

## Portal Documentation

By: Nialls Chavez

The purpose of this documentation is to show how the connections between portals work and to give a break down of what a connections.con file and portals.p file does. In basic terms, when the Map creation GUI is run, and a map creator has a portal put on the GUI, what happens is that the location and the ID of that Portal is placed in a HashMap, then that HashMap is placed in an enclosing HashMap which then sets the key to the name of the Map created , So , now we have a HashMap With the key being the name of the file we just created, and the value being the HashMap containing the Portal ID delimited by the location of that portal.. The Java syntax for this HashMap is:

```
HashMap<String, HashMap<String, Point>
```

This object is then saved inside the Portals.java file which created an object output file which saves the state of that HashMap and contains a getter and setter method, which allows the HashMap to be modified and allows for the HashMap to be modified and added to include more portals. Because the Portal Object needs to hold all the data for the World state.

While this is awesome for holding portals, we don't have any way of connecting these portals together to get from point A to point B. That is where the PortalConnections.java comes in. this Gui allows for people to set custom portals between objects, this lets there be essentially as many connections as there are portals.

The way that this file works is by reading in the portals.p file that was already created upon save of a new Map. Once this is done, It reads in all the points and allows the user a quick and convenient way to connect the portals. once the user is satisfied with their selections, they save the file which inturn creates a connections.con file. This file contains another HashMap which contains the selected portal(A HashMap with the key being the file name, and the value being the Point on the map) with the value being the portal you want to connect to(a HashMap with the key being the name of the file it lives in, with the value being the location of the portal it is connected to. This Looks like :

```
HashMap<HashMap<String, Point> , HashMap<String,Point> ()
```

This is then saved into a connections.con object which contains a connections object which again has a getter and setter method in it for getting the current connections, and setting in a new one.

This pretty much sums up how it works. Yeah.