This is for the benefit of newcomers to the modding scene to give them some insight into the BF2 editor and its features for changing gameplay objects and using the new single player tools. I will be showing you how to use the current navmesh area to set up your own custom playing areas and objects. This tutorial will focus on using the stock battlefield 2 map, "Songhua Stalemate." I'm currently navmeshing this map, so thought I would kill two birds with one stone :)

By Kysterama (Allied Intent Community War Mod and BFSP forums)
ALLIED INTENT WEBSITE
BFSP FORUMS

Note: If anyone knows German and can translate these tutorials for the site, I will gladly put up a German version as well. 22% of all visitors to this site are from Germany. Just contact me at - kysterama@hotmail.com if you want to help out our friends in Europe :)

Current Single Player Maps with 64 sized Navmeshes

BF2 Maps
- Fushe (Korben)
- Kubra (clive)
- Mashtuur (ky)
- Daqing (clive)
- Zatar (clive)
- Oman (clive)
- Songhua (ky)
- Cleansweep (clive)
- Dragon Valley (clive)
- Karkand (ky)
- Dalian (ky)
- Sharqi (ky)
- Wake (Wcc)

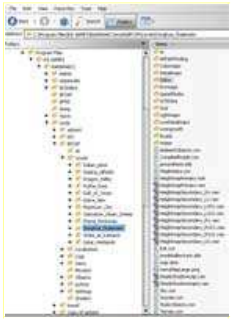
Special Forces Maps
- Surge (clive)
- Warlord (ky)
- Leviathan (ky)
- Mass Destruction (clive)
- Night Flight (clive)
- Ghost Town (clive)

PAGE 1 - Preliminary Setup

This is very important: Changing any files in your default Battlefield 2 mod directory will cause 'modified content' errors and you will **not** be able to join ranked servers. If you're on dial-up and don't care about multiplayer, still follow this step to keep your original install clean in case of mishaps during your modding.

- We are going to make a complete backup of your BF2 directory. Go to mods/bf2 and copy and paste it. When that's finished you should have 'Copy of bf2' there. This is going to be your untouched multiplayer and backup version if you completely screw up your files playing around in the editor. Don't umm and ahh...trust me - **DO IT.**
- The reason I'm not getting you to setup your own mod directory is that there are a few complications in getting a custom mod working perfectly in the editor, whereas the bf2 mod folder is 100% supported. I chose this path to make it as simple as possible to work for the 'layman.' If you ever want to play online just rename the bf2 folder we are modding to something like - "bf2_modding" and rename the 'Copy of bf2' back to its original 'bf2' name. Easy.

STEP 1 - Preparing the level folder for the editor



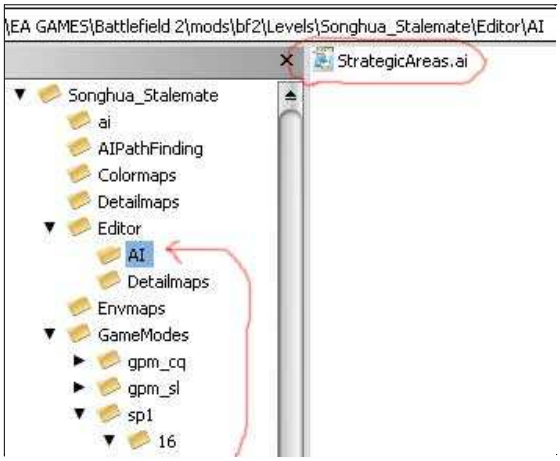
Click to enlarge

1.) Unzip '**client.zip**' and '**server.zip**' into the level's folder. Select yes when it asks if you want to overwrite files. All these files are currently 'read only.' so select all the files -> **right click** and choose **properties** -> uncheck the "**read only**" box and then **apply**, and then apply to all subfolders.

2.) Make a new folder called '**editor**'. Now browse to -> gamemodes\gpm_cq\64\ and copy **gameplayobjects.con.** Paste it into the **editor** folder you just created. This is Dice's original 64 sized multiplayer layout and we are going to customize it for Single Player. You can grab the 32 player one if thats your intention.

SIDE NOTE: For your information.

If you want to use or see an existing Strategic Areas from any of the 64 navmesh maps (from Korben, Clive or myself or even a sp16 one from Dice), simply make an **'ai'** folder inside the newly created '**editor**' folder from the previous step. Now go to **'gamemodes/sp1/16/ai'**, and copy the **'StrategicAreas.ai'** into your new '**ai**' folder. Now the editor will display them.



(Ignore the 'Detailmaps' folder. You won't have one because it's part of the navmesh

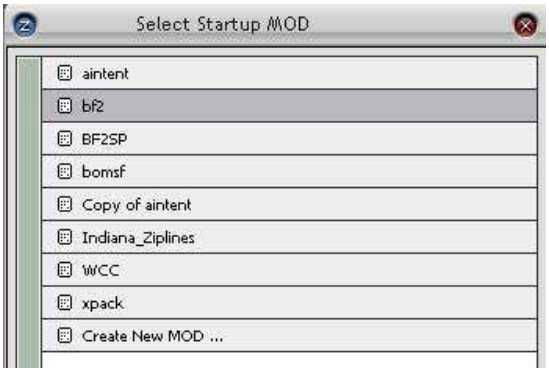
process.)

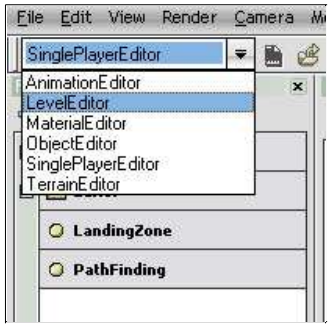
Click to enlarge

3.) Open gameplayobjects.con with notepad. The combat area is defined by all the coordinates at the bottom of the gameplayobjects.con file. We are going to delete the original combat area for the purpose of demonstrating how to create your own. Delete all the text below the line **'end if.**' Save and close.

STEP 2 - The editor

1.) Start up the bf2 editor. Choose **'BF2**'. When that has finished loading, change to the "Level Editor" in the drop down box at the top near the menus.





2.) Go to **File** -> **Load** and select the map you just prepared (for me, its Songhua Stalemate)

Click to Enlarge

3.) Once that has loaded you will see the map in all its 3d glory. Now for some preliminary steps to make your level editor experience easier on the eye.

- Select the **'Render**' Menu and uncheck '**Toggle Draw Fog**'
- Select the '**Render**' Menu - '**AI**' - and select "**Toggle Draw AI**" (this will enable you to see the strategic areas and neighboring when you start putting them in later)

EDITOR NAVIGATION TIPS

Hold down middle mouse button to 'look' around in the level.
Use the '**W**' '**A**' '**S**' and '**D**' keys to move around the level, much like in the game.
When used in conjunction with the middle mouse button you can 'fly' around the level to get to places you want to edit.
Scrolling the mouse wheel forward will speed up your movement while scrolling it back will slow you down. (You will see the 'spd %' at the bottom of the editor for your current speed.
You can change camera views in the "Camera" menu. Experiment with this, but the "Normal Mode" is the most useful.
The 'Render' Menu allows you to turn things on and off in the map. For instance if you are placing objects on the ground, it is a good idea to turn on undergrowth and overgrowth to make sure you're not spawning something into a tree!

Next Page

Page 1
Setup
Page 2
Objects
Strategic Areas
Page 3
Neighbours
Combat Area
Waypoints
Playtesting
Page 4
Navmeshing and Maya

Now we have the editor set up just the way we want it. It's time to get our hands dirty. Zoom your way around the map and get a feel for all the capture points and layout of the level. You'll notice all the CP's (Capture Points), Vehicles and Spawn points. Your goal may be to design a new layout from scratch with completely new CP's etc or use an existing one. The quickest method to do this is to grab a stock Dice gameplay layout and move it around to your liking. Now for a quick tour of the BF2 level editor tool bar. You will be using these tools constantly.

PAGE 2 - GAMEPLAY OBJECTS

 Select tool. Does it exactly what is says

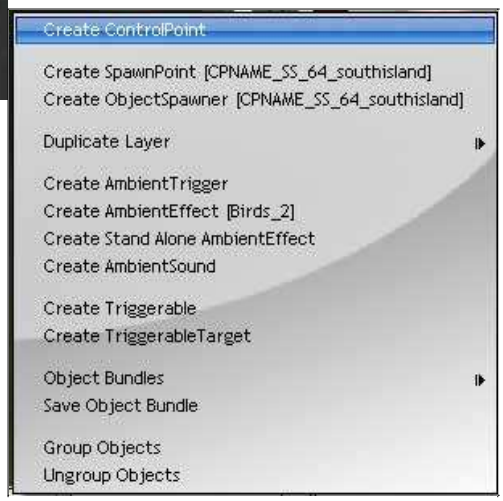
 Move tool. Use this to select objects and move them in up/down, left/right, back/forward

 Rotate tool. To rotate any object for its desired placement

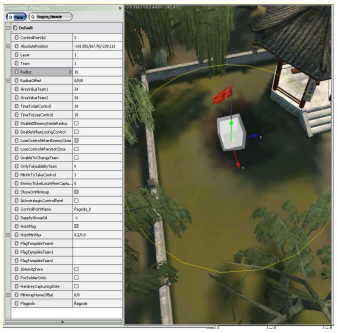
 Combat Area Tool - This tool defines the area of play

This page deals with creating your game areas from scratch. If you want to just use an existing Dice layout then skip to the bottom of this page to the heading **Strategic Areas and the Single Player Editor.** You can easily move any of Dice's original layout to anywhere on the map using the move tool.

How to add a new Capture Point from scratch



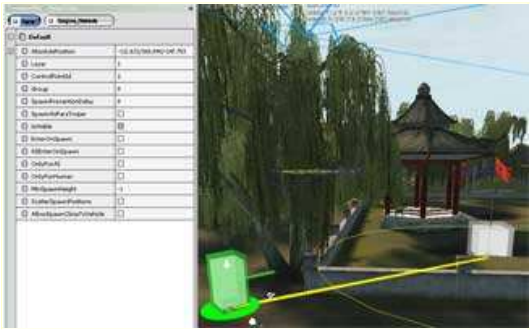
- Find a suitable area of terrain for your CP. Preferably somewhere that has some small amount of cover and somewhere that can accommodate spawn points that are not in a "line of sight" situation with the flag pole. Look at the Dice CP's and spawn points to see what I mean. It adds more intensity to the struggle if the spawn point is a little bit away from the flag and around a corner.
- Make sure the '**Select Tool**' or '**Move tool**' is selected (*You cannot place items when the rotate tool is selected*) and **right click** on the spot you want the CP to be. From the pop-up context Menu choose '**Create ControlPoint**' .Name your new CP. Try and make the name something location specific that makes sense.


Click to Enlarge

Now in the tweaker bar we have to set up the proper values for the game. We need to add values to two things specifically. The flag capture **radius** - between 10 and 15 is normal, but size it based on your terrain and surrounding objects. You can tell its radius visually by the blue circle surrounding the CP. Now the value to each team. This is how much the ai 'wants' the get to that flag. The average here in most maps is about 34 for each team. Team 2 always refers to the U.S. and Team 1 always refers to the opponent - either MEC or China. If you were to say, put in a higher value for the Mec and a lower value for the U.S. then the MEC would work harder to get and keep that flag than the U.S. would.

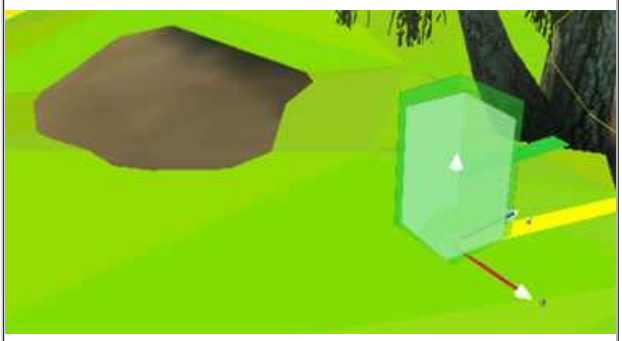
- If you want to make the base uncappable, checkmark the **'UnableToChangeTeam'** box
- If you want the game to start with this flag as neutral, set the team value to **0**. Change it to '**1**' for the flag to start as Chinese/Mec or **'2'** for the U.S.

Spawn points



Click to Enlarge

Now you will need some spawn points around. With the flag point itself still selected, find a suitable spot and right click again on the terrain, choose '**Create SpawnPoint [cp name]** ' it will automatically name it based on the Capture point's label so press okay. If you need to move it, select it with the move tool and drag it to a better spot. It should have a bright yellow line connecting it to the CP.



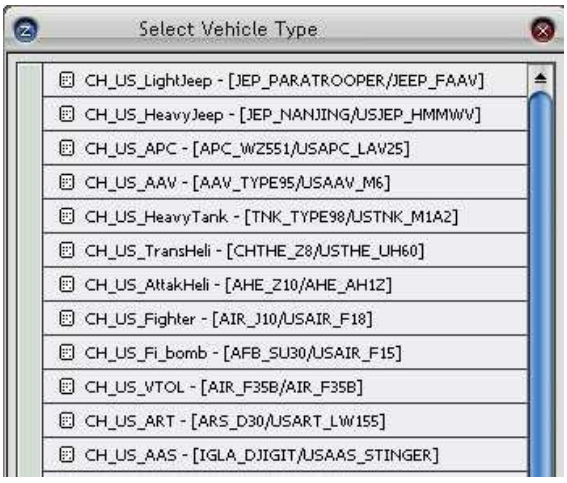
- We have to keep the bot navmesh in mind here. If you place a spawn point outside the mesh, the bot will spawn and instantly die. So go to the '**Render**' menu - '**ai**' - '**toggle infantry navmesh**'. Make sure your spawn point is sitting on the big green sexy blob.
- Now place 2 or three more spawn points around the place in good locations.

Spawn point ON the navmesh = GOOD

NOTE: You can tweak spawn points for bots. For instance, in some cases it is better to spawn a bot directly into a vehicle for certain types of gameplay. (especially on the aircraft carrier decks.) Make sure the spawn point is directly next to the vehicle and then checkmark the 'AiEnterOnSpawn' box in the tweaker bar and also the '**OnlyForAi**' box. You can also make spawn points "human" only by selecting the '**OnlyForHuman**'

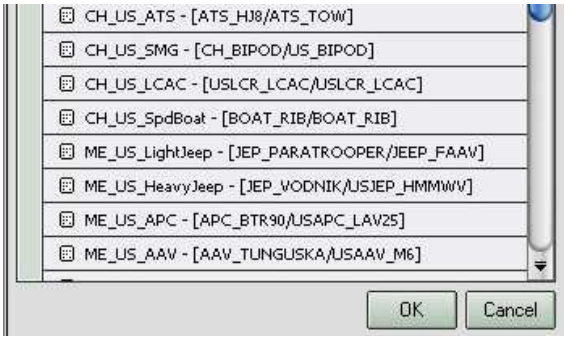
Spawn Point OFF the navmesh = BAD

Object Spawners



Object spawner is a fancy name for vehicles and commander assets.

- Make sure the CP is selected and right click on the area you want to spawn a vehicle and select '**Create ObjectSpawner**'. A list of object spawners will appear. (see pic to the left)
- The top half is China/US vehicles and the bottom half are MEC/US vehicles. Whoever controls the flag determines which army's vehicle will spawn there. Select a vehicle you want to spawn at that flag. (see below I chose the CH_US_APC.)



.....*Object Spawner selection menu*

Once again, make sure it is on the Vehicle navmesh ('**render**' menu - '**ai**' - '**toggle vehicle navmesh**')

- Create an object spawner for the UAV trailer, Arty and the Mobile Radar if this is to be the main base. You can move all of your objects about with the 'move tool' until you're happy with their placement.

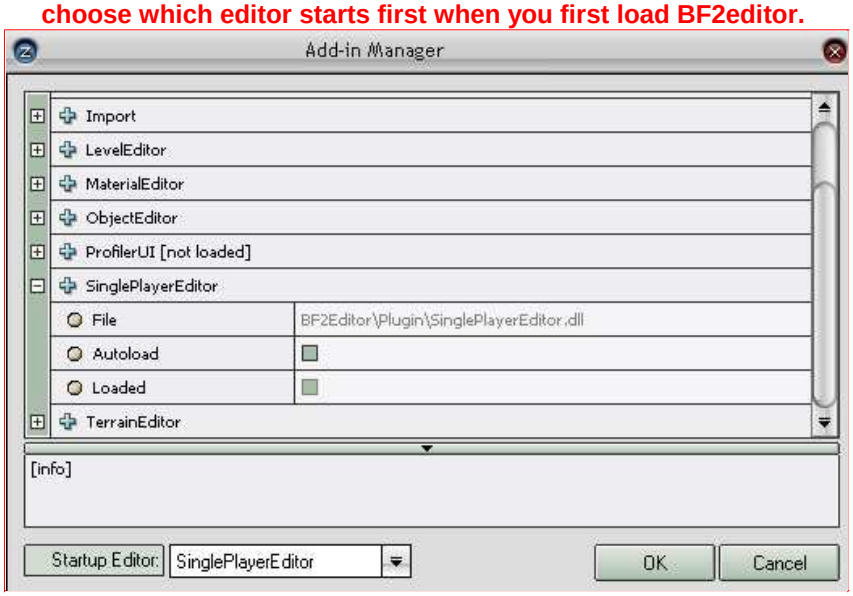
Side Note:

You will notice that objects don't interact well with anything that is not terrain. Eg. the aircraft carrier -> if you place a spawn point or object on the deck it will go straight to the sea floor. Same with buildings or roof tops. You can turn on "snap to object" in the 'Snap' menu. (Thanks for the tip NecronoM)

SAVE YOUR WORK NOW!

Strategic Areas (SA's) and the Single Player Editor.

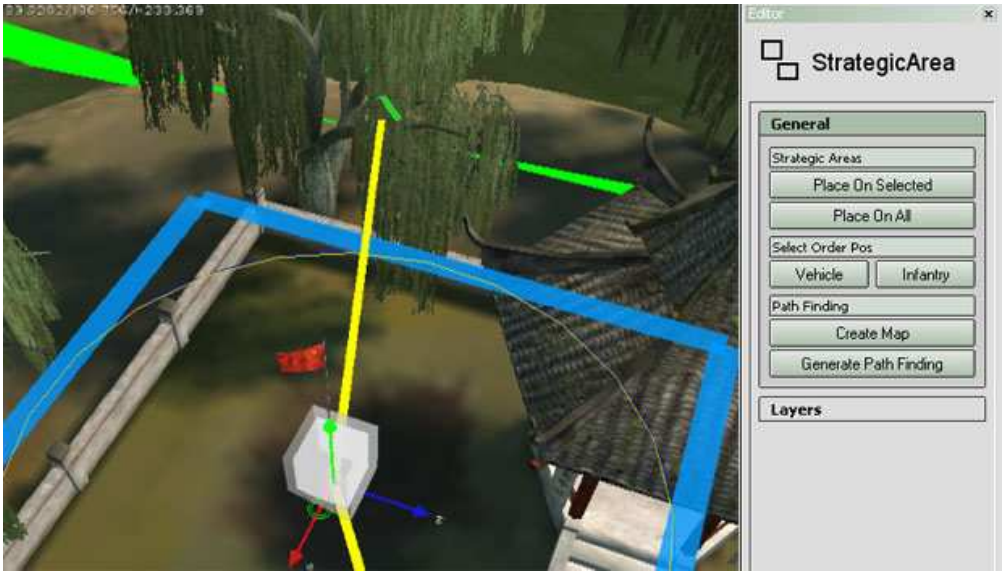
Setup note: If you don't see the single player editor in the editor drop down box, go to -> Tools -> Add-in Manager. Click the little cross next to 'SinglePlayerEditor' and checkmark the 2 boxes, 'AutoLoad' and 'Loaded.' In the 'Startup Editor' box you can choose which editor starts first when you first load BF2editor.



Now we have a Capture Point and a few spawn points and some object spawners. What the bots need now is motivation to come and capture the flag. This is where we get to use the Single Player Editor and place Strategic Areas.

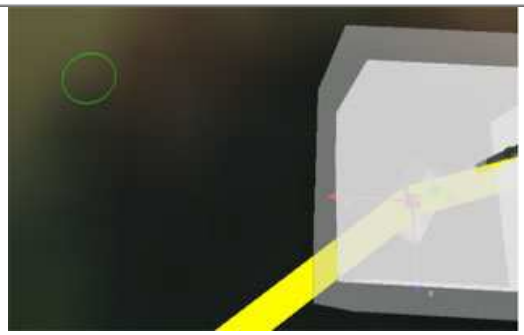
- Select the **Single player editor** from the drop down list

- Now you have a strategicArea toolbox on the right of the screen.
- Drag a small box around the base of the capture point so that it is selected
- While the CP is selected, click on '**Place On Selected**' from the right menu bar.



A brand new Strategic area placed on the CP

- You should now have a blue box drawn around the CP radius (make sure '**render**' - '**ai**' - '**Toggle Draw ai**' is checked if you don't see it.
- Now you need to place the infantry and vehicle order positions for the SA. These specify where the bots will stop when they come to cap a flag. Always put the infantry one inside the strategic area radius. The vehicle one can be outside the radius if you want the bot to stop, exit the vehicle and come into the CP's radius on foot.
- So click on the **'Infantry'** button - a box will say, '**click to place order position**'. Click **OK** and then click on the spot you want it to place the Order Position.



A placed Infantry Order Position next to a flag

Click okay and the click on the ground inside the CP's radius for the Infantry Order Position.

Now repeat the same process to place the Vehicle Order Position. This time click on the '**Vehicle**' Button, click **OK** and place it somewhere near the CP. (it can be inside the radius if you want the bots to cap the flag from inside the vehicle or outside the radius if you want them to dismount and cap the flag on foot - I find it better to put the vehicle Order Postion just outside the radius. Because passengers don't count toward flag-capping, its better that they all get out and walk to the flag.)

We've just completed an ai compatible and fully featured Control Point for BF2 single player. Repeat the above steps to create more control points, spawn points and object spawners throughout the map and you will be ready to go to the next step which is creating Strategic Neighbours.

Using Dice's Existing Layouts

If like me, you are using Dice's existing layout, you can easily select any objects in game , reposition them, delete them, rename them....whatever you like. I find it easier to start with a stock

layout like the 16, 32 or 64 gameplayobjects.con and make all my changes to that. Doing it this way, means that you can start immediately placing the Strategic areas on the existing Control Points. I added the above 'Starting from Scratch' Information for those of you interested in knowing the process when custom maps are made.

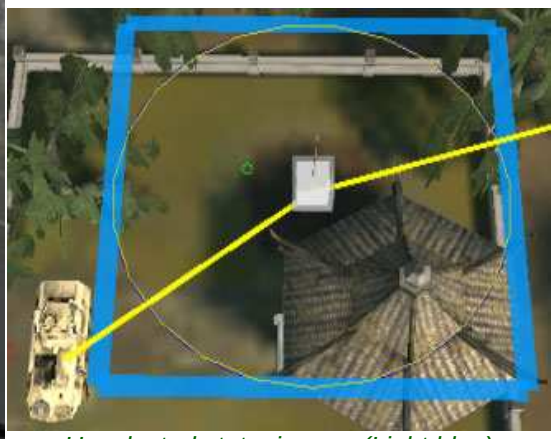
Don't forget to check that every spawn point and vehicle is on their respective Navmesh! AND SAVE SAVE SAVE!

PREVIOUS PAGE - NEXT PAGE

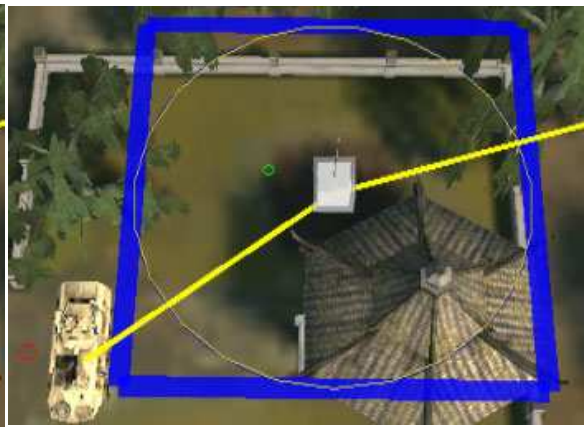
PAGE 3 - Strategic Neighbours

Now we'll deal with strategic neighbouring. It means exactly what is says. Any control point that is near the current control point needs to be specified as a neighbour. A neighbour tells the bots which is the next / closest CP to go to once they have finished capturing the current CP.

- Select the Strategic area bounding box by clicking on its edge. It will turn dark blue when selected.

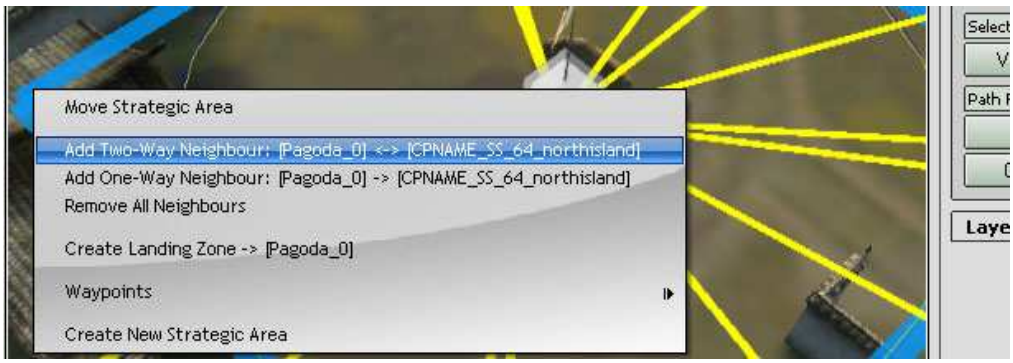


Unselected strategic area (Light blue)



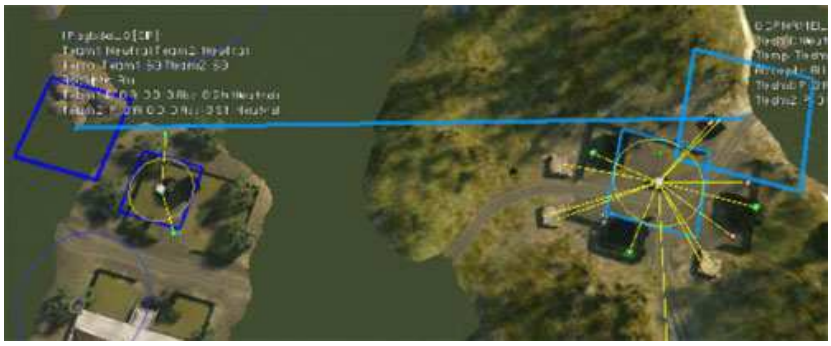
selected strategic area (Dark Blue)

- Once selected, go to the nearest capture point and right click on the **corner** of its Strategic area bounding box. This can be a little finnicky and may need a few tries before it works properly. A context menu pops up - Select **'Add Two-Way Neighbour: [CPNAME...'**



Right click on the corner of a neaby strategic area box to add it as a neighbour

If you haven't noticed, each Strategic Area has two blue bounding boxs - One at ground level and one directly above it in the sky. If the two strategic areas are connected by a blue line in the sky, then you have just successfully neighboured 2 Capture points together



Two Strategic Areas Neighboured to each other.

- Now perform the same operation to all strategic areas. It is fine to have multiple neighbours connected to one strategic area. Set this part up how you would like the bots to travel from one CP to the next. Its handy to hover above a flag and visualise "which ways do I want the bots to go from this control point?" you can create choke points etc using neighbouring. You can set up 'line-of-sight' or 'supply lines' type gameplay by the way you neighbour SA's. Or you could just simply connect every strategic area to each other so its just a frag fest of bots going everywhere at once.

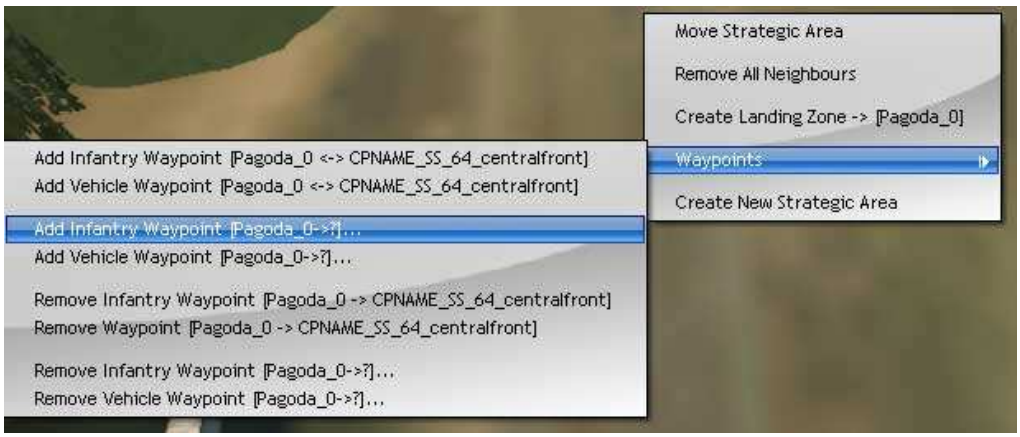
Multiple Stategic Areas Neighboured to each other

SAVE YOUR WORK! PLEASE NOTE THAT SAVING IN THE SINGLE PLAYER EDITOR WILL ONLY SAVE YOUR SINGLE PLAYER CHANGES. MAKE SURE YOU SWITCH BACK TO LEVEL EDITOR AND SAVE YOUR WORK THERE AS WELL!

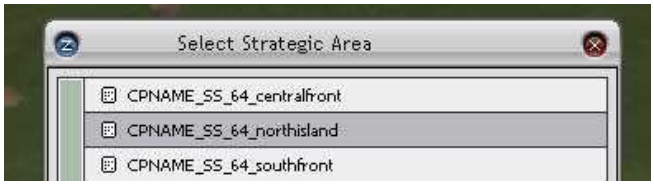
Bot Waypoints in the Single Player Editor

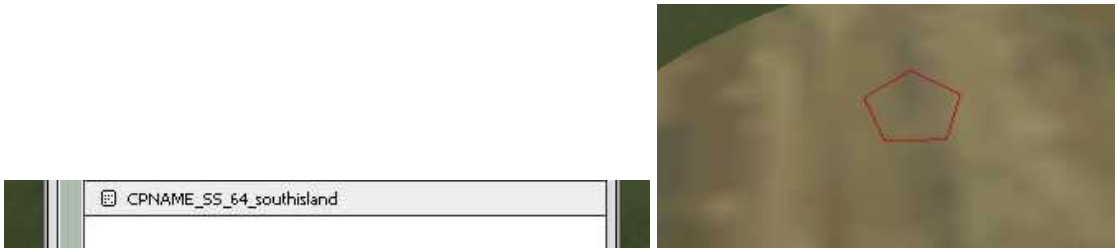
These are not strictly neccesary, as bots will usually find the most direct route to a CP by themselves. But placing waypoints gives you the ability to choose *sneaky* ways for the bots to travel to control points. This is done in the single player editor.

- **Select** a strategic area box. (the blue square around the CP)
- Now go to a point on the map between this strategic area and a neighbouring SA that you would like the bot to traverse to. Right click on the ground and select, '**Waypoints**', **'Add Infantry Waypoint ->?**...



- Now the CP selection box comes up, so choose the neighbouring CP that you are making a waypoint to. (In my example its the 'north island' CP)





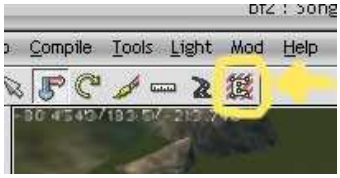
Select the neighbouring CP from the list and then the infantry waypoint will show on the terrain.

- The bots will now go from my 'pagoda' CP to the north island CP via way of that waypoint.
- Follow the same process to place the Vehicle waypoints
- **AGAIN - Make sure the waypoints are on the infantry and vehicle navmesh respectively. AND SAVE!**

Combat Area

- The combat area is the coordinates that specifiy the area in which the player can go. If they go outside this area (the red stripey no mans land on the mini-map) then the commander threatens to shoot you for desserting.
- **NOTE:** This step is integral for creating navmeshes. The combat area you specify tells the navmesh generator which area it will create a navmesh in.

- Switch back to the **Level Editor**
- Select the **Combat Area tool**



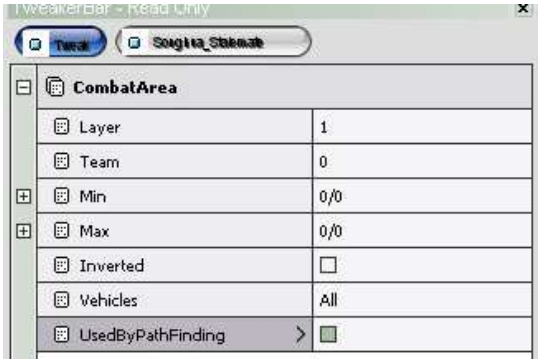
- **RULES**: The combat area MUST be placed down in an **anti-clockwise** direction!
- Right click an area where you want the Combat area to start - select '**add control point**'

Add the first Control Point

- Now work you way in an anti-clockwise direction around the border of your combat area, right clicking and placing control points for the combat area. it will draw a line from point to point until you come back full circle to the starting point. Right click on the first point and it will ask if you want to close the combat area - Of course you do! **(Note: my example is rediculously small just to demonstrate how to do it. In reality, the Combat area should surround the entire map and every CP.)**

Place control points like 'connect the dots' in an anti-clockwise direction. When you get back to the first point you placed, it will ask if you want to close the combat area. Click 'Yes'

- **IMPORTANT**: Once the combat area is closed, go to the tweaker bar and checkmark the box, **'UsedByPathFinding'**



SAVE YOUR CHANGES!.

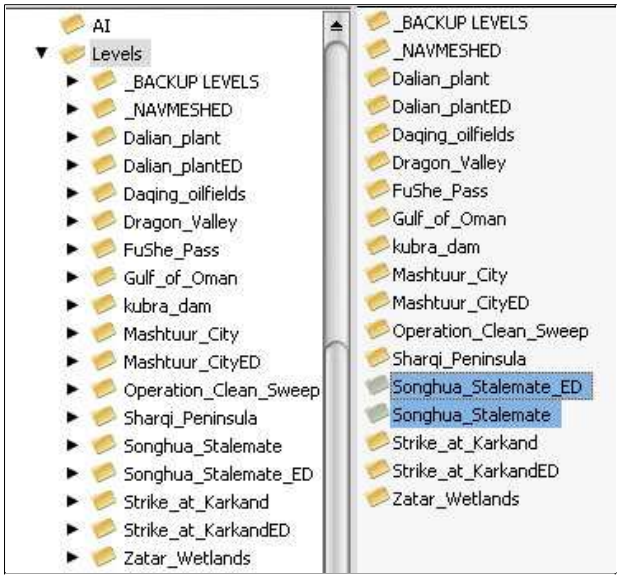
Final Packaging and Playtesting

Congratulations, you have just made a fully fledged single player ready map.
There is just one final step to get it into the game and playable. Because this level is completely unzipped it won't load in the game. You need to re-package the server.zip. The editor is still unfortunately buggy in this regard and gets some server.zip files mixed into the client.zip. Luckily for you, this whole time we have only edited 2 files from server.zip. - Gameplayobjects.con and StrategicAreas.ai.

- So now you have a level directory completely unzipped and unplayable in it's current state. Here's what I do. (I'll be referring to Songhua Stalemate from now on as the level I'm editing. If you're editing a diferent level, then just use your level name obviously)

Close the editor

- Rename your level folder to 'Songhua_Stalemate_ED' in windows explorer.
- Navigate back to your original 'Copy of bf2' mod directory (Lucky we backed it up!) and copy the original Songhua Stalemate level folder back into your BF2 mod level folder. So now I have Songhua_Stalemate_ED and Songhua_Stalemate next to each other.



- Open the server.zip in the Songhua_Stalemate. Navigate to - Gamemodes\Sp1\16 and leave it open to the side of your screen like in the picture below.
- In windows explorer, navigate back to your Songhua_Stalemate_ED folder. Go into the editor folder and drag and drop the gameplayobjects.con and 'AI' folder into the open zip file overwriting the files within.

Click to Enlarge

- Now all your changes are in the zip ready to play!

Start up a game and test out your map in single player. Keep an eye out for stange bot behaviour and keep a note pad ready to jot down the bugs you find in diferent parts of the map so you can fix them in the editor later. If you have to load up the map in the editor again, simply rename the play version level folder to something like 'Songhua_Stalemate_PLAY" and then rename the "Songhua_Stalemate_ED, back to plain old "Songhua_Stalemate."

The editor and game won't open the level unless it has its original name.

PREVIOUS PAGE - NEXT PAGE

Current Single Player Maps with 64 sized Navmeshes

BF2 Maps
- Fushe (Korben)
- Kubra (clive)
- Mashtuur (ky)
- Daqing (clive)
- Zatar (clive)
- Oman (clive)
- Songhua (ky)
- Cleansweep (clive)
- Dragon Valley (clive)
- Karkand (ky)
- Dalian (ky)
- Sharqi (ky)

Special Forces Maps
- Surge (clive)
- Warlord (ky)
- Leviathan (ky)
- Mass Destruction (clive)

Now onto the Navmeshing side of things. I must warn you, this is not for the feint of heart and requires a degree of knowledge in either 3d studio max or Maya. I'll tell you right from the outset, that this tutorial will focus on Maya as I have very little 3ds Max experience. I hope to get some support for a 3ds max tutorial from the helpful people at BFSP forums soon. Now this whole process was cleverly worked out due to a collaborative effort between many people. So thanks are in order to Keith (Korban Dallas), Clivewil, Necronom, Outsider and mschoeldgen for all the input and sharing of their ideas to get this navmesh process worked out within days of the editor's release.

A NOTE ON MAYA: The Personal Learning Edition does not include the wavefront OBJ plugin and therefore will not allow the importing of Navmesh files at all. You must obtain a full copy of Maya for this tutorial.

PAGE 4 - Navmeshing

Believe it or not, the first few pages of this web site are actually preliminary steps for setting up a navmesh, so if you have followed it this far, then you're halfway there.

Lets start with a checklist of things you must have in order to get to this next step.

1.) A copy of Python installed
2.) At least 2 Capture Points (flags)
3.) Strategic Areas placed on the Capture points that are neighbored.
4.) A Closed Combat Area (Placed anti-clockwise with the 'Used by pathfinding' Checked.
5.) An 'ai' folder in your level directory containing an 'ai.ai' file. (can be copied from an existing single player bf2 map)

SETUP

Here's what we need to do in order to get the ball rolling.
All the navmesh related files are in **Program Files\Ea Games\Battlefield 2\Navmesh**. I suggest sending a short-cut of this folder to your desktop to save a lot of time in the future. Its also handy to have a shortcut on your desktop to the bf2 levels folder as well.

First things first - Makes sure you have a copy of python installed on your computer. The link for it is in the above checklist. None of this will work without it installed.

Now edit the file (open with notepad) -> **GenerateNavmeshLocal.py** located in the above directory and change the mod that you are working in. As an example we are going to navmesh Mashtuur City from scratch so you get a sense of how to mesh a map that has no single player support at all. So at the top of the generateNavmeshLocal.py make sure 'bf2' is the specified mod.

```
import os
import sys
import string
# Change the line below to the mod you are working on
mod = "bf2"
```

Save and close the file. Remember that if you're trying to mesh a map that is not in the bf2 level folder. change the mod name in GenerateNavmeshLocal.py to the correct mod.

SETUP PART 2

There are 2 files in the **Program Files\Ea Games\Battlefield 2\Navmesh** folder that have to be copied into **Program Files\Ea Games\Battlefield 2.**

They are **saveQuadNoP4.con** and **exportGTS.con**. Copy them there now. This is not strictly necessary as you can do the same operations that these files do manually in the editor, but for the time being lets just be efficient and let the program take care of these steps.

Now we're ready to make a navmesh.

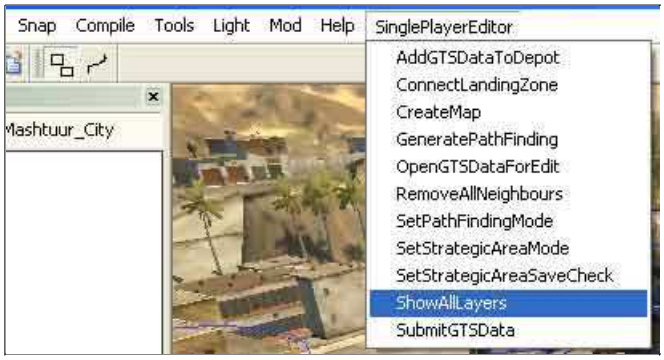
Part 2 : Navmesh Generation - Outputting gts data.

So, we're going to do Mashtuur City. This was the first level I ever navmeshed and will present us with some mesh problems. I though this would be a good map to tackle so I can show you how to overcome these problems.

I've already setup the map in the editor with **Strategic Areas**, **Vehicle and Infantry order positions** and **bot waypoints**. The strategic areas are **Neighbored.** I've enclosed the level with a **combat area** that was placed anti-clockwise and has **UsedByPathfinding** checked in the tweaker bar. These steps are very important, especially the combat area. What you define as the combat area actually defines the area that will be navmeshed. (all of this is covered in the first three pages on this site)

One last step is to copy an 'ai' folder from an existing single player map. So open any SP bf2 level's server.zip and inside you will find the folder, '**ai.**' In it is the file '**ai.ai**.' Copy this folder now into your Mashtuur City Level Folder. (Yes, the Mashtuur folder that has the fully unzipped client.zip and server.zip in it.) Now if all has gone well to this point then exporting gtsdata will be a breeze.

From the SinglePlayerEditor drop down menu, select **'ShowAllLayers'**



Now in the right side bar press the '**Generate Path Finding**' button.

A few moments will pass while the editor creates the necessary gts data that the navmesher needs to create the infantry and vehicle navmesh. Once it is done, you should see a whole bunch of gts data in **levels\Mashtuur_City\GtsData\Meshes.**

Now I want you to select and copy the GtsData folder. Now navigate to -> **Program Files\Ea Games\Battlefield 2\Navmesh\work** and create a new folder called '**Mashtuur_City**.' Now inside the Mashtuur_City folder paste the GtsData folder. Your file structure should look like this -> **Program Files\Ea Games\Battlefield 2\Navmesh\work\Mashtuur_City\GtsData\Meshes.**

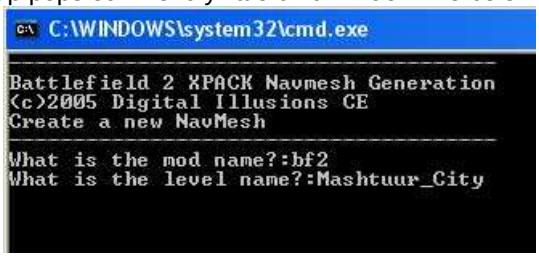
Close the editor.

Part 3 : CreateNavmesh.bat

Yes it's time to make the actual navmeshes! This process takes many hours depending on the size

of the level and the complexity of the terrain and objects within it. Fortunately, this part is probably the easiest out of the whole process :)

Go to **Program Files\Ea Games\Battlefield 2\Navmesh** and double click the **CreateNavmesh.bat.**
Up pops our friendly little cmd window like below.



In mod name, type '**bf2**' and in level name type, '**Mashtuur_City**" (Has to be the exact same syntax as the actual level folder name.) Press enter and the little navmesher goes off to do its magic. Now go make a cup of tea, watch a movie or two, have a nap etc etc. This process eats up 99% of your pc's resources for quite some time, so you may want to do this part over night while you sleep.

Brace yourselves, because after this it starts getting time consuming and tricky as we work in Maya to get a vehicle and infantry navmesh fully working in-game

Part 4 : Navmesh Rules

Okay, lets talk about the rules involved in a navmesh. If any one of these rules are broken, your navmesh will not work in the game and will cause a crash to desktop on load.

1.) Must be one island .

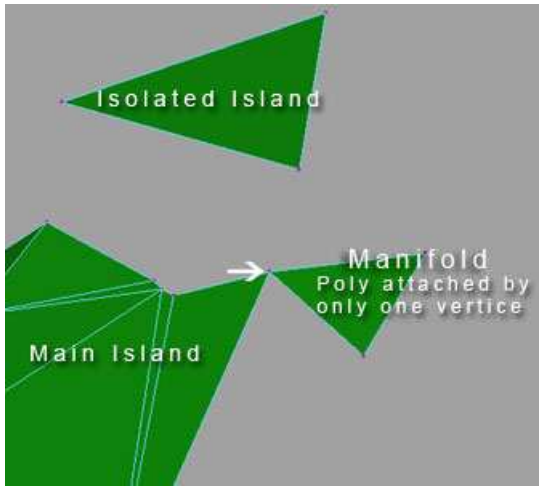
This means just what it says. Because the navmesh is simply a 3d copy of the terrain in which it covers, it is easy to visualize it as an 'island.' There can **not** be two or more separate pieces of navmesh, or 'islands', in the level. I'll go into this in more depth as we work through the Mashtuur obj files in Maya.

2.) Every Polygon must be a triangle.

By default, the navmesh generation process will never produce a poly that isn't a triangle, but it is worth noting that if you are creating extra mesh by hand, it must always be subdivided back to triangles. This will also be covered in depth later.

3.) No Manifold Poly's.

In layman's terms this is when a poly is attached by only one vertice to the main mesh like in the example picture to the right.



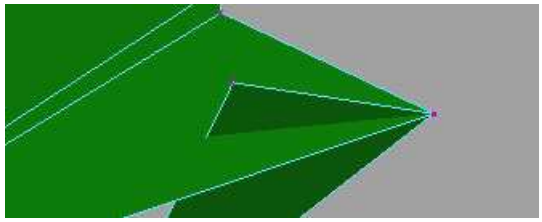
An example of the main island, with a floating isolated island and an illegal 'manifold' polygon attached by only one vertice. These are very easily fixed in Maya. A poly or 'face' must always have one side connected to the mesh, never just one corner.

4.) No Floating vertices.

Also pretty rare in my experience and only really turn up if you follow the wrong processes when deleting and/or editing parts of your mesh. A vertice is just one of the little purple dots at the corner of each triangle in the picture. Just make sure there are none of these floating in space.

5.) Must be non-self-intersecting.

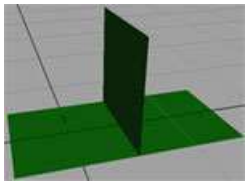
In laymen's terms - the mesh must never pierce itself. When dragging vertices around for an hour



its quite easy for this to happen. As shown in the picture a vertice has accidentally been placed up through another face. So this is now classed as self intersecting.

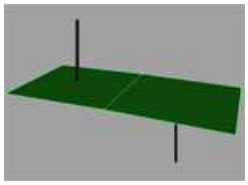
BF2 does not take kindly to this sort of barbaric, face piercing behavior!

6.) No Double Edges, Vertices or Faces.



- No double edges. See in the example how more than 2 faces share an edge. This is not allowed. Also, Double vertices are not allowed. A vertice must never be in the middle of an edge, it must always be a corner of a triangle.

7.) No Opposite Facing Normals



Another rare one but worth mentioning. This is when two faces are connected to each other but actually face in opposite directions. Also bad, and only really happens when you're up till 4am every night, eyes half closed and you are welding vertices together haphazardly with an almost drained rechargeable mouse, while in the midst of a micro-nap.

That about sums up what **NOT** to do to navmesh geometry. Most of these are caused by the mistakes of the user. The only ones that the navmesh generator usually gets wrong are floating islands (isolated poly's.) So with all that in mind, we're now going to jump in Maya and fix our first navmesh.

Part 4 : Maya Setup

First lets setup Maya. We need to enable the wavefront OBJ plugin first of all. Go to **Window -> Settings/Preferences - > Plug-in Manager.**

Now tick the **'loaded'** and **'auto load'** boxes for the **'the objExport.mll'** plugin.

This both sets up the OBJ plugin for our current session and each new time you start Maya in the future as well.

Well that's it for the Maya setup. Now we're ready to import our navmeshes.

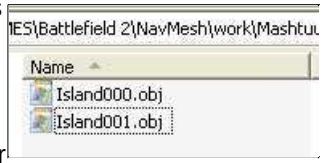
Part 5 : Importing the Navmesh's

BACKUP YOUR OBJ FILES!
Backup your Navmesh\Work\Mashtuur_City\GTSData\Output Folder now!

These files are the ones the navmesh generator took hours upon hours to create. If you accidentally mess up your obj files, you don't want to have to wait for the navmesh generation process to complete them all over again. So backup this folder so you can always come back to it if need be. What I
do is just copy the output folder and paste, so I have 'copy of output' and the original 'output' folder side by side. Very quick and easy.

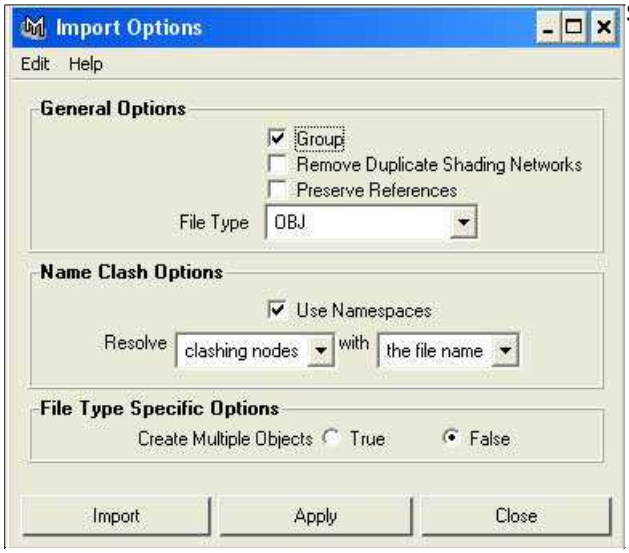
So by now you should know there are 2 separate navmeshes. The vehicle and the Infantry. One of the best and most ridiculously simple ways to check if either of these 2 navmeshes have floating islands (i.e. two or more pieces of mesh that are not connected to each other,) is to simply look in **Navmesh\Work\Mashtuur_City\GTSData\debug\islands\infantry** and **Navmesh\Work\Mashtuur_City\GTSData\debug\islands\vehicle**

If each folder only contains one file then you have no extra islands and possibly no editing to do. If there are 2 or more files then you know for sure that the navmesh needs editing. In the picture to the right for example, you can see two island files in my infantry debug folder. This indicates an unattached island that I need to either delete or attach to the main mesh. (The file size will help you determine which is the main mesh - the bigger of the two.)



Step 1: click on the **File -> Import -> 'Setup Box'**

The setup box I'm referring to is the little highlighted square to the right of 'Import.' Clicking on this little square brings up the import options dialogue box



Step 2 : Make sure **'Group'** is selected and the File Type is **OBJ.** Just Make sure that your import Options look identical to mine on the left.

Then click Import and browse to -> **Navmesh\Work\Mashtuur_City\GTSData\Output**

In this folder there are 4 main Obj files to choose from. **Infantry.Obj, Vehicle.Obj, Preopt.Obj and Manifold.Obj.**

If you click the "set project" button at the bottom, you can set the default folder every time you want to import something. I chose my project folder to be the **'work'** directory.

Import the infantry.obj now.

Part 6 : Maya Navigation.

Now for another learning curve. I'm just going to teach you what you need to know to edit a navmesh. Please remember this is all from my personal experience and is the one way I know how to get the job done. I'm not a Maya expert by any means.

1.) Movement in Maya
Alt + Left Mouse Button = Rotate the scene
Alt + Middle Mouse = Move Up, Down, Left Right
Alt + Right Mouse Button = Zoom in and out
Scroll Mouse Wheel = Zoom in and out as well.

2.) Operations in Maya
There are four types of polygon operations we need to know how to switch too.
Vertex (which are vertices,) *Faces, Edges* and *Object Mode*. When the navmesh is imported, we can simply switch to any of these by first selecting the mesh by clicking on it. Then hold down right mouse button and a context menu pops up. With right mouse button still depressed, drag the

mouse in the direction of the the poly edit type and let go. (For some reason I can't take a screen shot of this as the context menu does not get captured.)

3.) Handy Shortcuts

F = **zoom** or **F**ocus in on a selected item. Handy for navigating to a particular spot.
Main usage for us : *Navigation*.
By selected a vertice for example and pressing 'F,' you not only zoom in on that part of the mesh, but your camera's pivot point changes to what you have selected. (This zoom and focus method can be used while having either a vertice, edge or face selected and comes in VERY handy.)

W = **Move Tool**. Brings up a modifier tool on whatever you have selected.
The move tool is the crux of your navmesh editing. While having either a vertex, edge or face selected, you can move it in any direction in 3d space.

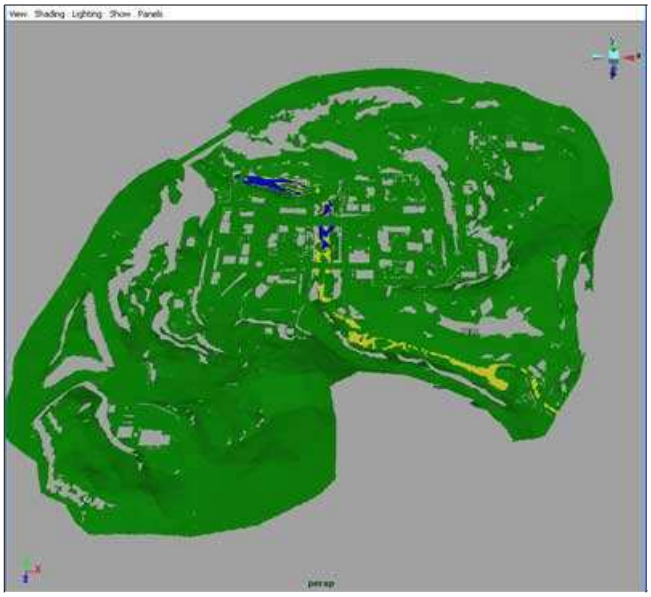
E = **Rotate.** Brings up a Rotate tool on what you have selected.
R = **Scale.** Brings up the Scale tool. (Make objects larger or smaller)
There are also buttons on the left sidebar of the screen Like below which activate the same features.

We'll be mainly using the Move Tool '**W**' and the Zoom-in Shortcut '**F**'

Okay so you've imported the navmesh and can't see anything in the scene yet? This is because the mesh is created away from the worlds zero point. (being the little grid you see when you first open Maya.) So with your mouse wheel **scroll backwards** until your navmesh shows in the scene.

Right now, the perspective view port (what you're looking at) is in wire frame mode. Click the '**Shading**' menu and select, **'Smooth Shade All'**

Behold your navmesh! The green blobby crutch, that bots feet love to touch.



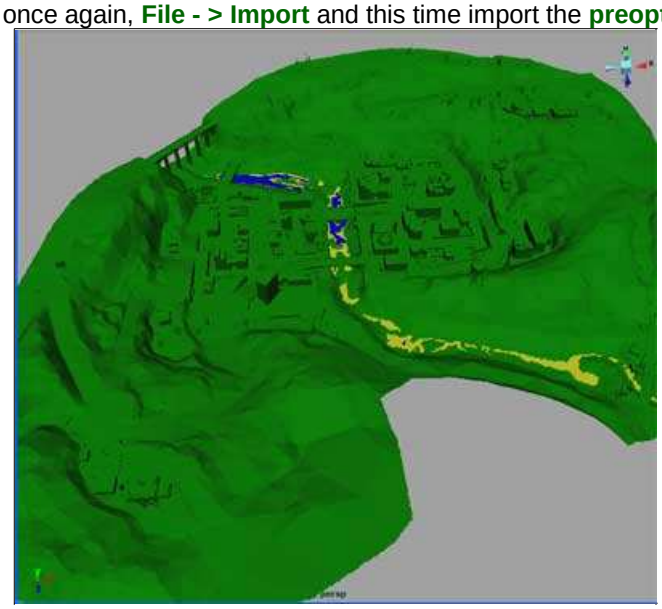
Practice the navigation techniques mentioned earlier until you get comfortable moving through Maya's 3d space, and zooming in on a different navmesh parts.

Part 6 : Having a visual reference - The Preopt.obj

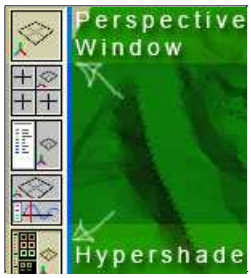
Well the mesh looks pretty but it is sometimes hard to see which part is which in a level without some visual frame of reference. I should also point out that the level is actually mirrored back to front when you import it into Maya. This is normal. The left and right get swapped. Don't worry, it exports back out correctly when we're finished.

So when you imported the infantry navmesh, you may have wondered what the other OBJ files are for. Well the **preopt.obj** is a perfect 3d rendition of the level and its static objects. This is perfect for us because we can import it into the scene right next to the navmesh to make sure that the navmesh covers all the parts of the level that we want it to.

So once again, **File - > Import** and this time import the **preopt.obj**



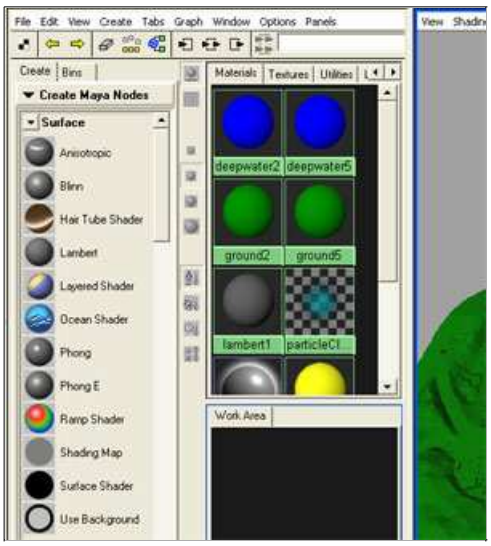
One small problem though is that the preopt has the same material as the navmesh making it one big green blob that makes it difficult to tell the two apart. There are a few ways to get around this, but the one I like is to simply change the colour of the preopt.



To change the colour of the preopt we need to open the **Hypershade** Window. The button for this is located on the left of the screen near the bottom as shown in the screenshot.

Note: Remember this for later. To get out of the Hypershade section, you just have to click on the **Perspective** button shown in the screenshot.

So now lets click the **hypershade button** and change the preopt's colour so we can see what the heck we are doing.



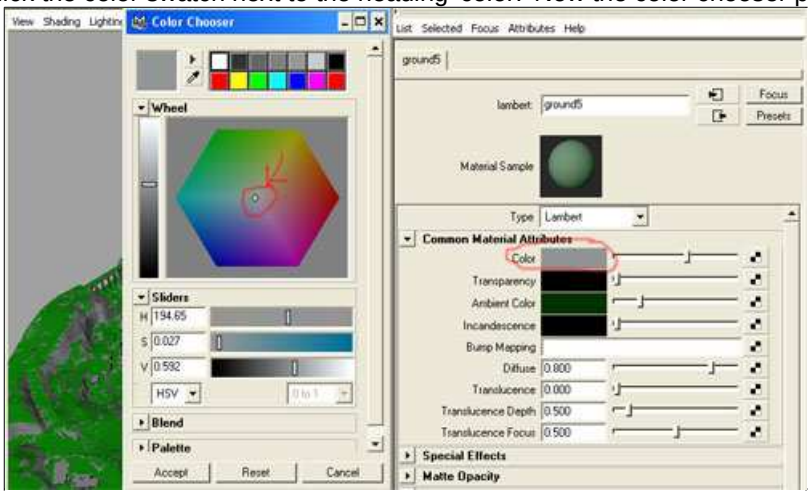
Now the Hypershade is showing two sets of the same material. One is the preopt and the other is our infantry.obj. The infantry.obj always contains these three materials which you should not change.

1.) deepwater2 - Blue
2.) ground2 - Green
3.) water2 - yellow

The preopt.obj is given the same materials but ending in a '5' instead. So we are free to change these.

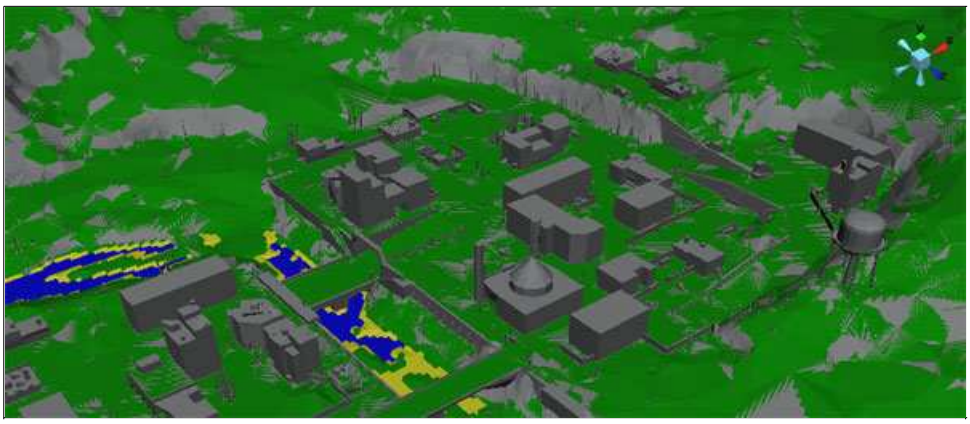
Click the **ground5** material. Now on the right of the screen pops up the ground5 common material attributes as shown below.

Now click the color swatch next to the heading 'color.' Now the color chooser pops up.



Now just click anywhere on the colored cube to pick a desired color. I always go for a nice neutral grey color as the green infantry navmesh shows up nicely on this.

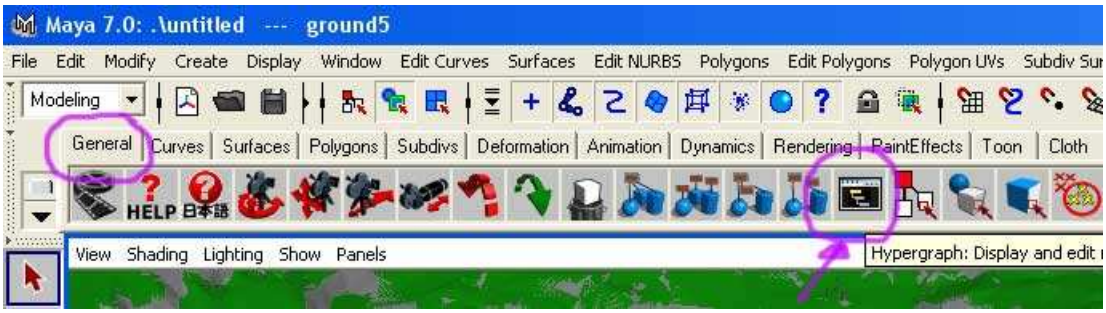
Now you can close the **Color Chooser** window. And click the **Perspective Window** button as show
a few screenshots back, to get back to our main view. The color attributes are still open on the right for the time being. You can play with the brightness of the color (the little slider next to the color swatch) and also the transparency if you feel it helps you work with the navmesh better.



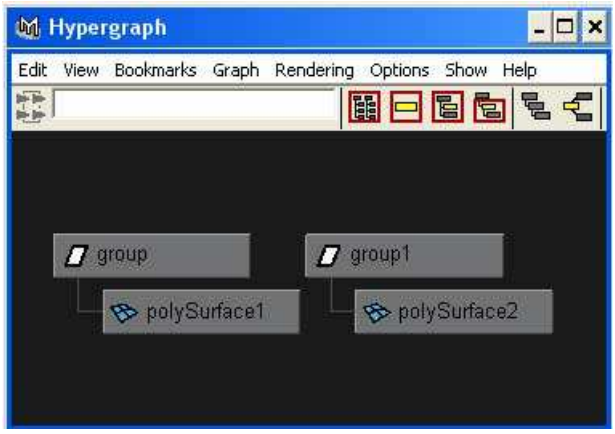
So now we can see quite clearly, everyplace that a bot can walk in the Mashtuur level. Just so you know, the deepwater material (blue) means bots have to swim there. The water material (yellow) shows that it is just shallow water and the bots can walk through the water.

There's one last thing you need to know about Maya that will make things a lot easier. Its called the **Hypergraph.** Opening the Hypergraph shows you at a glance what is in

your scene. The joy of it is that you can select the objects from this window without having to hunt through the scene with your eyes to find the pieces. This comes in very handy when there are many small islands like rooftops etc that are near impossible to spot with the naked eye.



Click the general tab and then the Hypergraph button. Now the Hypergraph window pops up.



You can now see the 2 meshes that we have imported. The first one, **polysurface1** is the infantry navmesh while the **polysurface2** is the preopt.

The same maya navigation applies to this window. You can zoom in and out with mouse wheel and move them around with **alt+Middle Mouse button**. You can even select one and press the **"F"** focus-zoom shortcut and the one you selected will zoom to center screen.

The next best thing about this window is that you can right click on any of the objects and then from the context menu select '**Hide**'

Try this with the preopt (polysurface2). The preopt will be hidden and only the infantry mesh will be visible. This is handy when you want to focus on a certain section of the navmesh and don't want the preopt in the way. To get it back just right click on the polysurface2 object again in the Hypershade window and select **'Show'**

For even more options to do with the Hide and Show command, experiment in the Display Menu at the top of the screen.

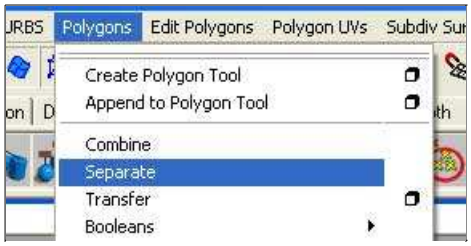
Okay, so now we know how to import the obj files, change colors, navigate around the scene, and use the Hypergraph window effectively. Lets get into the meat of navmesh editing.

Part 7 : Editing the Navmesh - Merging Vertices

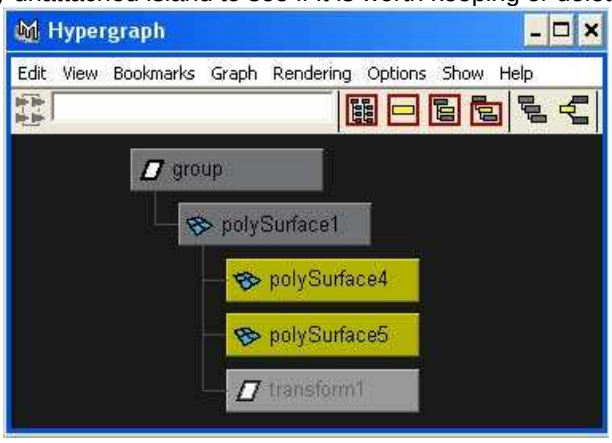
STEP 1: Identify the isolated islands.

Okay, before we start, lets **hide the preopt** so only the navmesh is visible. Our first step is to identify the islands in the scene and decide what to do with them.
Earlier we could identify that there was more than one island by looking in the actual debug folder under the navmesh directory. But when we look in the Hypershade, it appears to be one selectable object only. Why is this?

It's because the mesh imports as one combined object. The islands are still there and you'll see them if you know where to look, but for them to become separate we need to do a '**seperate**' command (funnily enough.) Our preopt is hidden right now - so click on the main infantry navmesh so that it is selected, (either in the hypershade or in the perspective view) and select -> **Polygons - >Separate** from the menu at the top.

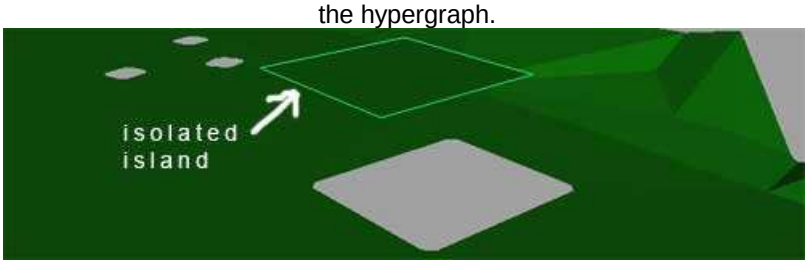


Now in the Hypergraph, everything that is not physically connected to the main mesh will show up as separate polysurface underneath the infantry mesh. Now we need to see this dirty unattached island to see if it is worth keeping or deleting.



To determine which polysurface is which, simply click on it. Whatever you click on in the hypergraph will be selected and highlighted in the main perspective viewport. In my case, **polysurface 5** is the main navmesh, while **polysurface4** is a little island. (This may be different for you because in my experience the navmesh generator gives a different result every time.)

So click on the the isolated island in the Hypergraph (polysurface4 for me). It may be too small for you to immediately see amongst the large infantry mesh. So now comes our handy little **'F' key** shortcut. Once the polysurface is selected in the hypershade, move your mouse pointer over the perspective view and press **'F'**. It should zoom straight to the island you selected in the hypergraph.

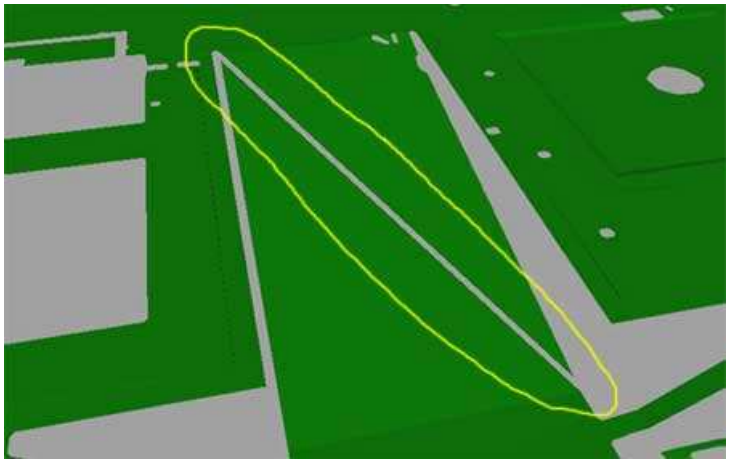


This little square is floating above the navmesh. We can easily determine that this is a completely unnecessary piece of mesh and we can delete it. So make sure it is still selected and simply press the **Delete** key.

Luckily I only had one island. If you have more then follow the the above process and repeat. If you feel you have an island that you need to keep, then keep reading and the following will show how to re-attach or weld pieces back together.

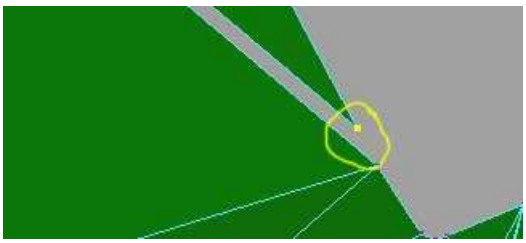
.

Theoretically, once I now export this it will work fine in the game.
Theory and practice however are two very different things. We need to now scan over the whole navmesh with our own eyes and see if there are any problems.

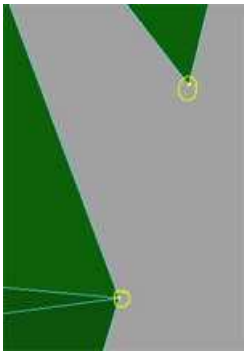


Ah, a rather large gap in the lower bridge area presents itself. This means the bots will not be able to get from the bottom to the top. We need to 'bridge' the gap, pardon the pun.

So right click over the mesh and switch to **'Vertex'** mode. Select the the vertex at the bottom on the right. (Either click it, or drag a box around it.)

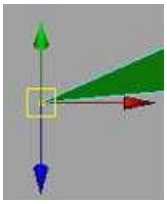


Now Press 'F' and that vertex will zoom in and become the pivot point for your camera.



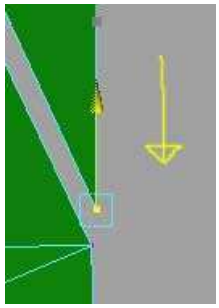
On the left screenshot: The Vertex at the top is the one we selected. The one at the bottom is where we are going to attach it to, thus sealing the gap at one side of the bridge.

What we do now is make sure the top Vertex is selected and press the **'W'** shortcut, which if you remember is the **Move tool**.
Once **W** is pressed, the move manipulators show up on that vertex as shown below ->



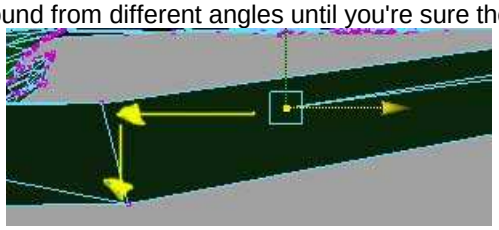
There are three directions in 3d space that you can move the vertex in. All you have to do is click on the corresponding arrow and drag it in that direction.

So navigate to where you can look down on the vertex from a top view to start with. (Remember the **Alt+left mouse** navigation tip?) Now click on the move manipulator arrow and drag it toward the bottom of the bridge

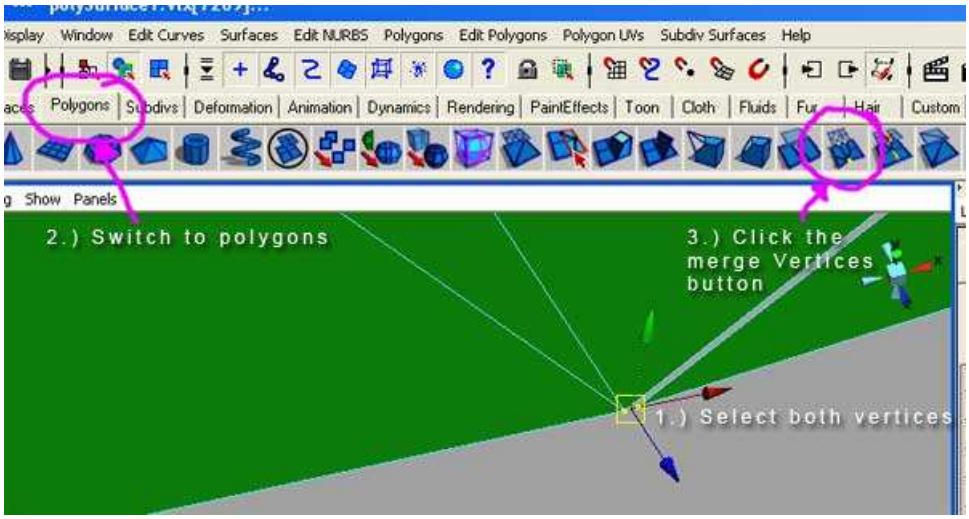


Now spin around to a side view and you'll see the vertex is too high. So drag it the rest of the way

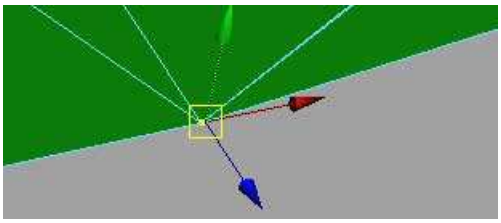
toward the bottom vertex and then drag it down so that the two vertice's are very close together. Rotate the view around from different angles until you're sure they are aligned well.



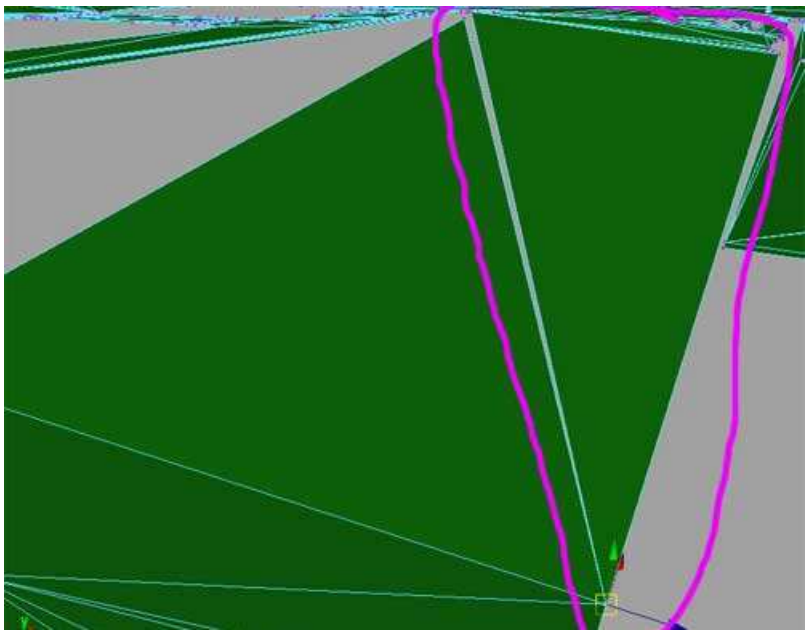
Once you have the 2 vertice's aligned very close together, we now need to do a **'Merge Vertices'** command.



So select **BOTH** vertices by dragging a small box around them. They will both become highlighted. Switch to **polygons** mode and then click the **Merge Vertices** Button. Keep in mind the vertices **MUST** be with a certain proximity to each other before they will merge. You can edit this proximity distance in the Merge Vertice options (*Edit Polygons -> Merge Vertices Options Box*) Or more simply just use the move manipulator to get them even closer together.

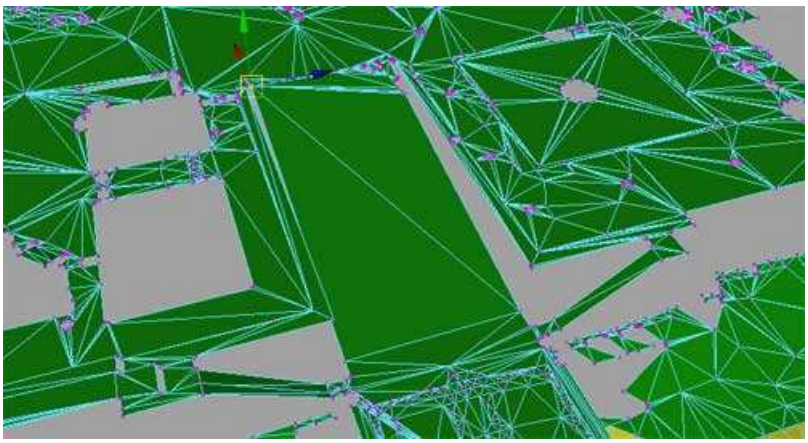


And there you have it! The two vertices become one and one side of the bridge is now re-attached to the other. You can use this same method to re-attach islands you want to keep.
We're not finished though! Only one side of the bridge is connected, so in its current state, we have just created a **manifold** piece of geometry which is not allowed at all.

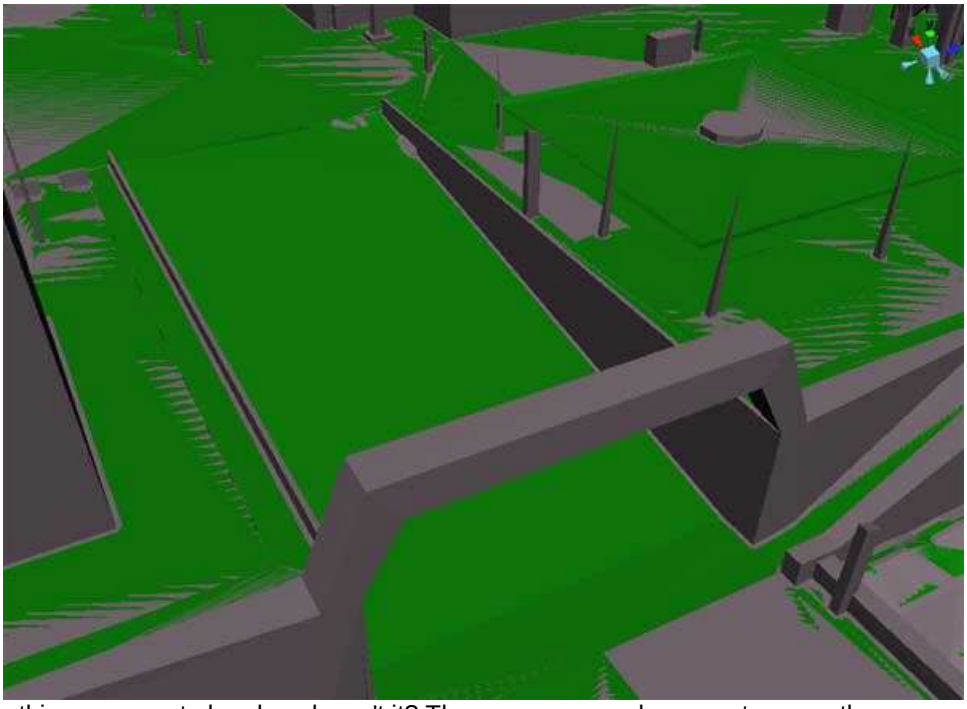


See the large triangle/face above. It is now connected at the base of the bridge but the top is still unattached, making this particular face manifold. A face can never be just attached by one corner like this. The game will reject it immediately.

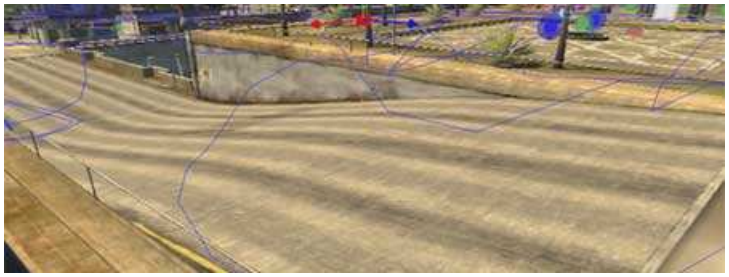
So now what? Yep you guessed it...we use the same merge vertices command at the top of the gap.
So repeat the steps outlined above until the two top vertices are aligned and merged.



Now the top gap is closed and the bridge is whole. Easy! Or is it?
Lets check it against the preopt to be sure. Go back to the Hypergraph Window, right click on the **preopt polysurface** and select **'show'** Now it might pay to note that the preopt is not perfect and should only ever be used as a guide. Lets see what the bridge looks like now.



Everything appears to be okay doesn't it? The green navmesh seems to cover the up-ramp off the bridge perfectly. However, I know for a fact that this ramp section in the game is not flat. The actual ramp model in-game curves off at the top. I only know this because I've navmeshed this level multiple times now and know that this particular section is a problem area. This is what makes this process so lengthy. You have to continually come back to Maya to keep editing until things are right. The only way to check is to export the mesh, go back to the Bf2editor and show the new navmesh from the render menu.



He's a screenshot from the editor. You can see how it gradually slopes down and is not just a straight surface from top to bottom. Sometimes we can actually import the model mesh into the scene as well to get a more accurate look at how the mesh sits with it. Unfortunately this mesh (div_citybrdgsloop is its name) has quite a simple collision mesh which is identical to the preopt's version of this ramp.

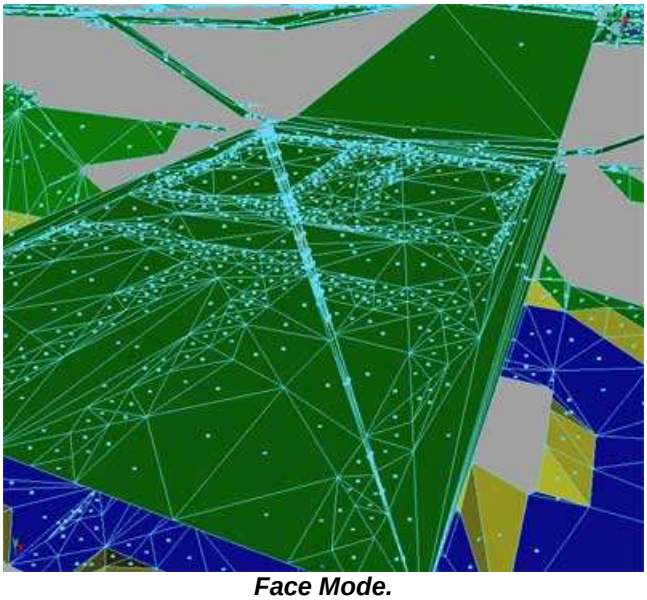
In future if you need to import a more complex model of something into the scene, look in -> **C:\Program Files\EA GAMES\Battlefield 2\NavMesh\work\Mashtuur_City\GTSData\debug\self_intersecting_objects**

In this directory you will find a lot of obj's that the navmesh generator spat out. If you're lucky, (unlike me this time) you may find a more complex model for you to check or build your navmesh around. So to overcome this problem, I had to just play with trying to make my navmesh follow the same curvature of the real bridge ramp by trial and error. I better show you some more general Maya techniques now to get you on your way.

Part 8 : More Maya editing techniques

1.) Deleting faces

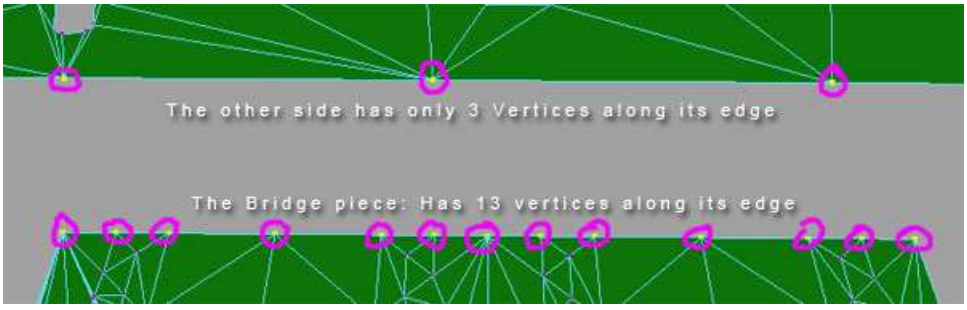
Often you will find more unnecessary sections in your mesh that you may just want to get rid of. All you have to do is right click on the mesh and select '**Face**' from the context menu.
Now you are in **Face edit mode**. See the picture below - Each dot in the middle of each face is the select point for that face. Select a face by clicking the dot, and simply press delete. Or drag a box around many faces and delete them all at once. The **F** zoom shortcut works on selected faces also.



Face Mode.

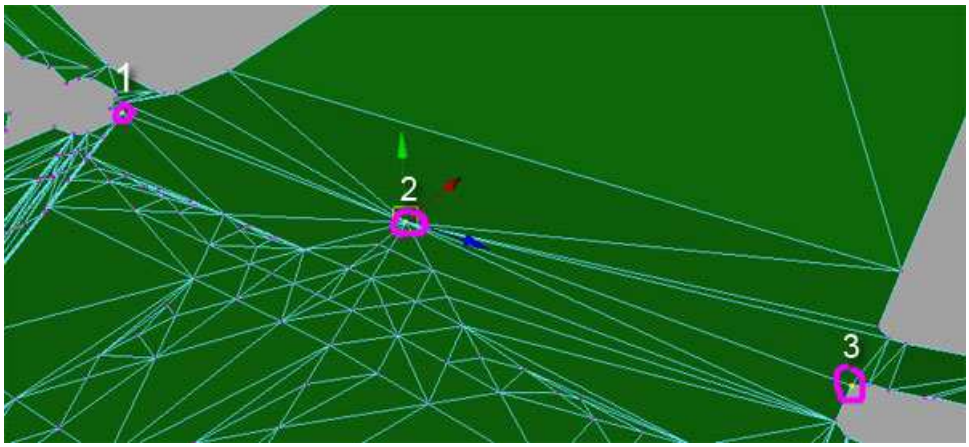
2.) Simplifying the Navmesh

Lets look at the same bridge section. The other side of the bridge also has a large gap. Switch to **Vertex mode,** and select all the vertices along the edge of the bridge and use the same '**Move**' (shortcut **'W'**) technique to align the verts. There is one problem with this part though :

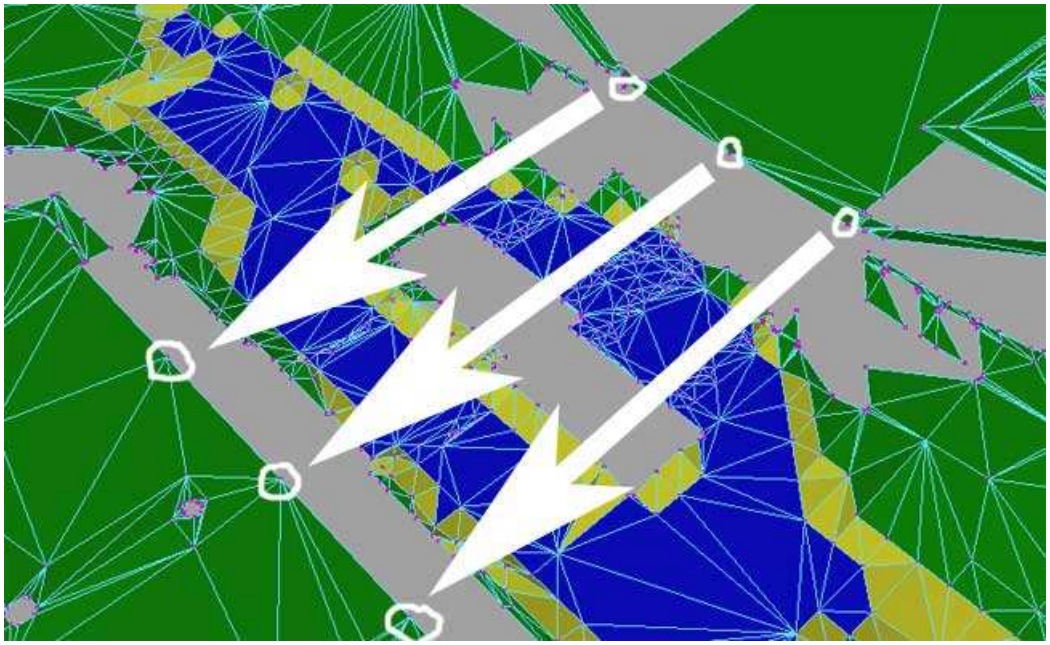


The 2 sides do not match up. How are you going to merge 13 vertices to just 3 on the other side? Technically you could merge 4 verts on one side to one on the other, but this is messy and you have the opportunity to create self-intersecting geometry this way which will render your mesh useless.

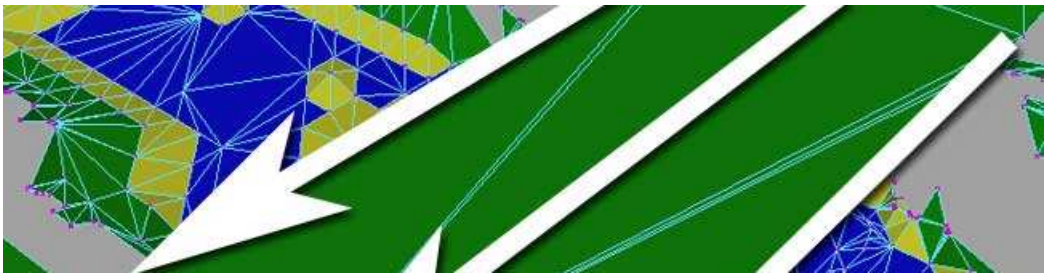
The answer is to simplify the bridge section so that it also only has 3 vertices along its edge. Check out the **'Face Mode**' screenshot 2 pictures above. That's the bridge section in question. For a simple flat piece of geometry it sure is overly complex isn't it? What's the quickest way to fix this? Lets delete it! Look to the other side of the bridge and find a part that is compatible. (i.e. also has
only 3 vertices along its edge)

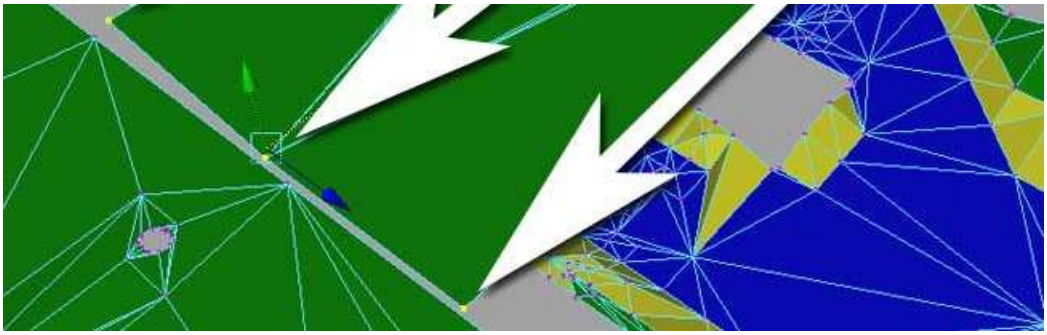


We have a winner. 3 vertices along an edge at the base of the ramp. Now switch to Face mode, select every face on this side of those 3 verts and delete them. Be careful not to select the water faces below the bridge at the same time or you will accidentally delete them as well. You can delete the faces section by section to be more safe. It may be easier to look up from underneath the bridge to delete all the faces.

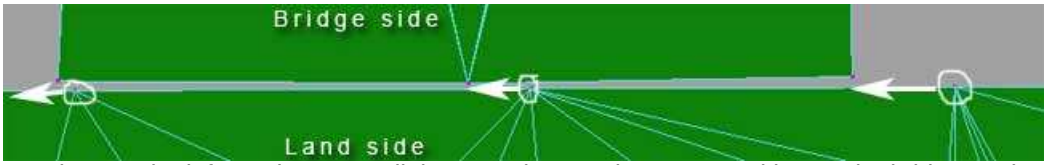


So now I've deleted that whole complex bridge mesh and am left with 3 vertices on each side! Now select the three verts at the bottom of the ramp (You can hold down 'shift' and select verts one
after the other if that's easier) then switch to the **Move tool 'W'** and create a new bridge by dragging it all the way across to the other side with the move manipulator.

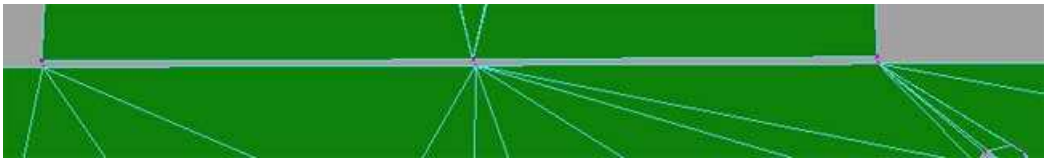




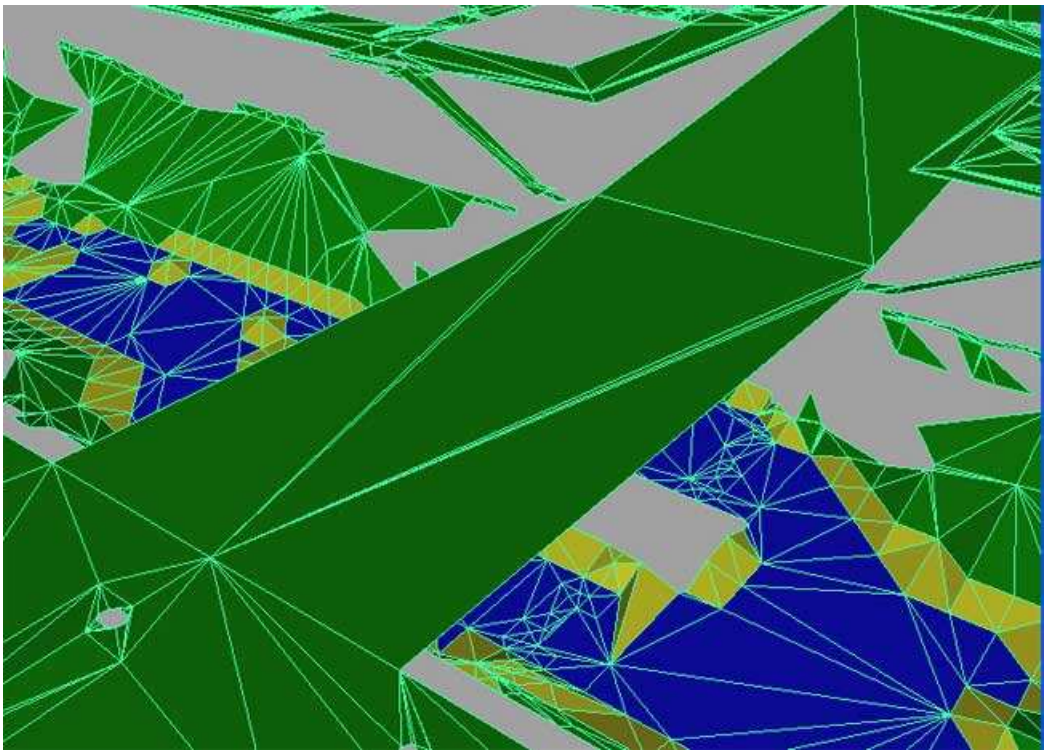
Hey Presto! A new simplified Bridge. Now lets line up the verts on the 'land' side to merge them all together. We don't want the bridge to become wider, so we need to move the verts on the 'land' side to accommodate the position of the bridge vertices.



Move them to the left one by one until they match up to the same position as the bridge vertices.



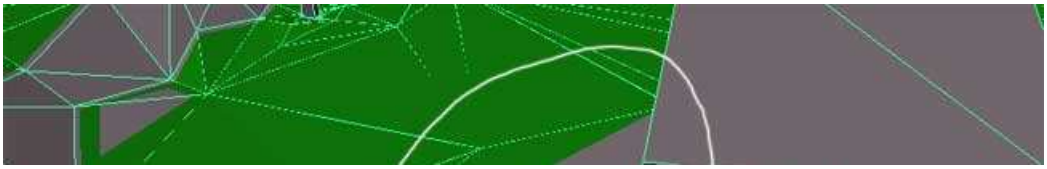
It should now look like the picture above. drag a box around the 2 vertices at each point and do a **'merge vertices'** command on each pair.

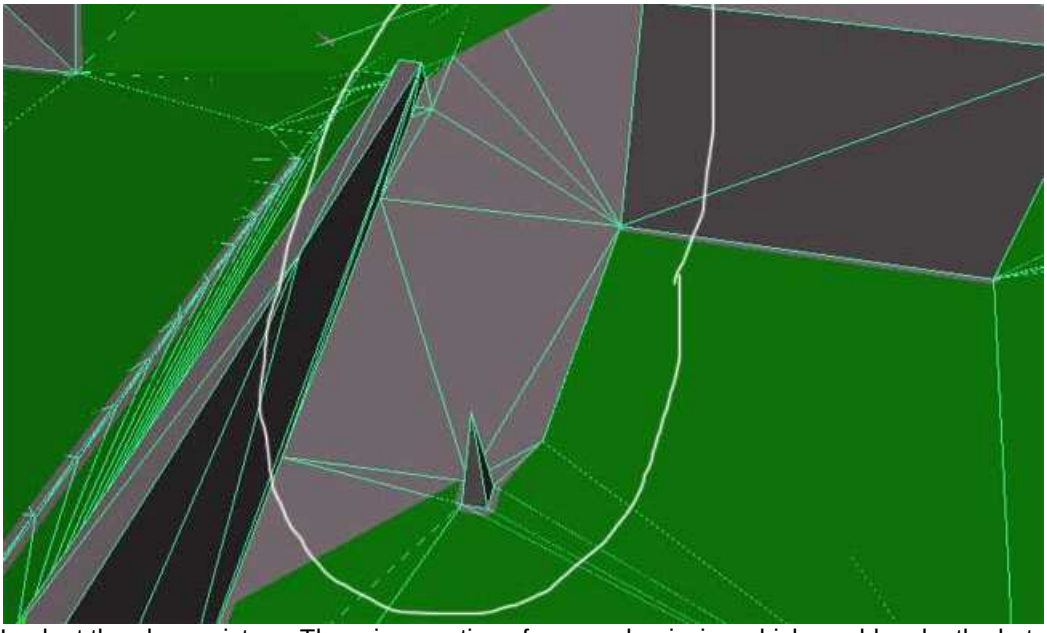


Finished! A new simple bridge all connected with no gaps. Now bots can run from one side to the other with reckless shooty abandon.

3.) Creating new navmesh parts (edge Extrude)

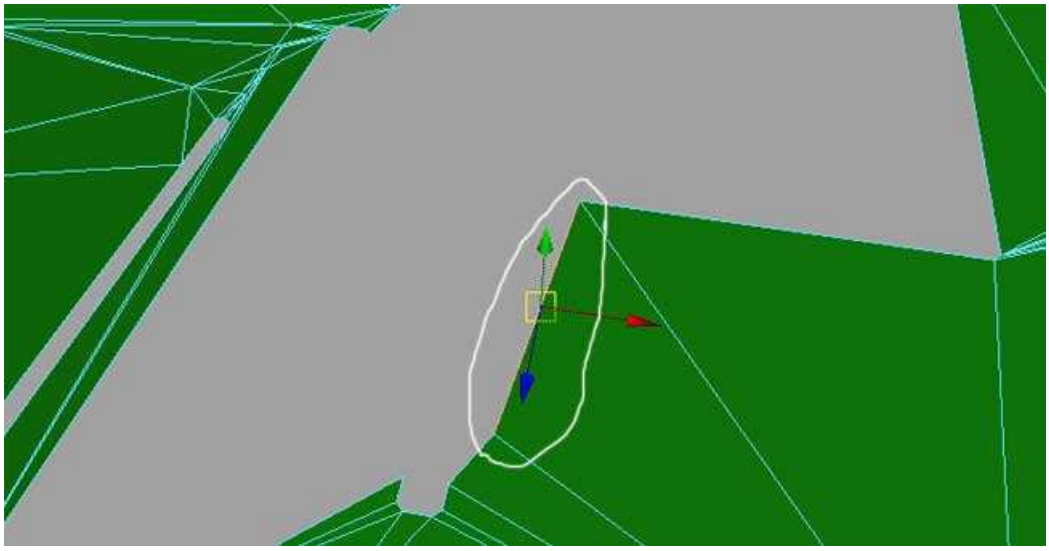
What about sections where there is no navmesh at all? Sometimes the navmesh process fails to produce the navmesh in places you need it to be. In my experience this happens way more with the vehicle.obj than the infantry.obj. Let me show you a simple edge extrude technique to overcome this.



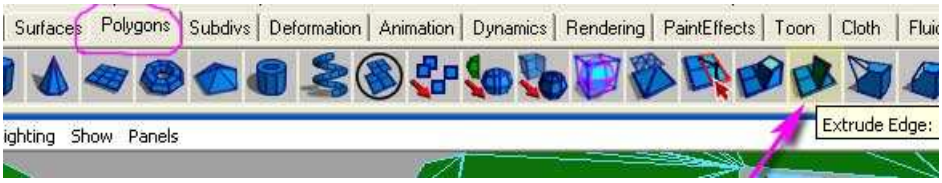


Look at the above picture. There is a section of navmesh missing which would make the bots run all the way around this building to go up the ramp on the left. Lets make some new navmesh so there is a pathway directly there.

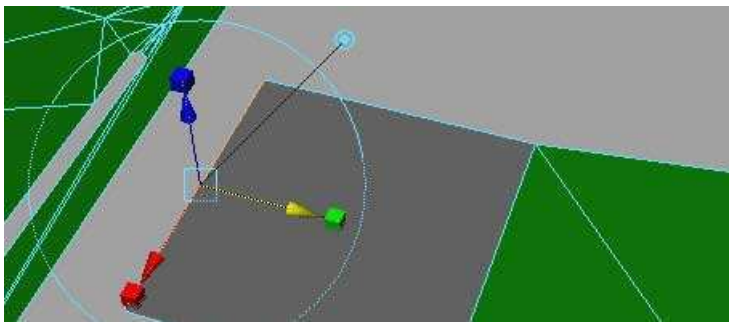
We are now going to go into **'Edge**' Mode. Right click on the Navmesh and switch to **'Edge**' Mode.
Hide the preopt again in the Hypergraph window so we can see what we are doing better.



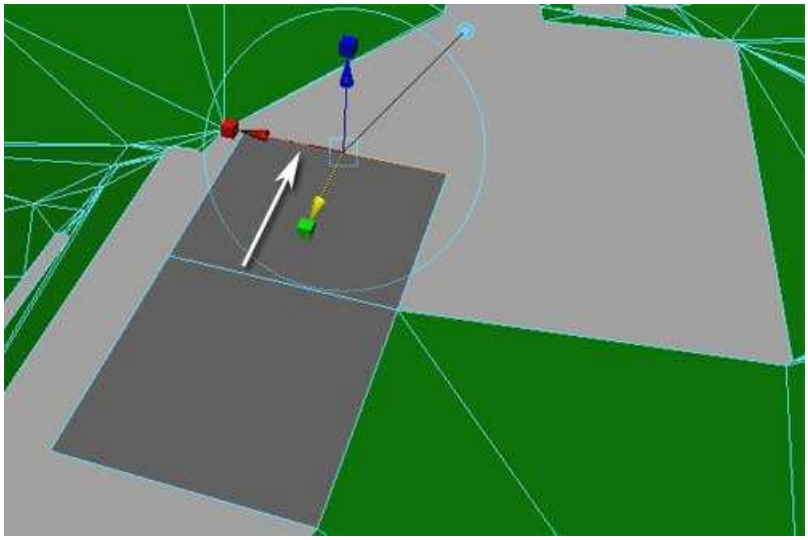
Now that we are in edge mode, we just have to click on the edge we want to extrude.



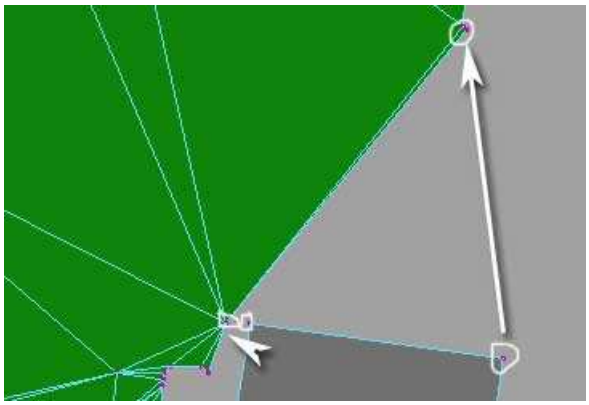
As soon as the edge is selected (it turns yellow when selected) make sure you're in the polygons section(above) and click the **Edge extrude** button **ONCE!** Be very careful to only click it once.



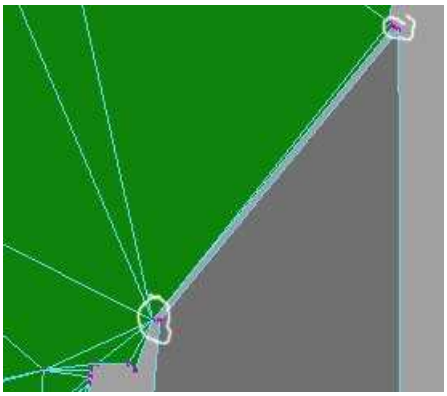
The edge extrude manipulator pops up, so click on the Yellow arrow and drag it across like you would with the move tool. Note we just made a 'square' This is bad news but we'll fix it in a second. Okay now select the upper most edge to drag it the rest of the way to where we want to join it to the other side. Select the edge and then click the **Edge Extrude** button once more.



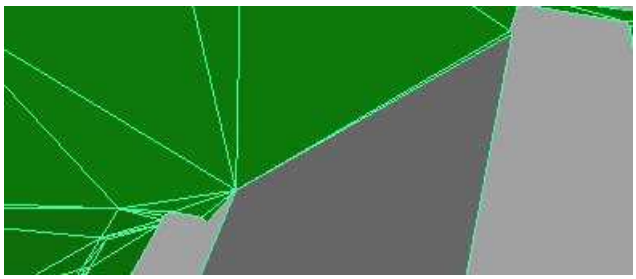
Almost there. Now right click on the navmesh and switch to **vertex Mode.**

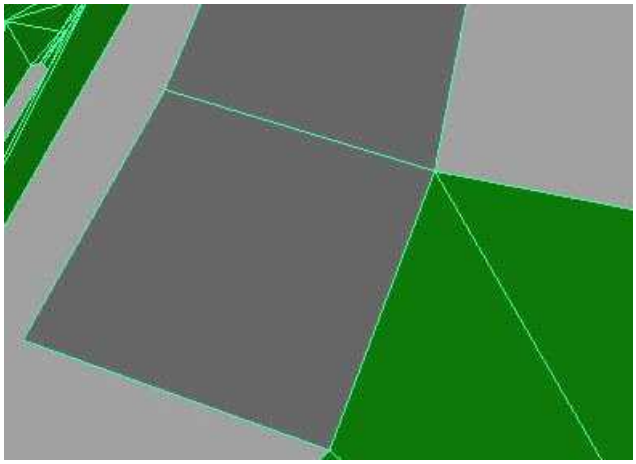


Move the two verts separately to the closest points on the navmesh using the move tool **(W)**



Select each pair of verts and click the **'Merge Vertices'** Button.

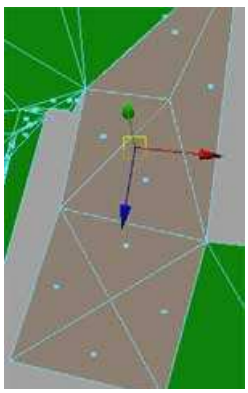




There we go, the mesh is joined and the bots have a path. But there is no material on our new part and they are still squares. Lets change them to triangles or else this mesh will not work.

Right click on the mesh, and switch to **Face** mode. Select the 2 new 'four sided' faces.

Now go to the '**Edit Polygons**' Menu and select **subdivide**. (Click on the subdivide options box to make sure the mode is 'triangles' if the faces don't become triangles)

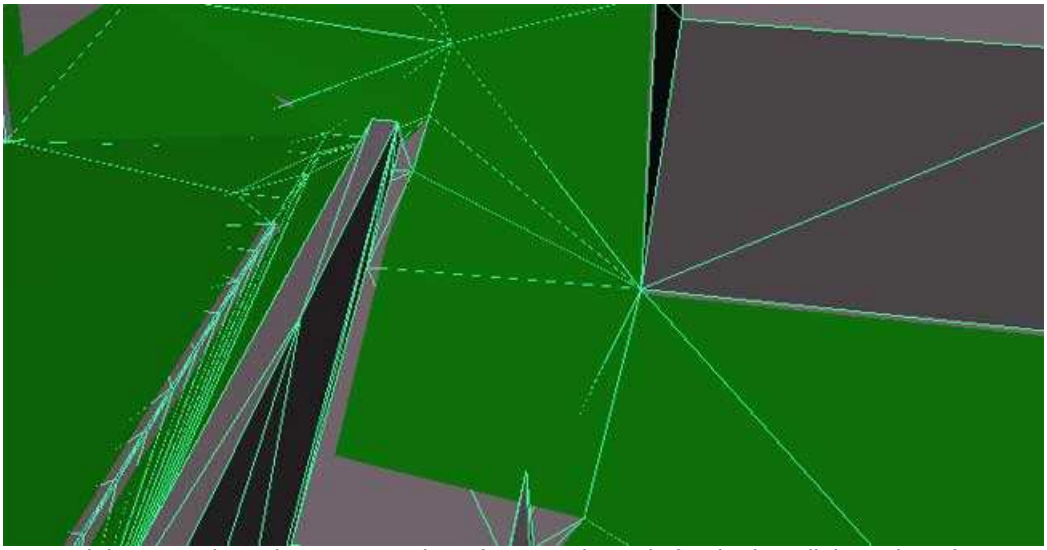


Now the square faces are subdivided down into triangles which makes our new mesh comparable with the navmesh rules.

Lastly lets give our new section the same material as the rest of the navmesh which is, if you remember, called '**Ground2**' This is really easy too.

Just right click on the new section while it is still selected. (select it if it isn't already) and select **Materials -> Assign Existing Material -> Ground2**

Now your new part of navmesh is the same material as the rest. And that's all there is to it!

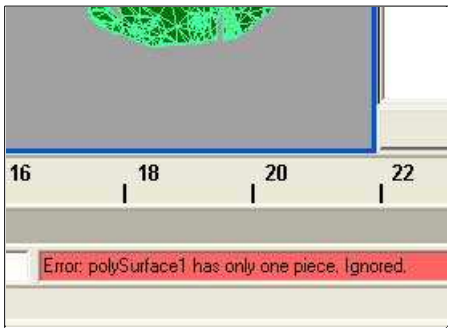


And there you have it, a new section of navmesh ready for the bots little curious feet.

Well, that's about all you need to know about Maya editing. What you need to do now is Export this funky new fixed mesh (after checking to see if there are any more gaps anywhere you need to fix.)

So go back to the **hypergraph window**, select the **preopt polysurface and delete it**. Don't forget to do this or you will export the preopt at the same time as the navmesh which will cause issues. So now the only thing in our scene is the infantry.obj navmesh. Lets do one last little quick test to make sure we still have one island.

Go to **Polygons -> Separate**

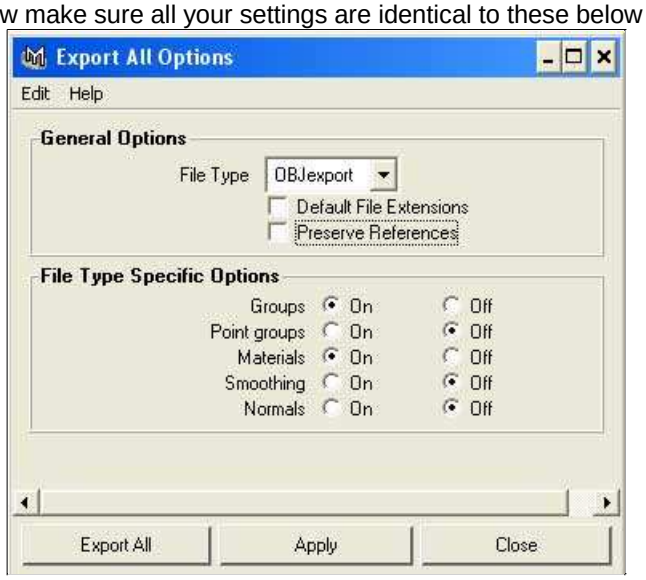


If you only have one island then this little error message pops up at the bottom of the screen, 'Error: polysurface1 has only one piece' which is good news! If you still have floating islands, they will appear in the hypergraph window. Select them and delete them or attach them as you see fit. Now onto the exporting.

Part 9 : Exporting the Navmesh

Go to **File -> Export All Options Box**

Now make sure all your settings are identical to these below ->



Just make sure that File Type is **OBJexport** and only '**Groups**' and '**Materials**' is selected.
Click on the '**Export All**' button. Now export it back to its original location ->

C:\Program Files\EA GAMES\Battlefield 2\NavMesh\work\Mashtuur_City\GTSData\output

Overwrite the original **infantry.obj.**

There! An edited Infantry navmesh ready for the game. Guess what you have to do now...
That's right - start a new Maya scene and **import** the **vehicle.obj**

AND REPEAT THE SAME PROCESS.

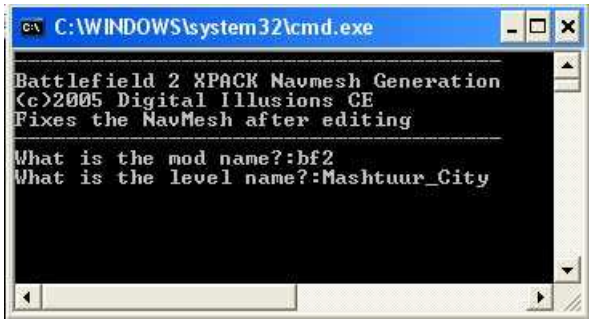
1.) Check for islands and delete or attach them
2.) Check for unmeshed areas that you could add navmesh to.
3.) Export it to the above location except to obviously overwrite the vehicle.obj instead.

Once you are sure you have a vehicle and infantry navmesh that only contain one island and cover all the parts of the level you want them to, it's time to get it in the game and test it.

Part 10 : Fixnavmesh.bat

Go to -> **C:\Program Files\EA GAMES\Battlefield 2\NavMesh**

Double click the **Fixnavmesh.bat**



Once again put '**bf2**' in mod name and '**Mashtuur_city**' in level name.
Press enter and let it finish. It will open the editor during this time and convert your edited obj files into quadtree files (*.qtr) and also create a Height Map from the Obj Info. The heightmap tells the bots how high the buildings and terrain go so that they don't

crash into them while flying.

Once it is complete the editor will automatically close.

Part 11 : Get it into the game

The new quadtree files (The final navmesh files) and now located in ->

Bf2\levels\Mashtuur_City\AiPathFinding

Open your vanilla playtesting Mashtuur server.zip and copy this **AiPathfinding** folder into it.

Also make sure your **gameplayobjects.con** and the **Ai folder** from the editor are in the server.zip under-> **gamemodes\SP1\16**. Mashtuur does not have a SP1 folder initially as Dice never released
this as a single player map. Create the necessary **SP1\16** folders under 'gamemodes' in the server.zip if they are not there.

Last of all, we need to get the map to show up in the single player menu inside the game.
Go to **bf2\levels\Mashtuur_City\Info** and open **Mashtuur_City.desc** with notepad.

In this section ->
<mode type="gpm_cq">
*<maptype **ai="1"** players="16" type="doubleassault"*
locid="GAMEMODE_DESCRIPTION_doubleassault">Unlocalized test for this mode on this map</maptype>

Add in the line **"ai=1"** as shown above in the 16 players section.

That's it. Your server.zip is ready with the new navmesh files in it and your new gameplayobjects.con and Ai strategic areas included. Now go and playtest it. Try and keep an eye on what you think might be problem areas. See how the bots react and if they get stuck etc. Make note of these areas and
look at them in the editor later with the navmesh enabled in the render menu. Perhaps the navmesh
needs more tweaking. Go back to maya if need be and fix it.

I hope that gives you enough information to get it happenning. Enjoy creating single player bot navmeshes :)

Myself and other contributors to this navmesh process can usually be found over at the BFSP forums.

By: Kysterama Feb 26, 2006.
Allied Intent Community War Mod
Battlefield Single Player Forums