Terrain High-Level Design

**Updated Terrain Requirements:**

* The game shall have at least two distinct environments (ex: desert, tundra…)
* Levels shall be generated pseudo-randomly
* Worlds shall be infinite in size

In order to complete these requirements we have decided to make the environment out of nine planes (as can be seen in Fig1). The planes will be arranged in three columns and three rows of three. Each plane will be equal in size. If the main character crosses the border between two planes, the planes will update in order to maintain the square shape. These planes have x and y positions. If a plane moves in the x direction to the nearest plane. The three planes with the biggest x difference than the player will update and move over twice the distance. If the player moves in the negative direction, the same thing will happen in the negative direction. The same principles hold for they direction. This will create an infinite world and an example can be seen in fig 1.

**Pseudocode:**

If distance from the player >=75 in the x

X +=150

If distance from the player < -75 in the x

X -=150

If distance from the player >=75 in the y

X +=150

If distance from the player < -75 in the y

X -=150

-75 0 75

o

= origin

= Player

o

-75 0 75

o

Fig 1

We have also chosen to make the environments different based upon the distance that a player is from the origin point. Every plane within the 100 unit circle will be of a different environment material grouping. The next circle will have a 200 unit circumference and then 300 and finally 400. Fig 2 is a diagram of how this would look. Each circle will have access the same array of materials. A random material will be applied to each plane that enters that circle. As the plane enters the next circle, a different random material for that area will be applied.

**Pseudocode:**

If distance > 0 and <100

Material = random m[0,3]

If distance >=100 and <200

Material = random m [4,7]

If distance >=200 and < 300

Material = random m [8,11]

If distance >300

Material = random m [12,15]

o

1 2 3 4

0 0 0 0

0 0 0 0

Fig 2