

BSP FAQ QUESTIONS:

1. What does the map Rebuild process for the Unreal engine do anyway?

2. What is the difference between Rebuilding the Geometry and Rebuilding the BSP?

3. Do you always need to rebuild the geometry and the BSP together?

3a. When would you ever want to just do one or the other alone?

4. On the Rebuild BSP menu, what settings can you safely change?

4a. What does each of the settings that you can safely change do?

5. What does the BSP slider do exactly?

5a. Why would you want to make more or fewer BSP cuts using the slider?

5b. What are the map or performance tradeoffs for using the slider settings?

6. I just added/subtracted a new brush and now I have a big black hole in my map. What caused it?

6a. How can I get rid of the big hole (aka BSP hole)?

7. My map looks fine in the editor but when I play it I see a "Hall of Mirrors" effect. What causes this?

7a. How can I find and fix very small holes in the editor?

7b. What should I be looking for to spot small holes when I use the editor's BSP view?

8. What does setting the brush order to "first" or "last" do?

8a. Why would I want to do that and when should I do it?

8b. A help file said that to optimize maps you need to reorder all your brushes with the large additive & subtractive ones first. However, when I try this large portions of my map architecture seems to disappear. What am I doing wrong? What are the rules for doing this right?

9. What is the difference between solid, semi solid and non-solid brushes?

9a. When I play the map I don't see any difference between a solid and semi solid brush. Is there a difference in the game?

9b. The default for adding a brush is a solid brush. When would I want to turn it into a semi solid or non solid brush?

Why? What are their uses?

9c. When should I never make brushes semi solid?

9d. What exactly is a decorative brush?

9e. Can I safely subtract from a semi solid brush.

10. Can editing vertexes cause BSP holes?

11. Why wouldn't you want to intersect your entire level and turn it into one big brush? Isn't that supposed to improve performance?

11a. Can intersecting lots of brushes into one brush cause BSP holes?

12. Are there any rules I can follow when building my map to help prevent BSP errors?

13 What are the most commonly found BSP errors?

13a. How can I avoid them and/or recover from them.

1. What does the map Rebuild process for the Unreal engine do anyway?

A: Comparing this to Q2, Rebuilding is like compiling. It essentially puts the map together as it should be. Sometimes during the process of editing a map you can delete, move, or add stuff and the maps information won't be correct. Rebuilding ensures that everything in a map is correct and is working as it should. Sometimes though Rebuilding will create problems known as BSP holes (talked about later).

2. What is the difference between Rebuilding the Geometry and Rebuilding the BSP?

A: Rebuilding the geometry is just that. The 3d view doesn't keep track of moving or deleting of brushes. By just Rebuilding the geometry you can very quickly get a good look at what you've changed in the 3d view. Kind of think of the 3d view as a compiled view of the map.

BSP on the other hand a BSP can help remove BSP holes (talked about later) and if you haven't Rebuilt the BSP lately can speed up the 3d view a little. Essentially though what it does is keeps a list of what polys to draw and what not to.

3. Do you always need to rebuild the geometry and the BSP together?

A: No

3a. When would you ever want to just do one or the other alone?

A: Let's say you're working on a fairly extensive map and time is money. So Rebuilding just the Geometry is much quicker than the Rebuilding all of it together (Geometry, BSP, and Lighting). Catch is though by just Rebuilding the Geometry, you don't Rebuild the Lighting so you end up with a fullbright scene.

4. On the Rebuild BSP menu, what settings can you safely change?

A: Anything in the Optimization section. Avoid at all cost the BSP Rebuild Options section. People have completely killed UnrealEd by changing these.

4a. What does each of the settings that you can safely change do?

A: The Lame/Good/Optimal settings adjust how well it should handle the poly visibility and how many polys can be added or merged together.

A setting of Lame causes less poly visibility optimizations and create can create unnecessary polys to be created.

Good is the fastest setting and does a relatively good job of course.

Optimal does the best of this and generally creates fewer polys and merges whatever it can.

5. What does the BSP slider do exactly?

A: The slider bar adjusts how many additional cut that the BSP needs to make. This is a physical number but a generic setting. Lowering it will lower the poly count and raising it can increase it. But there are reasons for this that will be mentioned in a sec.

5a. Why would you want to make more or fewer BSP cuts using the slider?

A: Good question and it took me a while to figure out why. A lower setting creates fewer polys, that means that the BSP has less to work with and if it has to it will draw more than is seen. This is good in larger open areas where stepping into a new area can cause a stutter in gameplay because the game has to look up the BSP tree for that info. If part of that BSP tree is already being drawn then the stuttering isn't as bad.

A higher setting is more intentional for areas where you need lots of occlusion (fancy word for saying "not drawing polys behind stuff"). While I've never run into this kind of area I would assume a small complex area would benefit from this the most.

5b. What are the map or performance tradeoffs for using the slider settings?

A: Well a low setting creates fewer polys. Good, yes can cause problems though. Such as unnecessary overdraw and anything that's touching that poly (players, monsters, decorations, etc) must be drawn to.

A high setting can create too many polys. If those polys are seen from a distance then the person looking could be facing upwards of 200-300 polys. A lower setting would create lower poly count. It also means that the game has more polys to keep track of which the CPU must do the work

of in order to keep the game playable.

In my opinion, I suggest sticking with the default 15 setting as it is also been optimized for compiling speeds and is generally the best all around.

6. I just added/subtracted a new brush and now I have a big black hole in my map. What caused it?

A: First verify that it is a BSP hole. Do this by Rebuilding the map and changing the 3d view's Mode to Textures. If the big black hole is still there you've got a slight problem.

6a. How can I get rid of the big hole (aka BSP hole)?

A: While there is no one guaranteed solution there have been several that seem to help.

Lower the detail of the space you're working with.

Begin to Zone off areas as this will create less polys per Zone.

Turn decorative brushes into semi-solids.

Change the BSP Optimization to Optimal.

Right-click the brush with the BSP hole in it then click on Order=>To Last. Then Rebuild the map.

7. My map looks fine in the editor but when I play it I see a "Hall of Mirrors" effect. What causes this?

A: The HOM or Hall of Mirrors effect is a BSP hole. Also in large maps where you can see great distances, then it might just be a Z-buffer problem. Nothing you can do about it since it's a 3d hardware issue.

7a. How can I find and fix very small holes in the editor?

A: Generally they're stumbled upon. Some are very obvious though. Best solution is to switch the 3d view's Mode to Textures or Zone/Portal view as this gives you a very clear picture to work with for finding BSP holes.

The fixing of these was just mentioned under "How can I get rid of the big hole (aka BSP hole)?"

7b. What should I be looking for to spot small holes when I use the editor's BSP view?

A: Generally a black hole of any size. Any missing polygons. If when you move the 3d view close to it and you can see the blue grid. Those are definite signs that it's a BSP hole. Some BSP holes though can only be seen from certain angles. Those are the ones that you stumble upon.

8. What does setting the brush order to "first" or "last" do?

A: Brushes are numbered when they are added. The "to first" command will make it like the first brush you subtracted or added in your level. Generally it's not that useful though. The "to last" sends it to the end of the line.

The best way to describe this is to subtract a cube. Add a small object inside it that stretches outside of the cube. Now subtract another cube adjacent to the first where the small object inside would be. You should notice that parts of the object is now missing. Now right-click on the object and click Order=>To Last. Then Rebuild. Now all of the object is there. Now select one of the cubes and make it to Last. Rebuild. Now part of it is gone. Now click on the shape and change the order to First and then Rebuild. Now it's gone again. Does that make sense at all?

8a. Why would I want to do that and when should I do it?

The "to first" command usually isn't that useful except for say a large subtracted brush, that's your terrain, that's subtracted after you've made everything. A "to first" change will make it so that the rest of your architecture stays intact and that you've got a different terrain

The "to last" though is a bit more useful. Let's say you've got a subtracted doorway. But you've made a design flaw and you accidentally added a wall in the middle of the doorway. Simply select the brush and change the order "to last" and then Rebuild the geometry and it'll be fixed.

Or if you made a small dropship out a brush and you need more room for it. So you subtract a bigger plot of land for it and move it there. When you Rebuild it won't be there. To fix that you'd select the brush and change its order "to last" and then Rebuild again. Then the dropship will be there again.

8b. A help file said that to optimize maps you need to reorder all your brushes with the large additive & subtractive ones first. However, when I try this large portions of my map architecture seems to disappear. What am I doing wrong? What are the rules for doing this right?

A: I don't recall reading that but I would assume that it mainly applies to such things as a giant subtracted/added brush that acts as your terrain.

9. What is the difference between solid, semi solid and non-solid brushes?

A: Solids are just that.

Semi-solids are like solids but don't make poly cuts into Solids. They do make cuts into each other though. Also they have a bad tendency to cause collision errors in the map if they are often within the players touching range. They can also be used for decorative objects to avoid BSP holes.

Non-solids are just that. Brushes that can be walked through. There can be some interesting uses for these.

9a. When I play the map I don't see any difference between a solid and semi solid brush. Is there a difference in the game?

A: Actually there is. If you type in "stat fps" into the console while playing you'll notice that the node count in the scene will be different. Generally a

room with semi-solid decoration brushes will have a lower node count (which is better than a low poly count)

9b. The default for adding a brush is a solid brush. When would I want to turn it into a semi solid or non solid brush? Why? What are their uses?

A: Change any decorative brush into a semi-solid. Torch holders, custom built chairs, tables, occasionally hand rails, even ladders are good examples of things that should be semi-solid. I understand that breaks the rule of keeping them out of play but generally the collision problems are far and few between.

Non-solid brushes can be use for custom fit ZonePortals. Ever have a fountain or above ground pool made but can't get a sheet brush to fit right cause it overlaps? Here's where a non-solid brush would come in handy. Just deintersect it to get the right shape, add it, make it a non-solid and then change the Surface Properties so that it becomes a functioning ZonePortal.

9c. When should I never make brushes semi solid?

A: Anything that you expect to subtract out of later. Semi-solids refuse to be subtracted. Also most walls of any building. Generally it's just a bad idea to cut out a large chunk of a map and make everything a semi-solid in hopes of reducing the poly count.

9d. What exactly is a decorative brush?

A: I knew this would come up :)

A decorative brush is stuff like tables, chairs, computers, anything that's meant as eye candy but is made with brushes.

9e. Can I safely subtract from a semi solid brush.

A: Trick question. You can't. Trying to subtract from a semi-solid is asking for trouble. What you have to do is to first make the brush a solid and then to subtract it. Then select the perform an intersection on the brush. Use that as your semi-solid.

10. Can editing vertexes cause BSP holes?

A: I've generally kept my vertex editing to simple cubes. I tend to move entire edges rather than individual vertices. I've run into very few problems doing this. But in short, moving a vertex the wrong way can cause BSP holes and even crash UnrealEd.

11. Why wouldn't you want to intersect your entire level and turn it into one big brush? Isn't that supposed to improve performance?

A: Intersecting the entire level into one brush creates a very complex brush for Unreal/UT to work with. It also can make it less editable for you. By creating this complex brush, you're kind of inviting a BSP hole or collision error somewhere. Any given brush used in UnrealEd should be kept to under 500 surfaces.

11a. Can intersecting lots of brushes into one brush cause BSP holes?

A: Depends on how complex the brushes are. If when you intersect the brushes and it creates more polys then you can even keep track of then it's probably too complex. But often intersection is used just to reduce the number of brushes you have to handle. I make decorative stuff into 1 intersected brush just for convenience matters.

12. Are there any rules I can follow when building my map to help prevent BSP errors?

A: Keep it simple. Don't get extremely detailed. While all the eye candy may look cool it's like telling a BSP hole to come to your map for a BBQ.

13 What are the most commonly found BSP errors?

A: Often just BSP holes. Some of the extremely nasty ones have corrupted entire maps making them unplayable. Then again the only cases of this happening involved highly detailed maps (to the extent that it should've been done in something like 3D Studio Max).

13a. How can I avoid them and/or recover from them.

A: Blatantly ripping from stuff that Myscha the Sleddog has said...

Do not subtract through semi-solids

Do not intersect ZonePortal brushes with Semi-Solids or other Zone Portal brushes

Keep invisible collision hulls away from solid/semi-solid geometry.

Do not intersect a semi-solid through a ZonePortal (such as a semi-solid column through a waterline)

Never use NON_SOLIDS ANYWHERE. (Wanderer: Now I'm gonna have to disagree on this. Sheet brushes are ok to make non-solid. And I've yet to run into any problems with non-solid brushes. If you do contact me.)

Wanderer again: Also something I should've stated earlier on fixing BSP holes. I've occasionally been able to fix some minor BSP holes by adjusting the slider and changing the Optimization to optimal. Had to do this once to cure a collision error. Not pretty but it worked.

Use semi-solids on decorative brushes to help lower the node count.

Make backups. Or better turn on Autosave. Clipped straight from the next version of the UnrealEd FAQ on enabling Autosave.

Enabling the Autosave feature

The best thing to do if you haven't done this is yet, is to enable the Autosave feature. Just go into Properties=>Editor and set Autosave to true. Then readjust the times to how much you want to back up. If you want to make sure that the an hours worth of work is saved up each time set it to 6 minutes. The only side effect is the occasional pause.

And should something happen, just go into the UnrealTournament/Maps folder and readjust the files by date and edit the latest auto#.unr file. Then rename it accordingly.

Even if this is old news it's still valuable.