Creating Navigation Meshes in Team Fortress 2  
By Sleepy Al  
 (http://steamcommunity.com/id/sleepyal)  
Updated 6/24/12

So, you've got a custom level. Maybe you made it and you want to play test it with bots, maybe somebody else did but you want to play with bots on it for fun. Look at [(VDC)](https://developer.valvesoftware.com/wiki/Navigation_Meshes) first, then come back here to know what's specific to TF2 and what's wrong from the wiki.

Also, at the end of the paper is my custom config if you want it, It's got the most relevant commands and you can play test without rebinding, but it's awkward. You can always bind your own, look here for which keys to use: [(link)](https://developer.valvesoftware.com/wiki/Bind)

* When you've already done nav\_generate and you realise you've got spots on your map that don't have navigation paths but they should be there, use nav\_mark\_walkable THEN USE nav\_generate\_incremental. Don't bother with that quicksave stuff. Afterwards, you'll have a bunch of selected areas. Type nav\_select\_invalid\_areas to clear the selection (if you end up having something selected after that then just delete it because it's an improper area).
* Don't use nav\_jump. Just make a bi/unidirectional connection between the floor and the area the bot should jump to.
* As of 6/24/12, bots have a ton of trouble with CTF, and don't work at all in PLR or Arena maps. In CTF, the bots will all make a beeline to the enemy intelligence, shooting at people as they see them but ending up walking right past them, and when a bot grabs the intelligence, he can't shoot, but friendly bots will escort him. Bots won't bother defending the intel, and Engies rarely if ever build. Snipers and Spies will walk the same path to the intel, rarely sniping/stabbing and preferring to SMG/revolver. TC works, 5CP works, PL works, KoTH works. 3CP is wonky when two points need to be defended at the same time, as the bots will be indecisive and move between points constantly, destroying whatever Rngy buildings they've built when they feel like moving. Once a point is capped everything works well.
* Bots have trouble going underwater, avoid letting them fall in it whenever possible. For example, in Well and Freight the bots will end up getting caught under the bridges that go over the water, and sometimes just end up drowning. Swimming up the hole in the water near the middle point in Freight is impossible.
* Bots can't use or shoot buttons, they won't use lifts or launchers purposefully, and don't understand teleporters, so maps with those will not have well working bots.
* Bots don't understand trains. They will act as if they aren't there, either by trying to walk through them as they pass or getting hit by them.
* Make sure there's a path underneath every ammo and health kit, even if they're not reachable, as bots will attempt to get to them. So, if there's a log hanging over a point that has a small health kit on it, if there's no path there the bots will run in circles underneath it, so make one.
* When you nav\_generate a payload map, where the payload starts a path is made on top of the payload with nothing underneath. Delete that path and make one underneath the payload.
* Bots don't understand transparent walls, like fences or windows. For example, in Barnblitz if there's a sentry on the final point bots will attempt to shoot through the fence near the right as you enter the point trying to destroy the sentry but end up just shooting a wall.
* Try to avoid areas where bots will have to crouch. If there's a spot under a staircase, try to only have paths there where the bot can walk while standing up. Bots have trouble crouching, even if the area is marked to crouch, and might just try to go through the area while standing and then get stuck in place trying to enter.
* Don't do place names, that's only useful for CS:S
* Don't do this: [(IMG)](http://cloud.steampowered.com/ugc/578959652856087745/54A518B2BD20E20DC7002817DCD12083BE5BF022/) Do this: [(IMG)](http://cloud-2.steampowered.com/ugc/578959652856091150/626143B7E335B4AE3AD6B9C2839E10F68199F824/) ||| Bots will need to have a path on top of a railing to jump over it, unlike how the wiki states that the path needs to be like the former and the bots will eventually jump over it. I mean, yeah, they'll eventually jump over it, but it's just dumb to see bots huddled against a rail for 30 seconds until they climb it.
* Since ladders don't work properly in TF2, some map will use tiny yet extremely tall invisible playerclip staircases in front of the ladder to give the illusion of one. The nav\_generate won't pick these up, so you should create a path from the floor to the top of the ladder, connecting the path to the top floor where the ladder ends and the bottom floor where it begins.
* Use nav\_precise if you want a bot to not fall off of a ledge, to walk on a super tiny path you made, to enter a doorway, etc.
* Nav\_generate usually won't make paths on horizontal ladders going straight or diagonally. Nav\_mark\_walkable usually doesn't work either. You need to manually make the paths as tiny squares connecting to each other near their corners along the path if it's diagonal, or just a regular path if it's straight. Make sure to make it nav\_precise, and split a square in front of the path with precise so the bots can line up with the path.
* I don't know if tf\_mark SNIPER\_SPOT or SENTRY\_SPOT work, I've always assumed that Snipers and Engineers decide on the fly but I could be wrong. I haven't had enough time to test if they work or not but if they're there they've got to do something.
* Don't do this: [(IMG)](http://cloud-2.steampowered.com/ugc/578959652856367291/478EFBB6236D46773C8BDA9BCE5FF0BE517D7427/) Do this: [(IMG)](http://cloud.steampowered.com/ugc/578959652856356556/6DC9E95CCC38B36C3155062EF320991B09803F2E/) ||| Bots will try to drop through the floor if you do the former, and will end up walking in circles and jumping. When dropping from a ledge, a bot should continue walking while falling, so make the drop diagonal by making a unidirectional connection from the ledge to the floor a little bit in front of where the ledge ends. Make sure if a bot can't jump up to an area that the fall is unidirectional, as bots will attempt to jump if it's marked bidirectional. Nav\_generate incorrectly makes areas bidirectional all the time so be wary.
* If a bot needs to drop through a hole, make sure he lands on a square path that's directly center to the hole and does not go under the ledge or the bots will circle around and jump. If a path goes both underneath and in front of a ledge the bot will attempt to drop through the closest area, which is directly underneath, so make sure to split areas that do that.
* If a bot suddenly stops dead in its tracks when it reaches a ledge, that's because the path it's on goes past the ledge into the air. Make sure the path doesn't do that by splitting it so it ends before the ledge ends, then have it drop via a (uni/bi)conditional direction.
* If this happens: [(IMG)](http://cloud.steampowered.com/ugc/578959652856709922/B0E2AB714DAEE57ADD30D234A0C905924EA6A2FC/) Then nav\_split the parts so that when you do nav\_corner\_place\_on\_ground (without selecting the corners unless you're a masochist that wants to do each corner one by one) it will look like this: [(IMG)](http://cloud-2.steampowered.com/ugc/578959652856724558/D383E84BB29AF0019C1FFAF232CFD0F28CF78AA6/)
* Delete areas that bots can't fit into, like those silver barricade props that have tiny gaps in the back where a square path is usually generated between the poles. Bots will attempt to hide there, and end up trying to fit into there for a while until they get killed or something grabs their attention.
* Delete stuff like this: [(IMG)](http://cloud.steampowered.com/ugc/578959652856794958/D6A35DA6D580289C8BC779A97D76D17D49E8E9CD/) ||| The bots will try to cut corners and end up lower on the staircase instead of crossing over. Try not to have bots try to jump up part of a staircase from the floor perpendicular to the staircase, just have them walk around and up instead.
* Delete areas bots will never, ever get to, like the bottom of a cliff or past the playable area where no player can get. Make sure there are paths to walk if a player can feasibly get there, since crafty wrangler Engies can build teleporters anywhere and if a bot teleports to an area with no navigation path then he'll be stuck on the teleport.
* I don't have much experience with nav\_avoid, so I can't really tell if it works well or not, and I'm also not entirely sure if you need to mark staircases or not with nav\_mark\_attribute STAIRS, but I do it out of habit.

TF\_MARK

* Let's say you have a Badwater-esque Payload map. When you want to make something blocked until a point is captured, or blocked after one is captured, you need to use TF\_MARK BLOCKED\_(UNTIL or AFTER)\_POINT\_CAPTURE, then TF\_MARK WITH\_(FIRST, SECOND, THIRD, FOURTH, or FIFTH)\_POINT. With payload, the first point doesn't exist. Rather, it's where the payload starts. So the first point you cap is actually the second, and so on and so forth.
* With 5CP, the first point on the left (RED's last point) is FIRST, and the point on the right (BLU's last point) is FIFTH, and in between is SECOND, THIRD (mid), FOURTH.
* 3CP has A as FIRST, B as SECOND, and C as THIRD. Depending on how the mapmaker does the capture logic this may be off.
* I have no clue what TC does, so if you have an area that's blocked when a different area is contested make sure that the area where it's blocked is marked with nav\_transient If you play and it's not marked as blocked (with a dark blue perimeter inside the path) then make sure the path is 100% inside the blockage and either raise the path entirely with nav\_corner\_raise (make sure you don't use nav\_corner\_select or else you'll only raise one corner) or delete and remake the path.
* Regarding spawn rooms, don't use TF\_MARK UNBLOCKABLE. Instead, use TF\_MARK (RED or BLUE)\_ONE\_WAY\_DOOR and TF\_MARK DOOR\_NEVER\_BLOCKS. The area will still have the blocked blue perimeter inside the path, but the correct bots will travel through it and the enemy team won't try to enter it. Also, if the spawn's exit only, make sure the exit is unidirectional. Finally, make sure the path that connects the outside to the inside is underneath the door. If you just connect the outside to the inside with a uni/biconditional connection the bots won't know there's a door there.
* Regarding setup gates, use TF\_MARK BLUE(or red if it's like Well)\_SETUP\_GATE and TF\_MARK DOOR\_ALWAYS\_BLOCKS). Snipers, Spies and Engies on the Blue team will try to pass through the door anyways, that's an AI problem and you can't do anything about that.
* Use TF\_MARK DOOR\_NEVER\_BLOCKS on regular doors in your map. Meaning, the path underneath the door has this command.
* Use TF\_MARK DOOR\_ALWAYS\_BLOCKS if it's a special door that opens when a point is capped then never closes (unless another point is capped or something)
* I don't think there's a way to check what TF\_MARK commands you've done on a certain area, so if you're unsure just do TF\_WIPE\_ATTRIBUTES and redo them.
* When doing a TF\_MARK command, it will be done on the path you are looking at, unless you've marked an area, in which case it'll do it to the marked area.

Remember, save often. When you're done, make sure to nav\_analyze! Also, try not to change the paths while bots are running, it can cause crashes.

CUSTOM CFG

Ok, copy this text below and save it as navedit.cfg in your tf/cfg folder. When in game, type in console exec navedit.cfg.

bind "0" "nav\_corner\_select"

bind "6" "nav\_connect"

bind "7" "nav\_disconnect"

bind "8" "nav\_split"

bind "9" "nav\_mark"

bind "-" "nav\_corner\_lower"

bind "=" "nav\_corner\_raise"

bind "KP\_END" "nav\_precise" //it's num1 on the numpad

bind "[" "nav\_begin\_area"

bind "]" "nav\_end\_area"

bind "'" "nav\_delete"

bind kp\_downarrow nav\_merge //num2 on the numpad

bind kp\_pgdn nav\_splice //num3 on the numpad