Combining Shapes

As you may have suspected, we could have possibly included another sphere into our first scene and had two spheres. Now imagine that we had one sphere at <0 2 2> and another at <1 2 2>. What do you think would happen? Would they overlap or would povray give an error message? Well the answer is they would overlap, Povray will never butt in because of the way you place your objects. We could possibly use three or four spheres to create something like a snow man. We might do it by placing spheres closely on top of each other, and changing the sizes of each sphere. The size of the sphere is defined in its declaration:

```
object {
 sphere { <0 2 2> 1 }
}
```

The 1 in { <0 2 2> 1 } is the size of the sphere, in this scene the sphere is 1 unit large, you may size it up or down by changing the number to be larger, or smaller. For example: { <0 2 2> .4 } would make the sphere smaller because I used a decimal less then 1, whereas { < 0 2 2> 1.5 } would make the number larger, because it is greater than 1.

We have covered the sphere, now we will talk about the box. The syntax for the box is:

```
box { <corner1> <corner2> }
```

Corner1 is the x,y,z location of one of the box's corners while corner2 is the x,y,z location of the other corners location. Each of the x,y,z values of corner1 must be smaller than the coresponding x,y,z location for corner2. The following is an example of box:

```
object {
 box { <-1 0 -1> <1 2 1> }
 texture { color red 1 phong 1 }
}
```

This box's bottom left back corner is located 1 unit left, and 1 unit behind the origin, while the oposite corner (The top right front corner) is located 1 unit to the right, 2 units above and 1 unit infront of the origin. This makes the box 2 units wide, 2 units high, and 2 units long, a perfect cube. Finally the color of the square is solid red with a phong size of 1. Note that on a box the phong is not as brilliant as on a sphere, the fact is spheres are nice and rounded, while box's are flat edged, this causes less highlight.