SideDef 0" and the left side SideDef "SideDef 1".

The SideDefs define what textures are seen when standing on this side of the LineDef. There are three textures to be defined for each SideDef. The first is the texture of the the Wall itself. If the LineDef is transparent, the texture is defined as "-". The second texture is called the "WallBelow" texture, and is the texture that will be seen "below" the level of the floor defined for the other side of the Linedef. For example, consider a stairway. As you go up the stairway, the front of each step is a LineDef. The stair on the other side of the LineDef is higher than the stair you're standing on now, so the "WallBelow" texture is the texture from the floor of the higher stair down to the floor of the stair you're standing on. In other words, it's the texture for the front of the step!

The third texture is the "WallAbove" texture, and is the texture that will be seen "above" the level of the ceiling on the other side of the LineDef. For example, if a column is jutting down from the ceiling, the column's "ceiling" is the bottom of the column, and the columns' "walls" will be the WallAbove texture for the LineDef defining each side of the column.

The last (and maybe most important) thing to understand about SideDefs is that **an enclosed set of SideDefs defines a sector.**

Sectors

This is the important most one to understand. **A sector is an enclosed area of space with a common floor and ceiling.** Therefore, a stairway is made of several sectors, one for each step. The outside courtyard on original Doom game 1, level 1, is two sectors, one for the actual courtyard, and one for the pool of green nukeage in the center. The pool must be another sector because it has a different floor (the nukeage).

Also, a sector's floor and ceiling can change levels, which is how we create lifts, doors, traps, etc. This is an important distinction: **Sectors move, NOT LineDefs.** A Door is a sector whose ceiling is defined at the same level as the floor. When we open the door, the ENTIRE SECTOR's ceiling raises up, and we walk through. When (if) the door closes, the ENTIRE SECTOR's ceiling drops back down to the level of the floor.

Now to tie it all together. To make a Doom Level, we first create some Verticies. Then, we attach the verticies together with LineDefs (creating SideDefs as we go). Then, we select a group of enclosed SideDefs to define a sector. We repeat the process until we have all the sectors done, then we save.

Using the Map

When you first select a level, the entire level is displayed in the window area. Resizing the windows will automatically resize the map.

You can zoom in and out of the map by pressing the + and - keys. 5 Zoom levels are supported in each direction

To center on a new spot, Hold down the CTRL key and click on that spot with the left mouse button.

In general, the various objects are moved around (if they are moveable), simply by clicking on the object and dragging it around. The right mouse button is used to bring up extended information about the current object, for further editing.

Thing Editing

 To get into Thing Editing Mode, select the thing toolbox item

When in Thing Editing Mode, all the Things appear as colored dots using the following color codes:

Red: Monster
Dark Gray: Player Start Positions
Cyan: Weapons/Ammo
Light Gray: Scenery (torches, carnage, etc)
Green: Barrels
Magenta: Goodies (RadSuits, Armor, Light Goggles, etc)
Dark Yellow: KeyCards and Skull Keys

To move an existing thing, just click on it and drag it to the new location.
To add things, select INSERT THING from the menu. The new things will be an exact duplicate of the previous thing. Click on the map for each position for the new thing, adding as many new things as you wish. When you are done adding, right click with the mouse to turn insert mode off.
To delete a thing, select DELETE THING from the menu. The current thing will be deleted.

To edit a thing's attributes, right click on the map when a thing is selected. The Thing Editing window will appear. To change what the thing is, select first the Type of thing using the "KMAGS" buttons. Each button stands for the following:

K: Keys (and Player Start positions)
M: Monsters
A: Ammo and Weapons
G: Goodies (potion, armor, stuff you can pick up except weapons)
S: Scenery (including barrels)

Once a button is selected, choose the desired thing in the list box.

The angle boxes are relevent only to monsters and player start positions, as other objects always face the player. The skill bits check boxes are for selecting which levels the thing will appear on, and other attributes:

1/2: Appears in difficulty levels one and two.
3: Appears in difficulty levels three
4/5: Appears in difficulty levels four and five. (Difficulty level five available as of Doom version 1.2)

Deaf: Clicking on this check box will make a monster 'deaf': he will not respond to gunshots within earshot. He will attack a player only when he sees him.

Net: Makes the thing available only in network games.

Vertex Editing

 To move an existing vertex, just click on it and drag it to the new location. You may also right click on the vertex and type in the coordinates manually.
To add vertecies, select INSERT VERTEX from the menu. Click on the map for each position for the new vertex, adding as many vertecies as you wish. When you are done adding, right click with the mouse to turn insert mode off.
To delete a vertex, select DELETE VERTEX from the menu. The current VERTEX will be deleted. (You cannot delete a vertex that has one or more LineDefs attached to it).

 **Multiple Vertex Editing.** By selecting this box (avaiable ONLY in vertex edit mode), you may drag a box across the map to select multiple verticies. Then, by moving the box, All verticies within the box will move at once. In addition, if you right click on the box with multiple verticies selected, you may ALIGN the verticies in either the x or y direction by clicking on the appropriate check box. This is good for creating straight walls.

Sector Editing

 I'm jumping right to sector editing because you can create new Sectors, and that sector's defining LineDefs/SideDefs, all in one step!

Once a given sector's Verticies have been defined, it's VERY easy to create a new sector from these verticies. Just go into Sector editing mode and select INSERT SECTOR from the menu. Then, connect the new verticies in a CLOCKWISE fashion by dragging the displayed red line from Vertex to Vertex and clicking the mouse button.

Once the sector has been defined, simply right click with the mouse. You will have just done the following:

Created one LineDef between each pair of Vertecies.
Created one SideDef for each LineDef, (on the right side, of course)
Create one new Sector. All the new SideDefs point to this sector.

That's it!. Now, just right click with the mouse to edit the sector's attributes. You may also edit the attributes of an existing sector by clicking on one of the LineDefs associated with that sector. Clicking on the LineDef a second time will switch to the OTHER SideDef for that LineDef, and it's associated sector.

To edit an existing sector, make that sector current, and then select "Edit Current Sectors LineDefs" from the menu. Then, select the new LineDefs to attach to that sector. You can pick a LineDef that's already attached to this sector while in this mode, and DoomCAD will not duplicate the definition.

Sector Edit Dialog
The following attributes can be edited for an existing sector:

Ceiling/Floor Height.
The upper and lower limits of floors/ceilings are unknown. I've seen ceilings as high as 512 or so.

Ceiling/Floor Texture:
To choose a texture, just find the texture you want in the list box and drag it to the Ceiling/Floor Texture box. Double clicking on the texture will display it in the Preview Window.

Special List: Choose one of these attributes for the sector. The -10/20% attributes are how the 'nukeage' areas are created. The secret areas are added up to get the secret percentage at the end of the level. (To get credit, the player must step on the sector marked secret. This is easily acoomplised by putting something good in the sector, like ammo or a medkit.)

Brightness: Can go from 0 (dark) to 255 (outside light).

Trigger: This number ties the sector to a LineDef somewhere on the level with the same trigger number. See the LineDef Editing section for a description of triggers.

LineDef Editing (also SideDef Editing)

 To Select a LineDef, click on it near its center. To edit attributes for the LineDef, right click on it. The following attributes are editable for each LineDef:

Length: Changing the length will change the position of the LineDef's TO vertex. Note that if other LineDefs are attached to this vertex, their lengths will change as well! This is good for making sure a Wall is exactly 128/256/nn long, to match its texture size.

Action: The largest listbox defines what the LineDef will do when it is 'Activated' The activation event is one of the following, denoted by the first characters in the listbox elements:

W: Linedefs is tranparent and must be <W>alked over to trigger action to triggered sector.
D: Linedef is a <D>oor: SideDef1 of this LineDef points to the door's sector
S: Wall is a <S>witch, and must be pressed to trigger action to tirggered sector.
1: Action happens once
R: Action happens repeately

Some actions happen only once at a time, but can be 'undone' to happen again. For example, if a sector is triggered to "Move Down to Neighbor Floor" by crossing LineDef A, and to "Floor Move up to Neighbor Ceiling" by pressing switch on LineDef B, that sector can move up and down by repeating the two triggers

Attribute Bits:
Bit 0 Impassable: Players, monsters, and projectiles will not cross this line.
Bit 1 Monsters cannot cross: They can't cross if Impassable is set anyway, so this is good for 'invisible' cages or the like.
Bit 2: Transparent. If you don't set this bit and select '-' for either of the SideDef textures, the player will see "hyper junk" (quoted from Matt Fell: I love that!) Also note that if this bit is set, projectiles WILL travel through this LineDef, even if there is nothing on the other side (probably resulting in a CRASH).

NOTE: Every LineDef must either be Impassable or Transparent. One and only of these attributes must be set.

Bits 3/4 :Unpegged textures do NOT move with moving ceilings/floors, so the effect is that the texture "just sits there" as the ceiling/floor moves, looking more like the texture is being created to fill the space. The best way to describe this is to see it: mark a door as having an Unpegged Ceiling and watch it open/close.
Bit 5: Secret: This just protects the drawing of what's behind it on the Automap. It does NOT have anything directly to do with finding secret areas that count toward the percentage at the end of the level. Those secret areas are sectors with a secret attribute set.

Bit 6: Blocks Sound. Sound does NOT travel through Impassable walls, anyway.
Bit 7: Not shown on automap. Good for outer level boundary walls.
Bit 8: Already on map: This Line is seen right from the start of the level, even if the player hasn't been there yet.

Trigger: Trigger numbers tie the LineDef to a Sector with the same Trigger number. Type the trigger number you wish in the box provided. If you wish to create a new trigger, select the 'new' button above the trigger box, and a previously unused trigger number will be assigned. Once a trigger has been assigned, clicking on the 'Sector...' button will bring up the Sector Editing box for the Sector having the same trigger number.

NOTE: A LineDef can be tied to more than one Sector, and vice-versa: If this is the case, clicking on the 'Sector...' button will bring up only ONE of the Sectors tied to this LineDef. The other Sector can be edited in Sector Editing Mode, of course.

SideDef Data:
X,Y: The Coordinates that textures will be 'pasted' onto the Linedef, usually left 0

WallAbove: The texture that is visible on any part of this LineDef above the corresponding sector's ceiling. For example, it this SideDef's Sector has a lower ceiling than its neighbor sectors, the side of the wall will be visible. This texture defines that part of the wall. (Note: Doors are initially defined as a sector whose Ceiling is at the same height as the floor, so the door graphic is actually the WallAbove texture. The Wall texture for a door sector should be left "-", so the door can be passed through when it is opened.

Wall: The texture that makes up the area between the ceiling and floor of the corresponding sector. This texture is usually "-" for transparent LineDefs, although SOME of the transparent LineDefs have textures (cages). This must be defined for Impassable LineDefs (on both sides, if they exist), or you get 'hyper-junk' on the wall.

WallBelow: The texture that is visible on any part of this LineDef below the corresponding sector's floor (like the front of a stair)

To edit the above textures, drag the desired texture from the scrolling listbox and drop it in the desired box. (The scrolling listbox shows the texture name, height, and width). To clear a texture, click the small button marked 'c' next to the desired box. To preview a texture, double click on the texture, either in the listbox or the Wall/WallAbove/WallBelow box (the palette for the picture may not be exact, Windows won't let me have all 256 colors!)

Three Dimensional Preview

 This is the feature you might play with for hours, just because it's so fun!.
3D preview allows you to look at any Doom level in a Three Dimensional black and white preview, with hidden line removed. Clicking on the camera icon will bring up a Blue Line segment, representing a camera. The lower left side of the line is the camera location (where the viewer is standing). The other line segment is the camera viewpoint (where the camera is looking). You can move either one of these viewpoints around. When you get to a spot you'd like to see in 3D, press the right mouse button.

When in 3D mode, pressing the CTRL key puts the camera in interactive mode. You will see a grid on the screen. Moving the mouse will rotate the grid in 3D space, which represents moving the camera location. When you press CTRL again, the level will redraw at the new rotated coordinates.

Software Creations

"Home of the Authors"

Don't forget where id Software's home is! The one and only Software Creations BBS.

1200/2400 V.42/MNP Lines : (508) 365-2359
2400-14.4k HST US Robotics Lines : (508) 368-7036
2400-16.8k V.32/V.42bis US Robotics lines : (508) 368-7139
14.4-28.8k V.32/V.42bis/V.fc Hayes Optima lines: (508) 365-9352
14.4-28.8k V.32/V.42bis/V.32terbo/V.fc US Robotics lines: (508) 368-3424

This is by far the best BBS on earth! Call today!

Note: Software Creations is NOT responsible for distribution of DoomCAD, it is being distributed solely by the author. Registration will be handled soley by me!

PreFab Mode

 Prefab Mode lets you create Prefabricated structures. Specifically, you can create stairways, doors, or rooms in a few quick steps.

To add a Prefab object, select Prefab mode then Insert Prefab <door/room/stairway> on the menu. A dialog box will come up, asking questions about the structure. When the structure definition is complete, you will locate the spot on the map to place the structure and click. The structure will then build itself.

NOTE: When placing a prefab Room or Stairway WITHIN an existing sector (like on a new map), be sure to connect the OUTER WALLS of the structure to the 'parent' sector. This is done by going into sector mode, selecting the 'parent' sector', and selecting "Edit current Sector's LienDefs" from the menu.

About The Authors

matt tagliaferri: just your average Doomaniac with way too much time on his hands. I quickly played all 27 levels of the game, and was left craving more. I have two motivations for writing DoomCAD:

Creating some quality levels for Doom play.
Having hundred of users create levels for my use! (ha)

other contributers:

Matt Fell*:* Author of *"The Unofficial Doom WAD File Specs",* my personal bible for this project. His tireless quest for knowledge made DoomCAD possible, because I probably wouldn't have had the patience to hack a 10 meg file!

Raphael Quinet: Author of Deu 5.0, the FIRST level editor with levels from scratch support. He was the first to delve into the complicated Binary Space Partition (BSP Tree) algorithm needed to correctly save a Doom level. His generous inclusion of source code for this algorithm was the final piece of the puzzle for DoomCAD's ability to create levels from scratch.

Everett Pence: another Doomaniac who's sick of playing the original 27 levels. He wrote the initial code for the graphic preview, and had many good suggestions for interface improvement.

Carlton Guc: Beta tester extrodinaire, and future hall-of-fame level author. His bent imagination is perfect for creating evil Doom levels. (Example: When DoomCAD was just a thing editor, he put a Cyberdemon in the outside "secret" area of E1M2, and he moved the red key out there! We were playing 3 player cooperative mode, and didn't have any weapons yet, either....)

Micro System Options: Creator of 3D Graphic Tools for Windows, the library used for 3D preview. Great product for 3D graphic gearheads like myself. Believe it or not, the 3D stuff took me only 6 hours to implement using this great product.

id Software: the ones who started this mess. They didn't do much <g>, except provide us with the most immersing 3D game to date, leave the file formats unencrypted and hack-able, then give us permission to hack it and charge money for it! Special thanks and applause to this stand-out organization.

Registration

To register DoomCAD, please send a paltry $15 to:

matt tagliaferri
4416 Brooklyn Avenue
Cleveland, OH 44109 (go **TRIBE!**)

id Software is graciously allowing Doom Editor authors to charge money for our work, please support us through registration. Registered users will of course get all the benefits of registration: free updates, unlimited praise, tech support, etc, etc.

Source code for various sections is available. I cannot provide source for the 3D stuff (not in my license), but if you'd like my node/blockmap algorithm, let me know. DoomCAD is written in Visual Basic 3.0, professional edition.

Nodes

(also BlockMap, SSectors, Reject, and Segs)

These are additional structures that you as Doom Level author need not concern yourself with, but Doom Level *Editor* authors (like me) pulled their collective hair out trying to figure out! These structures are created from the other structures (Verticies, Sectors, etc). The automatic generation of these structures is quite complicated, and if it fails, your Doom level will not display properly. If your level fails during Nodes/Blockmap etc generation, please send the WAD to me and I'll try and figure out what the problem is. Please look at the troubleshooting section before you send it to try and find the problem, it may be a simple thing!