

# Manual VIS Help/Tutorial

## Question #1:

*I have heard many discussions about manual VIS, however I have no idea how to achieve this and what the advantage is. Please open my eyes for me.*

## Answer #1:

Basically, **manual vis** is just that. It is you, the mapper, telling the engine what should and should not be drawn in your map.

This is accomplished through the use of **vis\_leafgroups** and **area portals**.

Here we will discuss **vis\_leafgroups**.

Leafgroups are more often used in outdoor maps where there are limited structural brushes to create view portals. These view portals are, put simply, 3 dimensional "areas" in your map. They are linked in what are called chains. Again, simply, a chain length is determined by how many portals can be "seen" by any portal. "Seen" in this case means that the area contained within a "seen" portal X will be drawn when in portal Y.

What the leafgroups do, is end that chain where you want.

You are basically saying to the engine "That targeted group of portals (a **vis\_leafgroup**) cannot be seen when within the boundaries of this leafgroup that I am in, and therefore should not be drawn."

To some degree, the engine does this process on its own. But sometimes, when there are few or no enclosed structural areas, the engine needs help.

By enclosed structural area, I mean an area cut off from another area by structural brushes that somehow connect/abut each other along a viewplane.

**Structural** is important here. Structural brushes create portals, detail brushes do not.

Detail brushes are not taken into account during the vis compile when determining portal chains. Structural brushes are, because they create portal(s). And following that, structural brushes can improve vis by blocking the engine from drawing what is behind them until it is time to draw that area.

A manual vis setup consists of brushes covering an area that are textured with the VIS texture (found in "common" or "utility") and then made a **vis\_leafgroup** by rightclicking in 2D view and then clicking vis-->leafgroup.

But one group by itself does nothing. It needs to be targeted to another leafgroup. By targeting it, you are now saying to the engine that the "targeted" group should not be drawn when I am in the "targeting" area.

Here is an example of how a leafgroup can improve performance.

Note in this first shot all the polys that are being drawn (represented by the white lines):

**NO PICTURE AVAILABLE**

Now look at this second shot after the **vis\_leafgroups** were targeted:

**NO PICTURE AVAILABLE**

Note the decreased polys, as well as the increased FPS.

This is the advantage that manual vis can give.

**Question #2:**

*When you are in a certain vis\_leafgroup, and it has a target, what happens when you get into the target if there is another that is supposed to appear in succession? I hope that gibberish is understandable. Basically I think I am just asking a clarity of the "chain" part, obviously you would need these to spawn "draw" certain areas depending on where you are.*

**Answer #2:**

I think you are confused about what was previously told about chains.  
The principle is simple.

If you are in a visleaf "A" that targets visleaf "B". Then you cannot see anything from B when in A.

If you are in "B" you can still see everything from "A" since we never targeted B -> A.

If in visleaf "A" (thats targeting "B" ). "A" is NOT interested what "B"'s targets are.

if "B" targets "C" and "A" targets "B" then C is still visible from A.

So there is no kind of hierarchy in this. So if you want the player not to see anything from "C" when he is in "A" you simply need to target "A" also to "C" (You can use multiple targets on one visleaf)

**Question #3:**

*Can you make all brushes structural and avoid this problem as long as you don't have large open outdoor areas?*

**Answer #3:**

Your question is implicitly already answered before. Structural brushes and visleafs count on the amount of stuff to be calculated by the vis engine. (2nd compile stage)

When making everything structural you will overflow the vis engine with that many brushes that needs to be calculated that it cannot handle it.

Bad usage of structural brushes or too many unneeded structural brushes will increase vis-compile time exponential. With wrong usage a little map can compile for days - without any positive FPS effects at all. A well designed map with a good policy of detail/structural brushes can compile vis in an hour or less. (the screens that Crunch posted were from Alsace that compiled in 59 minutes vis - with a bsp size of 17M and a visdatasize of 648k - a bad map can have a bsp size of 800k and a visdatasize of 80k and will compile for days )

Here some tips to keep compile times acceptable and getting gain of the usage of structural brushes

Make only stuff structural that are good candidates to block viewlines. e.g. long huge walls. Underground tunnels (to not see the upper part). Do the rest with visleafs to divide your map. And test that! Check out if your structural brushes and visleafs really do what you designed them to!

Be sure to make things detail like: Window frames, building ornamentals, angled roofs, odd angled brushes in general, stairs and all other brushwork that is highly unlikely to be candidate for a viewblocking effect.

**Extra info by Bdbodger:**

Basically you make a brush textured with the vis texture and with it selected use the right click menu to make it a vis leafgroup. When you have made several you select one, click on a second one, use control K so the first one targets the second one. You should see the second one turn blue and a line going from the first one to the second one. Now click on a third one and use control k again. Same thing happens a line from the first one to the third one and the third one turns blue. When you do this when a player is in the first one nothing inside the second or third one will draw. This is also called manual vis. It is a lot of work, you also have to select the third one and the first one and use control k also to do it in reverse. Just because you have done it so that area 3 does not draw when you are in area 1 does not mean area 1 will not draw when you are in area 3. Anything inside a vis leafgroup that does not target another vis leafgroup cannot be seen unless you are inside that vis leafgroup . All vis leafgroups should target one or more vis leafgroups. I hope that helps a bit.