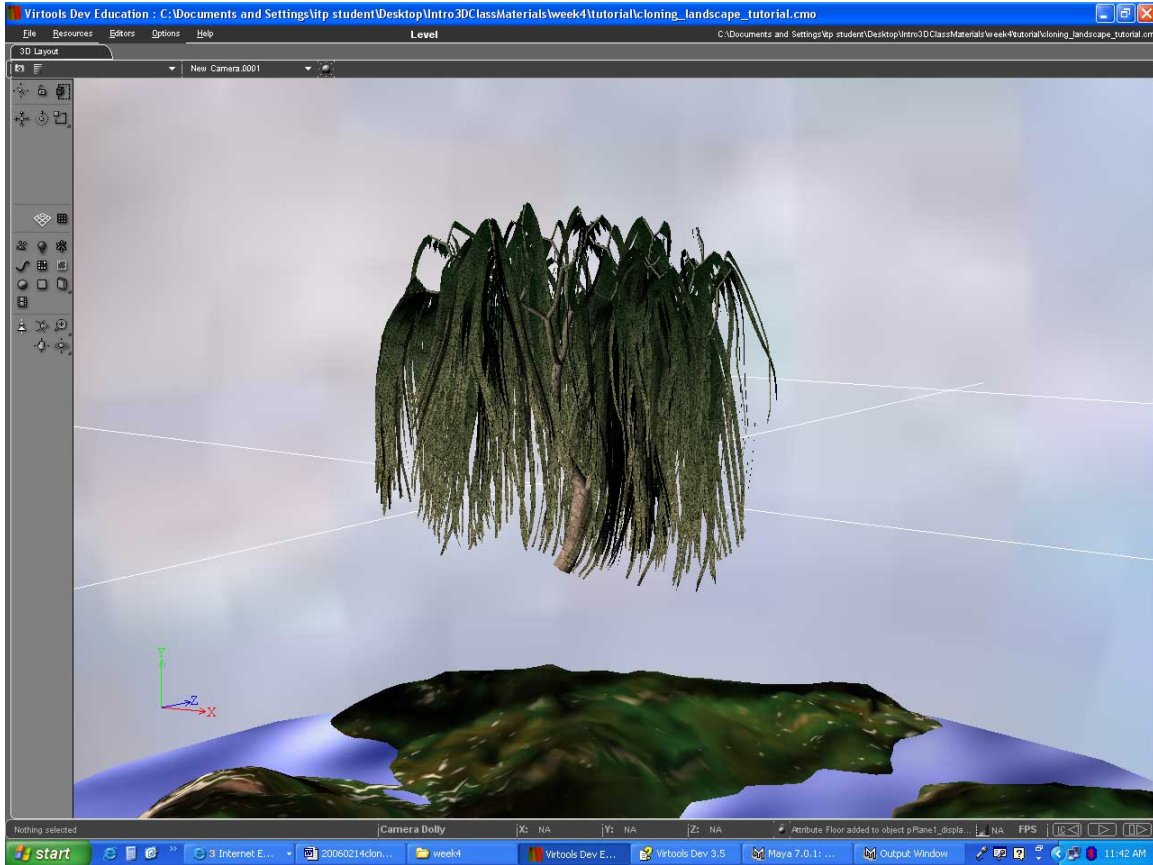


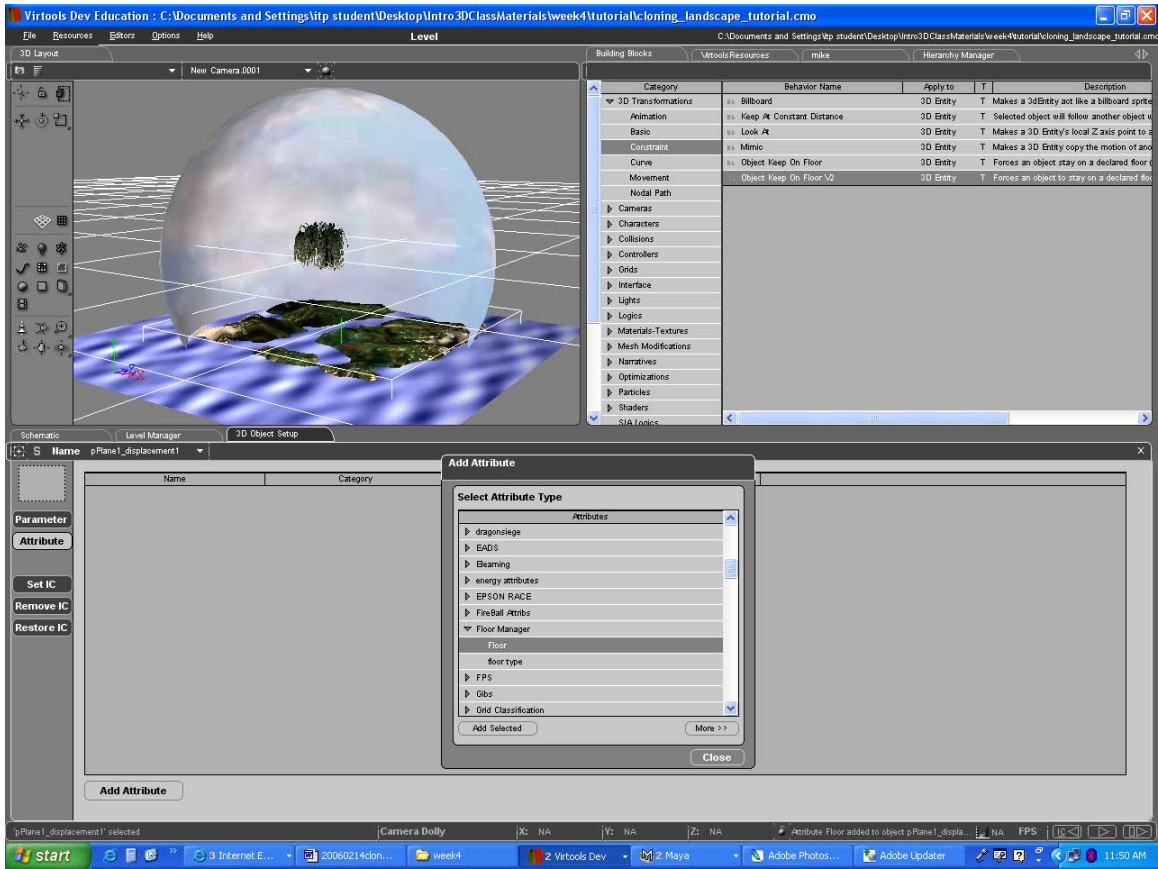
# Attack of The Clones!

Cloning is something you often want to do in Virtools to effect a distribution of similar objects around your scene without massively increasing the size of your Virtools file. A condign example would be strewing your scene with lush vegetation without having to have a large number of polygon intensive models.

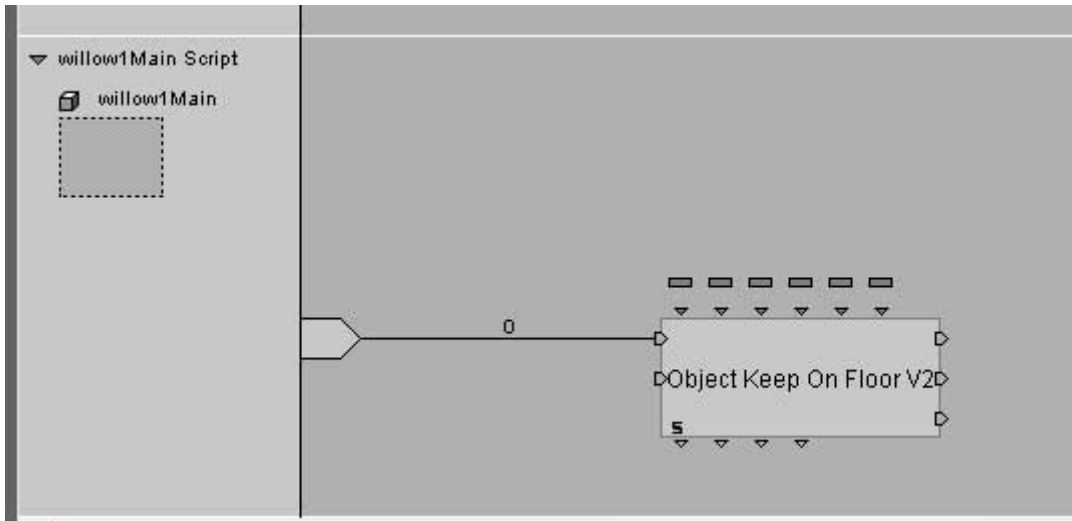
So let's start with a landscape containing a lovely plant. In this case, we just imported a willow into our water landscape.



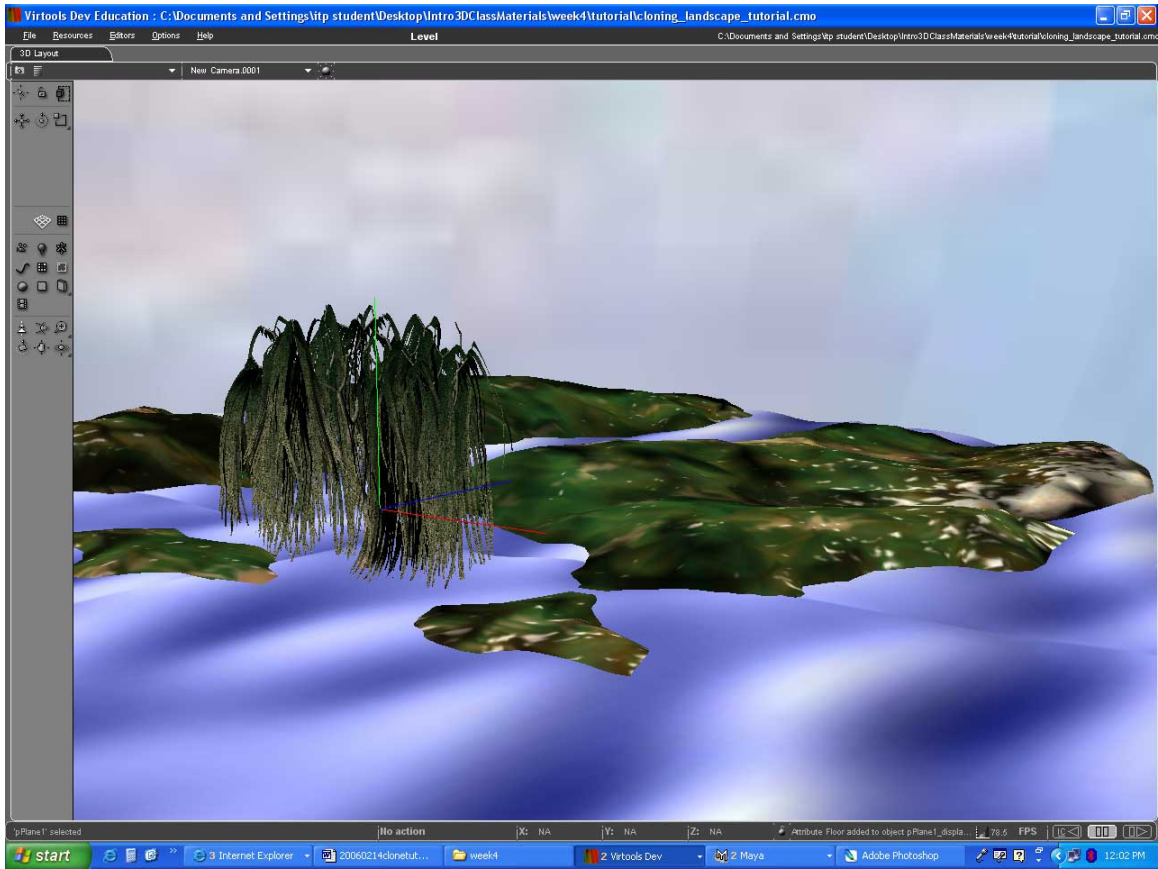
Now our willow comes in floating in the air. We could manually position it to sit on the landscape, or perhaps we could take a shortcut by constraining its position to our ground plane by 1) Adding a floor attribute to our ground 2) Adding a script to our willow including the 3D Transformations -> Constraint -> "Keep On Floor" building block. Let's try that. First bring up the object setup for your ground plane. At the far left of the interface select attribute rather than parameter. Then at the bottom of the setup, click "Add Attribute." Browse through the resulting pop-up until you find Floor Manager -> Floor. Hit "Add Selected." You should the "floor" attribute added to your ground plane.



Now add a script to your willow, and drag the Keep on Floor building block onto it.

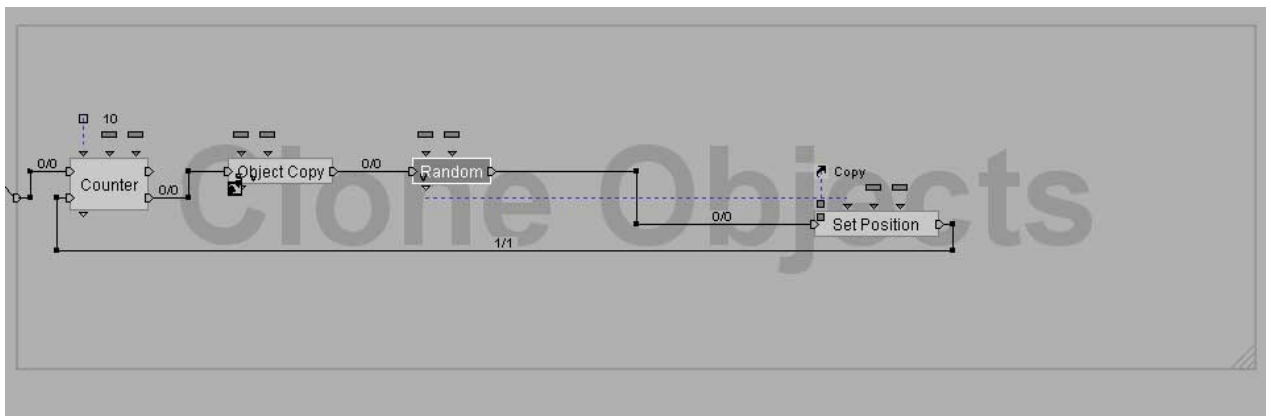


This is an internally looped building block so it will stay running all the time. You may want to set IC on your willow in case something weird happens, but then see the effects of this:



So you may want to set the initial position so that it doesn't fall into the water, but otherwise it seems to be working. Now let's look at cloning.

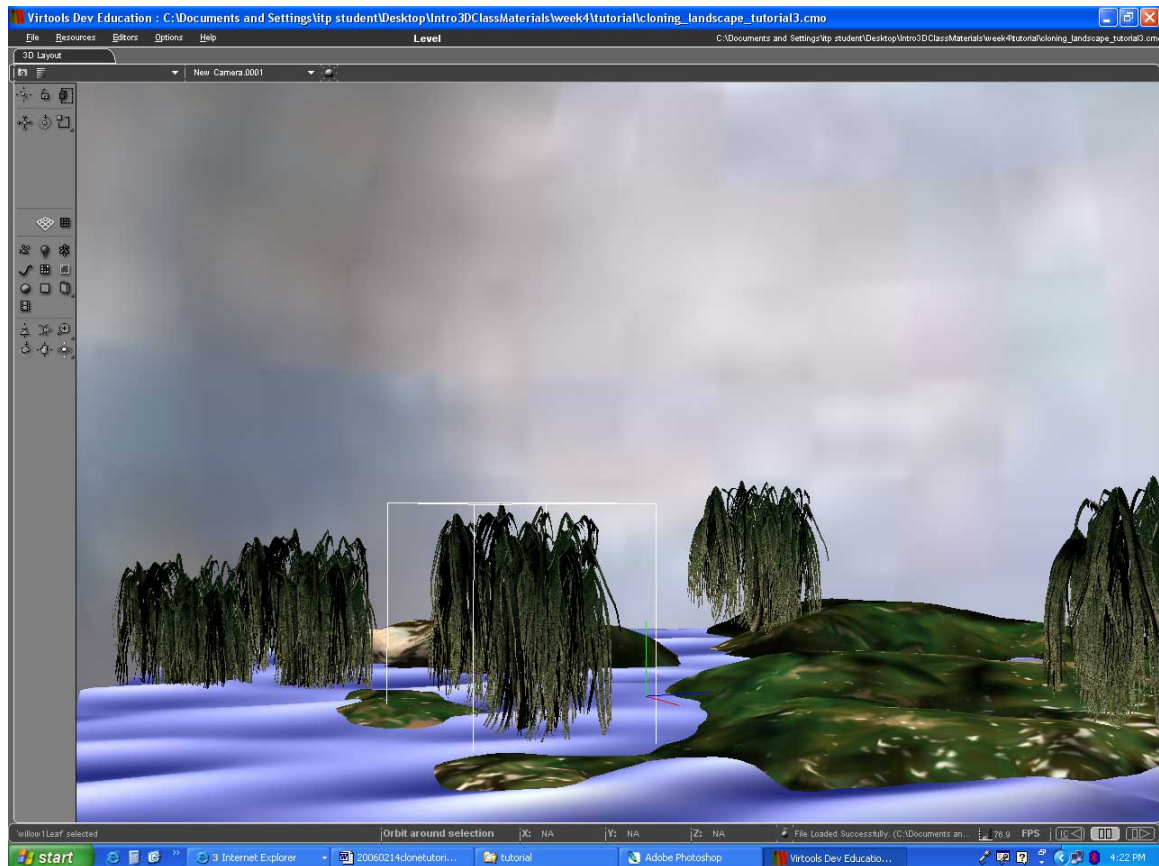
It's frequently useful to be able to encapsulate series of building blocks from other .cmo's in behavior graphs (see class 4 class notes). If you then right-click on a behavior graph and select "Save As" you can save the script as a .nms file (Virtools Script File). If you right click in the Level script of our scene here you'll see that you can "import behavior graph." Try importing the script "clone.nms" from 3d\Olson\Week4\Tutorial . Double click on it to reveal the following BB's:



So basically the way this works is that we:

- 1) Start a counter loop to indicate the number of copies to create. Try 5 to start.
- 2) Copy a specific object. Find your tree object.
- 3) Create a random position vector within your sphere. Min: (-4,1, 4), Max (-4,1,4) works more or less for me.
- 4) Finally set this random vector to be the position of the copied object. I used the ground plane as my referential, which can make things easier, but is not necessary.

You should see something like this:



If random placement will not suit your purposes, there are a number of ways to predefine the locations of cloned objects, including loading specific locations into an array, and then parsing that array instead of using Random to set the values. You could also pre-place a series of frames and iterate through the group using "Get Position" to provide the "Set Position" coordinates for the copied objects.

Note: do not apply copying scripts to the objects to be copied. This will cause recursive loops that can crash your program. IE when the copied object is created, it will begin trying to copy itself 5 times.

Go forth and Multiply!