

TWEAKING PATHNODES

by Schlacht (dehags@airmail.net)

This document will explain a few things I've learned while working with pathnodes so that you can avoid the pitfalls that I did not. It will not tell you the exact details of adding and manipulating pathnodes in UnrealEd. Other tutorials cover that quite well. Nor will it explain using pathnodes with lifts or using the special purpose nodes mentioned below. It will tell you some basic do's and don'ts about adjusting your pathnodes so that bots will perform better on your map.

THINGS YOU'LL NEED TO READ

So now you've got your level made, textures all aligned, and it's time to setup those pathnodes. Or maybe you've setup your pathnodes and things aren't working quite like you expect. In either case you're going to need to know a few things.

You really need to read the [Unreal Creature Care and Feeding Guide](#) - by Steven Polge of Epic MegaGames, Inc. It will tell you everything you need to know about setting up pathnodes so that bots will be able to use your level.

It tells you about placement of the pathnodes, how to set up pathnodes so bots will use lifts, and then goes on to discuss special purpose nodes like "AmbushPoint," "JumpSpot," and "TranslocDest" which tell bots that "This is a good spot to lie in wait," "You can jump to this spot," and "You can translocate to this spot."

It also covers "UT Game Types" - telling about things you must do for your level to work with bots for team games, "Triggers" - details about trigger operation, and "ZoneInfos" - information about zone settings.

Definitely required reading. It would be a good idea to save this on your hard drive for future reference.

OTHER PATHNODE INFORMATION / TUTORIALS

[Unreal.net - Pathnodes](#) - (by Sharky) Brief tutorial about where you should place your pathnodes.

[Unrealized: Ask the Expert](#) - (by Myscha) Tells how to add the path nodes to your map and mentions some good tips about placing your pathnodes.

The below text assumes that you have read the section titled "The Navigation Network" of the [Unreal Creature Care and Feeding Guide](#).

GETTING DOWN TO BUSINESS

Once you've read the "The Navigation Network" section you should be able to place your pathnodes fairly effectively. Keep in mind that the network of pathnodes will almost certainly require tweaking to make bots work well with your map. Since you've (hopefully) spent a lot of time tweaking and tuning your map's appearance to make it as good as you possibly can, you probably want the bots to know your map inside and out so they can give unwary players a serious thrashing.

If you have read the "Navigation Network" section of the [Guide](#) you should know how to make UnrealEd define the paths between the path nodes and how to view the paths between the nodes. You should also be aware that the **BLUE** paths are more "preferable" to bots than the **RED** paths are.

When adjusting your pathnodes you should always save your map first and then only move one or two of the pathnodes before re-defining the paths. This will let you observe any changes to the paths between the pathnodes and decide if the changes were good or bad. Once you get better at adjusting you might decide to move more pathnodes at once.

If adjusting a pathnode causes one or more of the paths connecting that pathnode to other pathnodes to turn from **RED** to **BLUE** then you have made those paths "preferable" to bots. This is GOOD. You might want to save your map at this point.

If adjusting a pathnode causes one or more of the paths connecting that pathnode to other pathnodes to turn from **BLUE** to **RED** then you have made those paths less "preferable" to bots. This is BAD. You should immediately move the pathnode(s) back where they were before. Re-defining the paths after putting the path node(s) back where they were should turn the paths back to blue again. Or you could just reload the map you

saved before moving the pathnodes. You saved it before making changes didn't you?

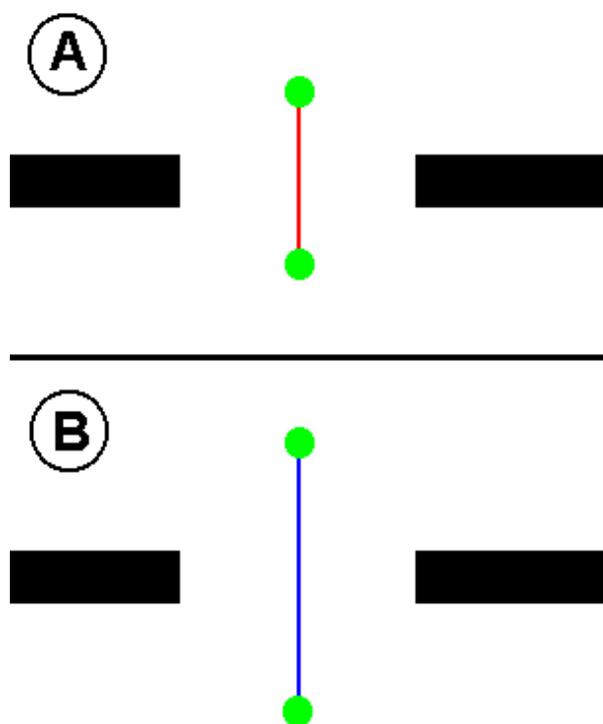
By making as many paths as possible BLUE you give a bot more path choices when it is navigating. While it may not be possible to make every path on a pathnode BLUE you should try to ensure that at least one BLUE path goes from a pathnode to each of the pathnodes closest to it. If a pathnode doesn't have any BLUE paths at all it is unlikely that a bot will go to that node at all unless forced to by other circumstances.

Things that can make a path RED: 1) pathnodes too close together, 2) the path between two nodes being partially or completely obstructed by a brush or coming too close to a brush.

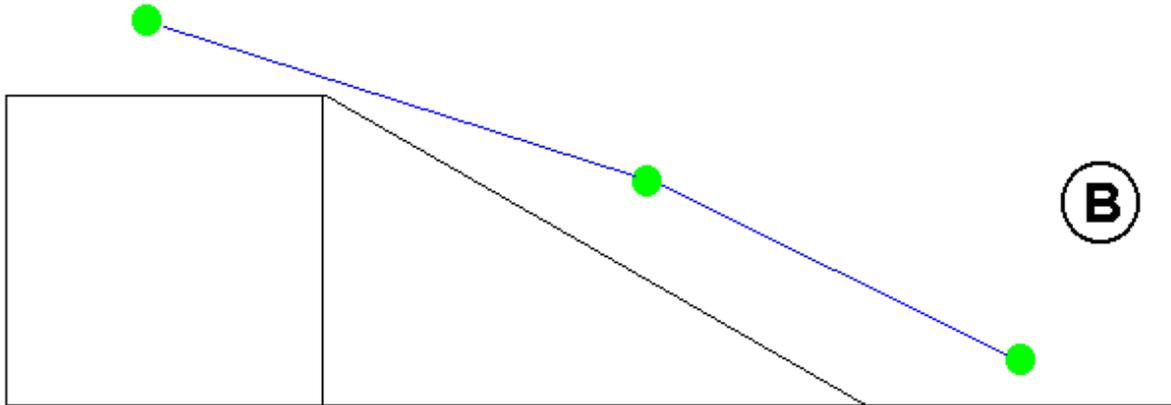
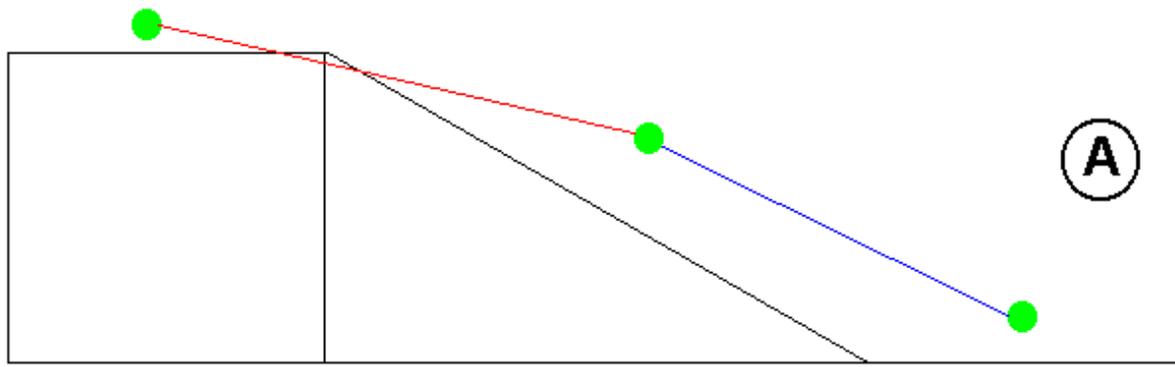
PATHNODE PROBLEMS

I encountered problems when I first used pathnodes. Bots would not go through the doors of my buildings and they would not go all the way up ramps. I had not read the [Guide](#) at this point and had no clue about viewing the BLUE and RED paths. Once I read the [Guide](#) things were clearer.

The problem with bots not going through doors was because the pathnodes on either side of the openings were too close together and had a RED path between them as in part A of the below diagram. Since it was not BLUE and "preferable" the bots were not going to use it while other BLUE paths were available. Moving the pathnodes further apart (as in part B of the below diagram) and re-defining the paths resulted in a BLUE "preferable" path between the nodes which the bots were more than happy to use.



The problem with bots not going all the way up ramps was because the pathnode at the middle of the ramp did not have an unobstructed path to the pathnode at the top of the ramp and they had a RED path between them as shown in part A of the below diagram. This meant bots would go partway up the ramp but would not choose to go all the way to the top since the path there was not "preferable." Moving the pathnode at the top of the ramp upwards (as shown in part B of the diagram) and re-defining the paths resulted in a BLUE "preferable" path between the nodes. This resulted in the bots choosing to go all the way up the ramp rather than stopping halfway up the ramp.



REACHSPEC NUMBERS - A BALANCING ACT

In my opinion you should define paths from the log window. This way you can observe the log window output as the paths are defined and make use of two values reported to you as the definition of paths ends. The number of reachspecs added and the number of reachspecs pruned will be useful as you optimize your network of path nodes.

Sample Output From Log Window

```
DevPath: Added 281 reachspecs
DevPath: Prune reachspecs
DevPath: Pruned 125 reachspecs
DevPath: All done
```

The number of added reachspecs will increase as you add more pathnodes, player starts and inventory items to your map. More reachspecs means a more complex network.

The number of pruned reachspecs is a measure of how many unnecessary reachspecs UnrealEd was able to remove from the network. This reduces the complexity of the network.

While the [Guide](#) does not specifically say so, it would seem logical that the number of reachspecs in the network will be equal to the number of added reachspecs minus the number of pruned reachspecs.

You must have enough pathnodes in your map in order to ensure that bots will move everywhere that you want them to in your map but you should not have too many pathnodes because you may have more than are necessary making the network more complex than need be.

By observing these numbers in conjunction with the color changes in your paths you can get a good feel for whether your adjustments helped or hurt. If an adjustment makes the added reachspecs decrease or the pruned reachspecs increase without making any more RED paths then your change simplified the network slightly, which can hardly be a bad thing. If an adjustment makes the added reachspecs increase or the pruned reachspecs decrease without making any more BLUE paths then your adjustment made the network slightly more complex which may be a bad thing (unless the adjustment did make some BLUE paths that you needed in which case added complexity may be the price you have to pay. There's the balancing act!)

FINIS

That's it. Once you've read and absorbed this document and the [Unreal Creature Care and Feeding Guide](#) and possibly the other pathnode tutorial / information mentioned above you should be ready to sit down and start making beneficial adjustments to your network of pathnodes.

Good luck with your tweaking.

Schlacht (dehags@airmail.net)

Contents (C) 1999 - Schlacht - If you copy or repost this elsewhere please be sure to include proper credit.

[Schlacht's Unreal Tournament Map Editing Stuff](#)