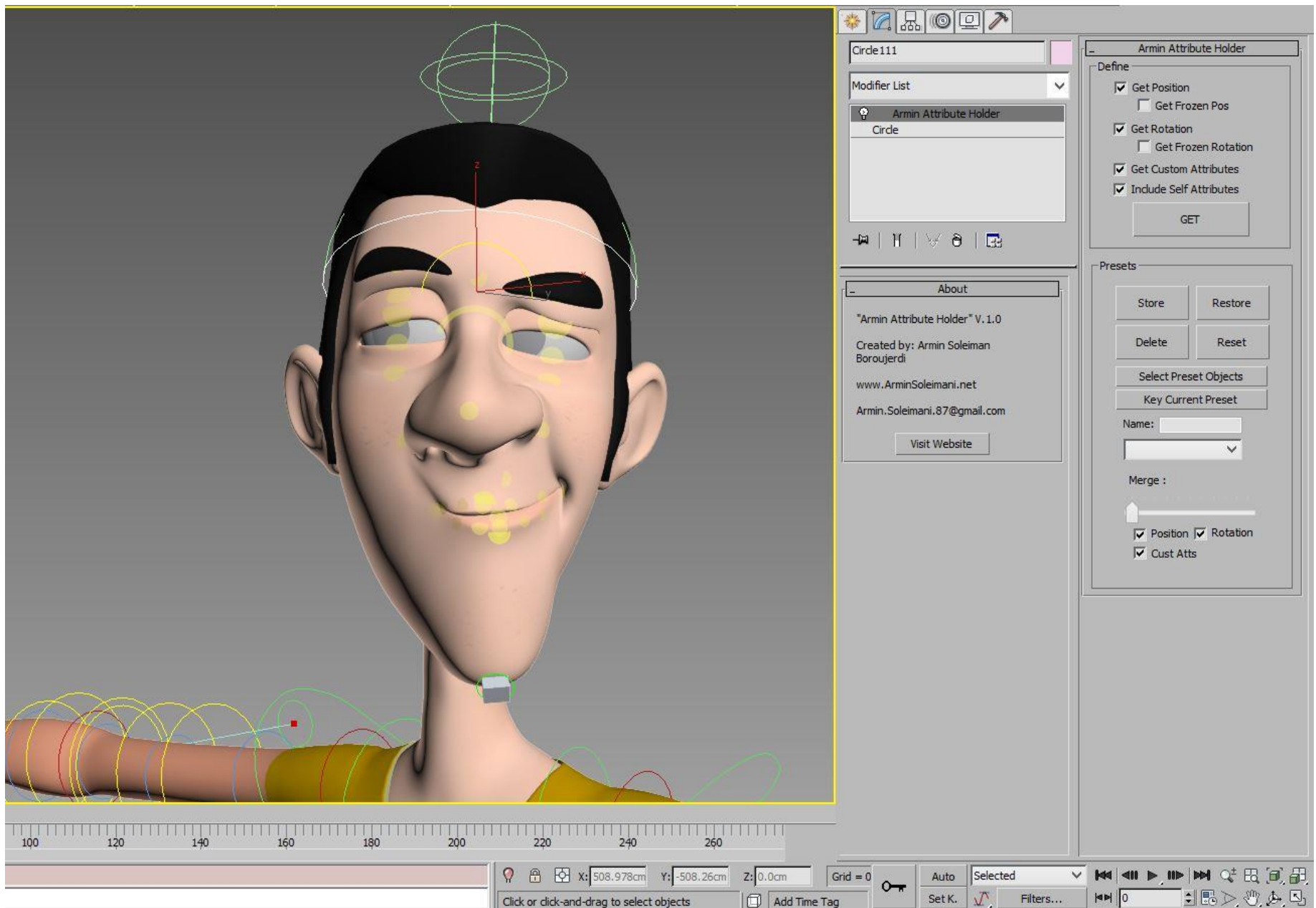


Armin Attribute Holder

V.1

for 3Ds Max



About Plugin

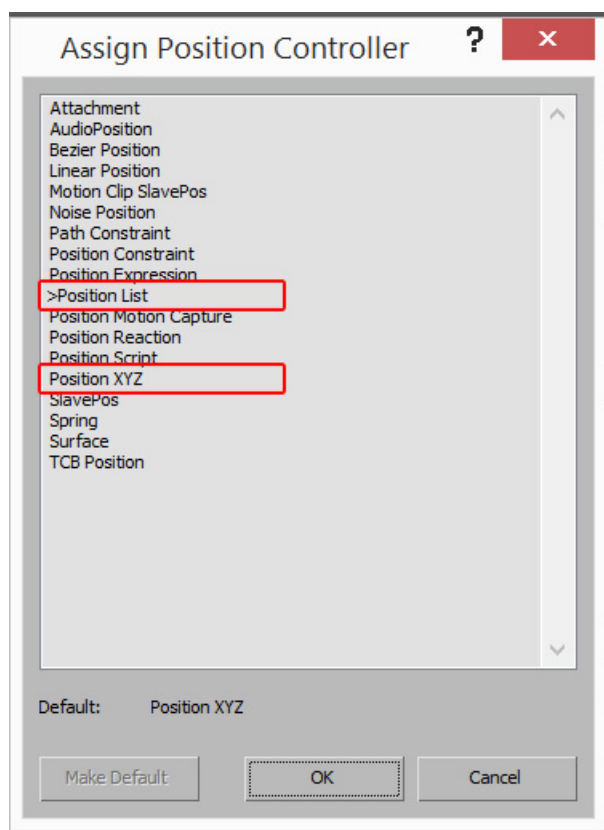
What's the difference?

"Armin Attribute Holder" is Automatic. Just select a bunch of objects with valid position and/or rotation controllers and/or custom attributes, and it will pick them up. So you don't have to Wire or copy/paste your parameters or controllers - one-by-one to the plugin unlike the famous "Pen_attribute_Holder".

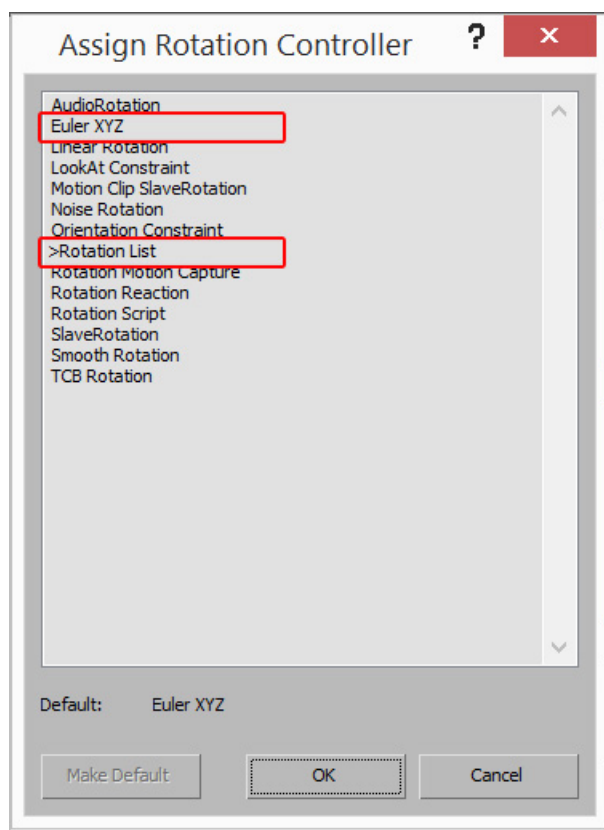
"Armin Attribute Holder" is a custom scripted modifier plugin, designed to store values of different types of controllers, such as Position, Rotation and Custom attributes. It can have as many objects as you like, stored in as many presets as you want. Originally it was designed to store different states of a controller (Position, Rotation, Custom Attributes), but after some developments, Now it actually could be used as a fully functional Pose Storing tool which could be used on any collection of objects with Position or Rotation and its user defined attributes. This plugin is also name-independent. So you may change the names of the objects assigned to the plugin.

Supported Controllers

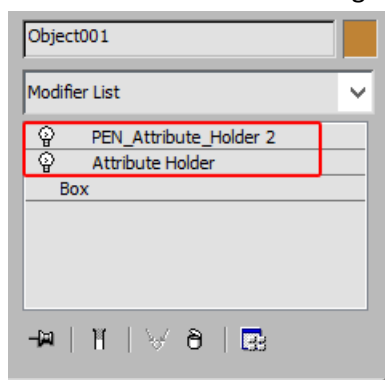
- Any object with "Position XYZ" or "Position List" controller.

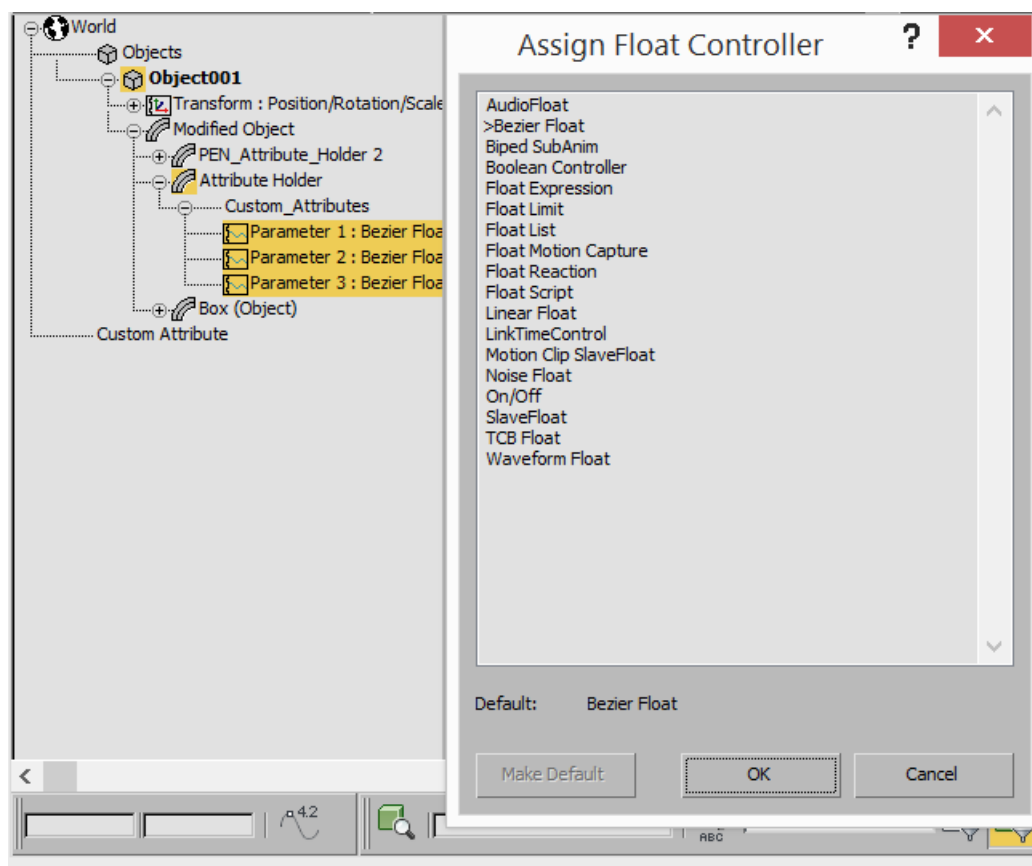


- Any object with "Euler XYZ" or "Rotation list" controller.



- Any object with "Attribute holder" and/or "PEN_Attribute_Holder_2" Modifier with *Bezier float* custom attributes. There are no limits in having the number of modifiers on a single object, or the number of its custom defined attributes and the number of its parameters.

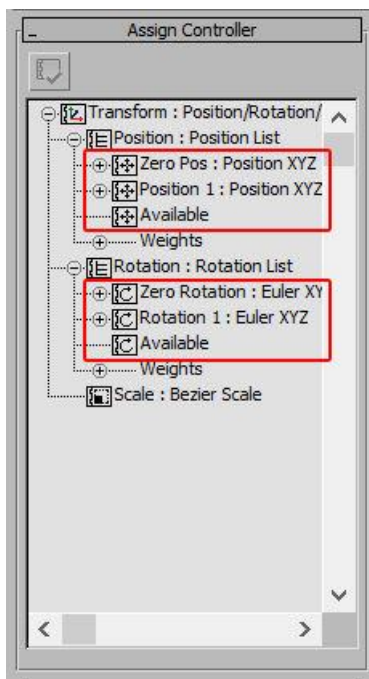




Limitations

It currently works with limited types of controllers – as named above. On list controllers such as *Position list*, it only recognizes the first children of the root *Position list* controller and will interact with its each sub-tree individually regardless to the active list state. Example:

Object001 > Position> Position List > Position XYZ (supported)



Any further children under the controllers marked above – like having a *position list* under the main *position list* - will be treated as a single value, just like Position XYZ, which means it will store its value but only restore it into the active controller in the Sub-position or sub-Rotation list. The same is valid for float controllers in the custom attributes.

DISCLAIMER:

This plugin is a beta version and is NOT fully tested. USE AT YOUR OWN RISK. Any damage caused by malfunctioning of this plugin is not responsibility of the author. However, any suggestions or bug report is welcomed. It is also free to use or share and selling it is not by the author's permission.

Understanding how it works

The way "Armin Attribute Holder" works, is that it looks for the supported controllers/modifiers in a set of objects that is currently selected, and keep an instance of those controllers to store and restore their values further on. The types of controllers to look for is customizable, and after saving different states with different controllers, it is also customizable to level of which type of controllers should be restored.

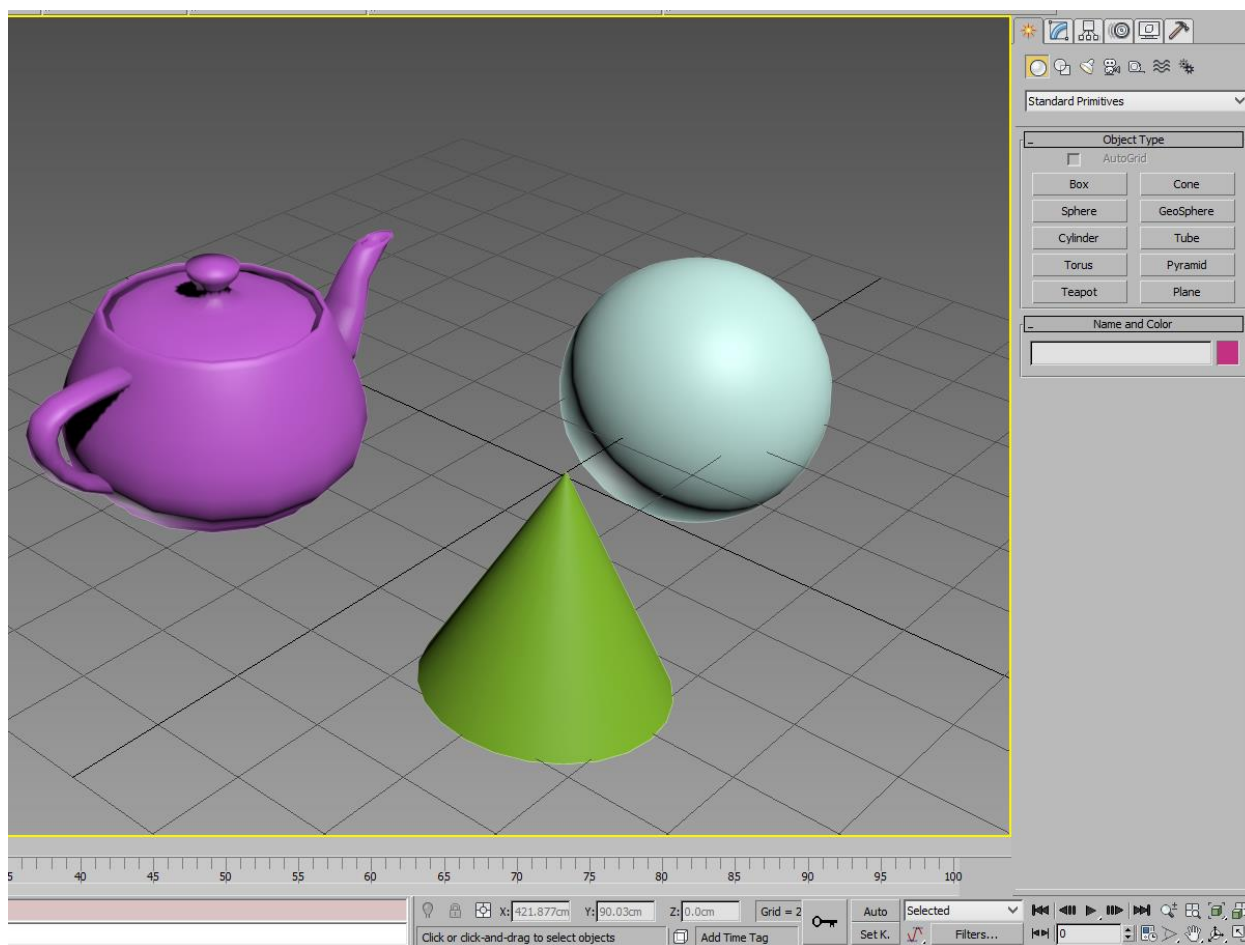
Getting Started

1. Copy the file: "Armin Attribute Holder.ms" into the directory of startup scripts. Example:

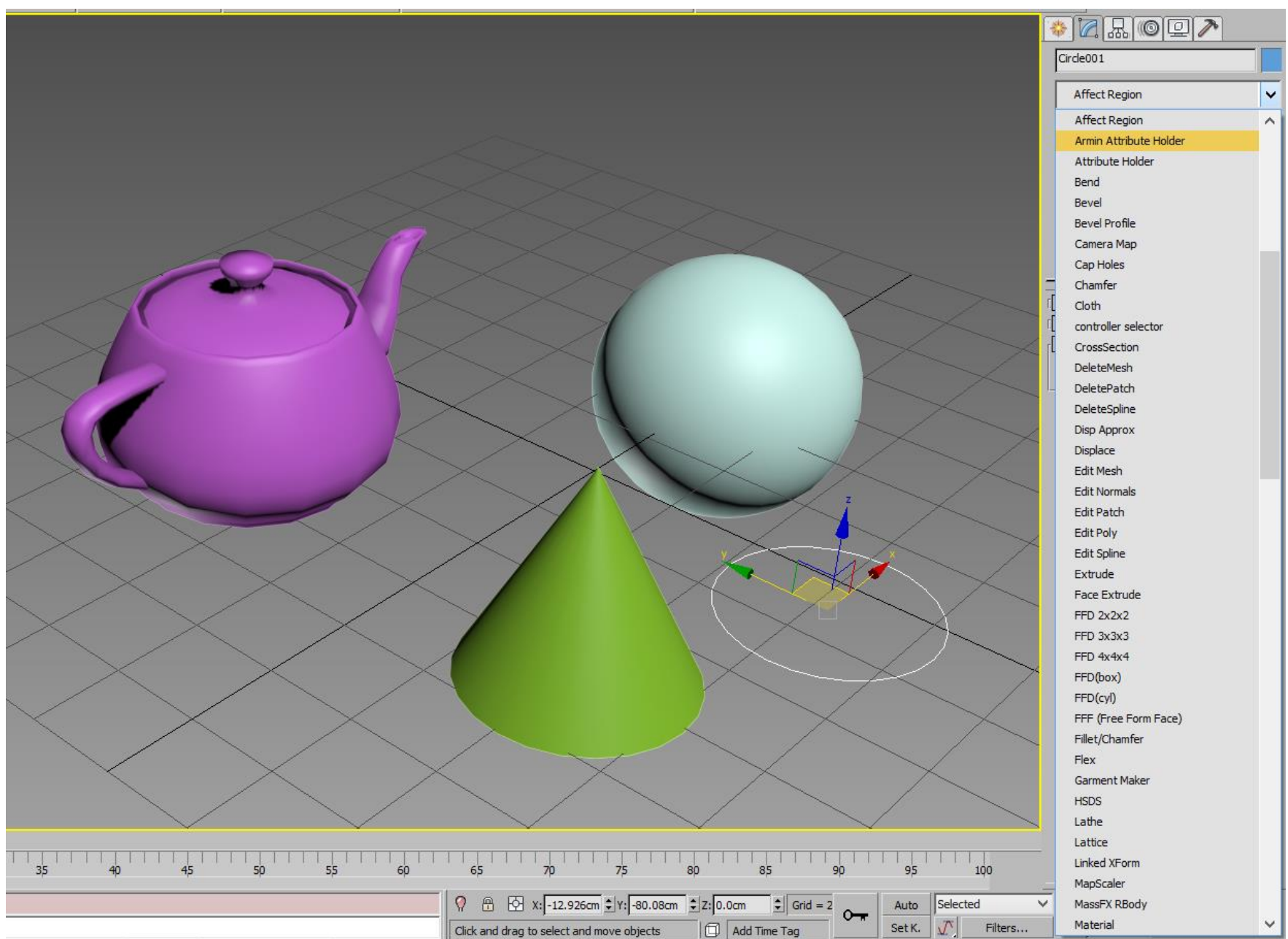
E:\Program files\Autodesk\3ds Max 2015\scripts\Startup

You can get the file from [Here](#)

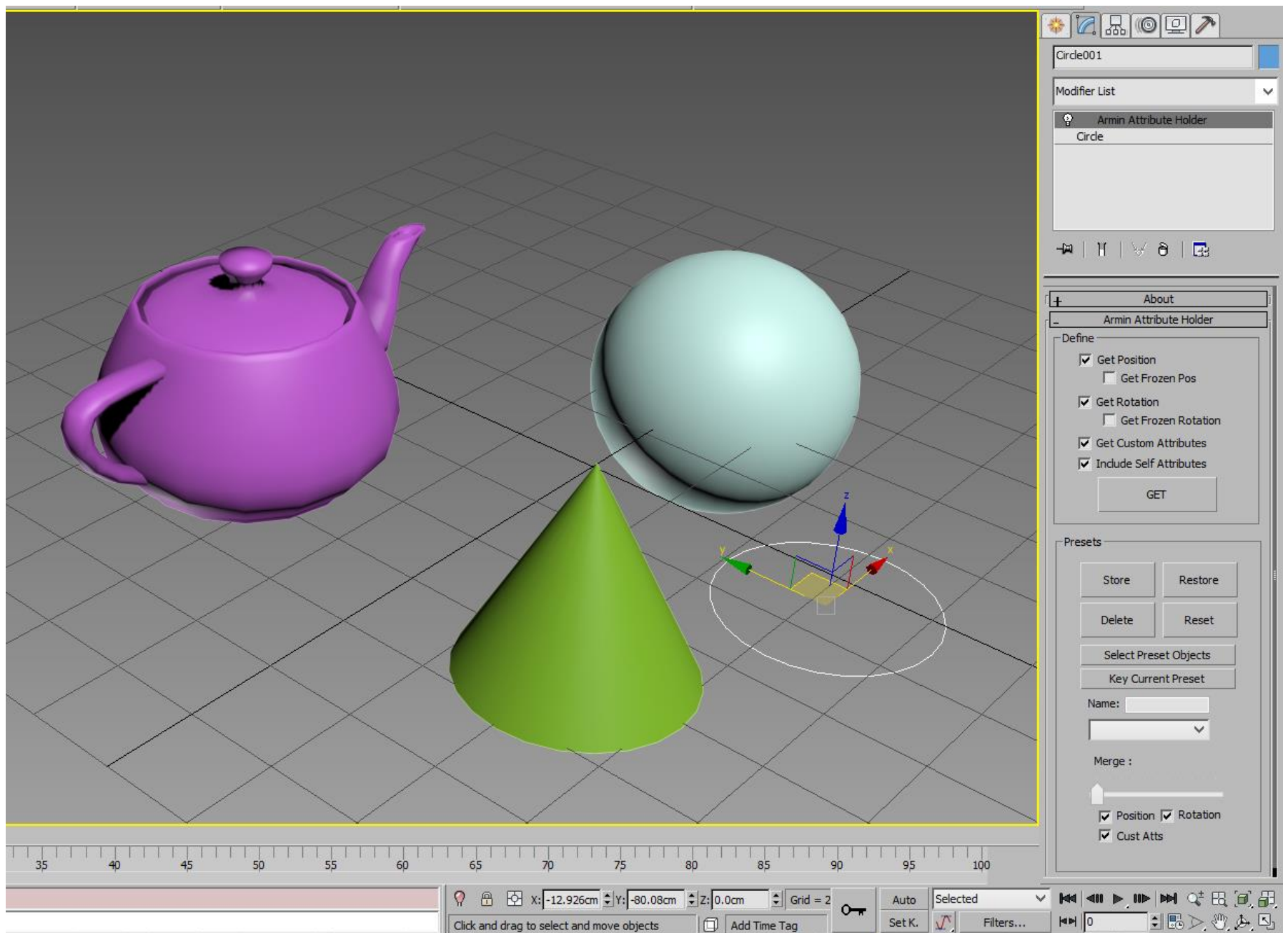
2. Restart 3Ds Max. Open a scene file that contains objects, or create a new scene file and create a few objects.



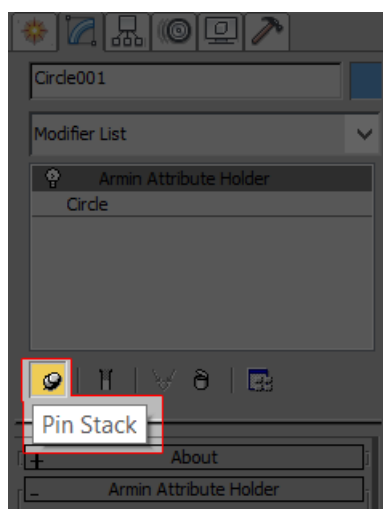
3. A – Create a new object to use as the holder of the modifier and select it. Usually we use shapes as animation controllers – like a *Circle* – when Rigging a character.
B – Go to the modify panel and open the modifiers dropdown list. You should see "Armin Attribute Holder" in the list. Select it :



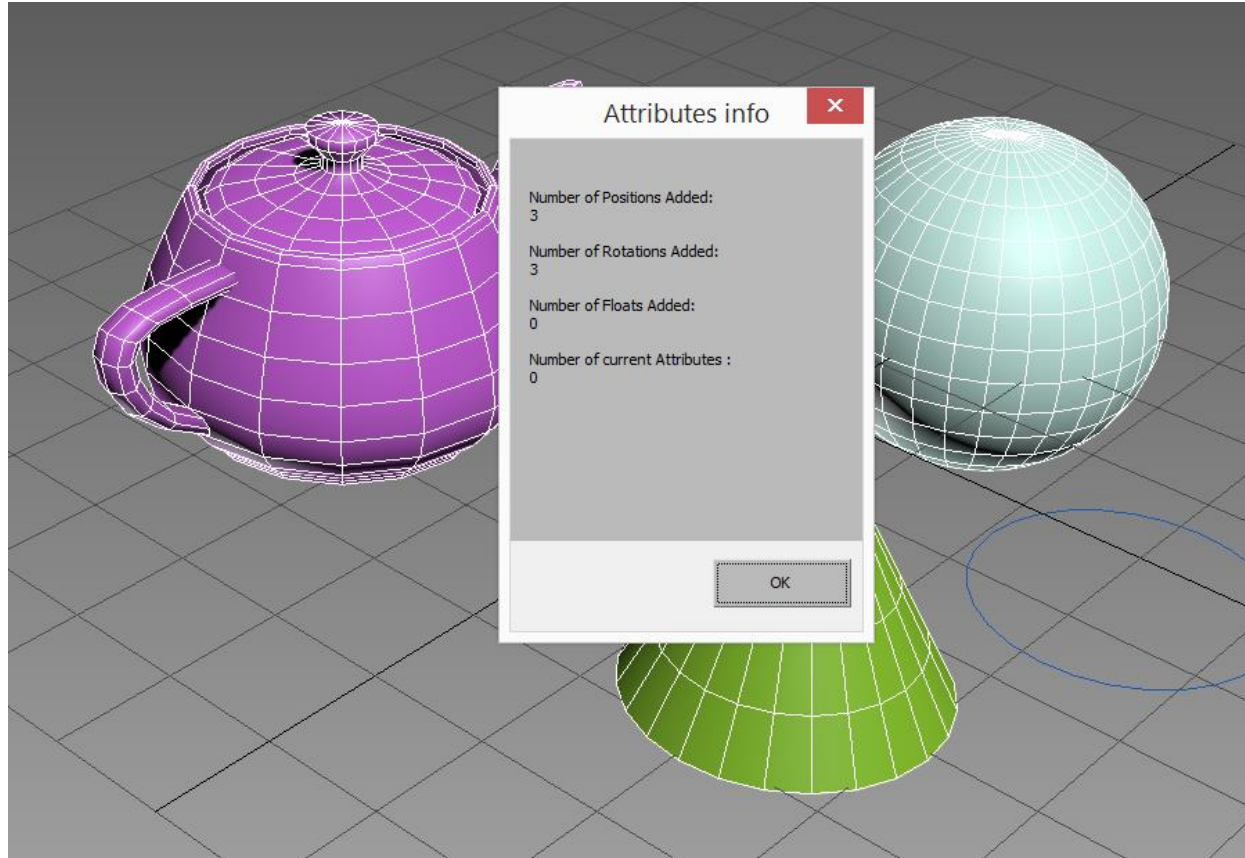
The default layout of the plugin should look like this:



4. Active the *Pin Stack* button under the modifiers list in the modify panel. After that you can now freely select other objects while still accessing to the plugin's functions:



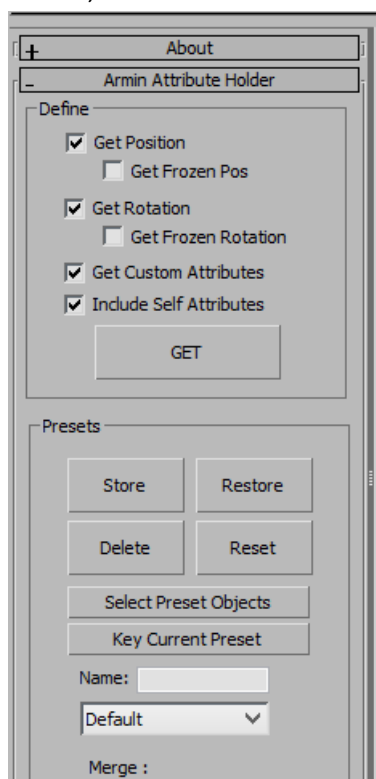
5. Select objects that you want to store their *Position* ,*Rotation* controllers or *Attribute holder* modifiers.
6. Press the “GET” button under the modifier’s rollout. Leave its settings to default. You should now get a message like this:



This message will inform you if you have successfully added the positions or the rotations or the Custom attributes of the selected objects. Note that the number shown in the picture above may vary. But you should at least get 1 Position and 1 Rotation in the message. Note that at this level, you haven’t stored any information about the position or the rotation of the objects. This step only assign the Position or the Rotation controllers of those objects -or Custom attributes of the objects – if they have any- - to the plugin.

After that, you are able to Store the position and the rotation or the Custom attributes of those objects.

7. Now, you may deselect your objects. You don’t need to have them selected to store their transformation after doing *step 6* . Under the Plugin rollout, enter a name into the “Name” field, like “Default”. Then press the “Store” button under “Presets” group :



Now you should see the name you gave to your preset in the presets dropdown list. That name represent the state of transformations of the objects defined for the plugin –step 6- when stored –step 7.

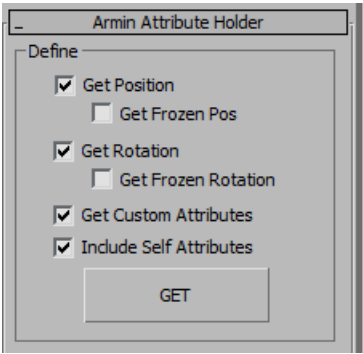
8. Move around your objects and change their position and rotation.
9. Just like step 7 , enter another name for a new preset like “state 1”, and then press the “Store” button. The new preset now should be shown in the presets dropdown list and be accessible.

10. From the presets dropdown list, choose “Default” preset, and then press the “Restore” button. Now your objects should return to the position and the rotation of when you stored them as the “Default” preset.
11. From the presets dropdown list , choose the second preset you have created in the step 9. This time use the “Merge” slider to restore the “State 1”. The “Merge” slider will restore your selected preset by percent, instead of just restoring chosen preset straight off.
12. You can add new objects to the plugin any time. The previous stored states will remain preserved but wouldn’t affect the newly added objects. You need to store new presets in order to have the newly added objects stored.

In depth help

Main rollout : “Define” group:

Overall explanation: Changing this group settings only works before you press the “GET” button. After that, changing these settings affect only for the next time that you press the “GET” button.



Get Position

Will assign the position controllers of the selected objects to the plugin.

Get Frozen Pos

when active, the plugin will get the first position controller under the main *Position list* of the selected objects. Otherwise, the plugin will leave the first position and start from the 2nd position controller under *the Position list*. It is inactive by default. Having it deactivated is good when you have frozen transform on an object in order to work only in the second position controller under a *Position list* controller. In this case, if you active the first controller and then change the object’s parent, an already stored preset for your object will remain intact.

Note: as explained above, in case you have a *position list* controller on an object, and you have stored the position of the object by having “Get frozen Pos” active, you may not change its parent or the stored states will not return the position of your object to the state of when it have been stored. Changing an object parent is also not recommended if you have its state stored where its position controller is “*Position XYZ*” at the root position controller.

Get Rotation

Will assign the Rotation controllers of the selected objects to the plugin.

Get Frozen Rotation

Just like the “Get Frozen Pos”, when active, by pressing the “GET” button , the plugin will include also the first Rotation controller of the Rotation List controller. Read “**Get Frozen Pos**” for more info.

Get Custom Attributes

When active, if an object in selection have a custom attribute under “Attribute Holder” or “PEN_Attribute_Holder_2” modifier, will be assigned to the plugin and can be stored or restored afterward.

Note : There is No limitation in the number of modifiers on a single object or the number of defined custom attributes under that modifier.

Include Self Attributes

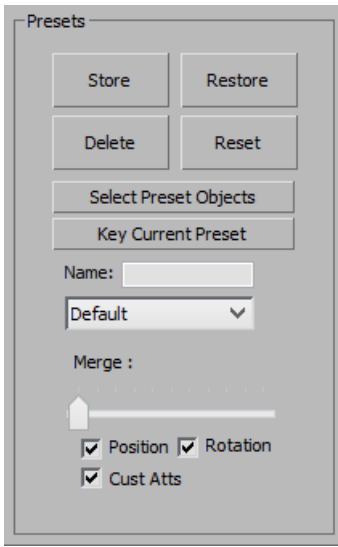
will assign the Custom attributes defined for this modifier – *Armin Attribute holder*- which means you can have custom attributes directly added to the “*Armin Attribute Holder*” Modifier and store them.

GET

By pressing it, the plugin will get the controllers of the selected objects based on the “Define” group settings.

Note: Pressing the “GET” button multiple times with different settings, will only expand the range of assigned controllers. So you could use it to assign the new controllers or include more objects, but you CANNOT use it to exclude an already assigned or “gotten” object/controller/Custom Attribute.

Main rollout : “Presets” group:



Store

Will store the current state of already assigned controllers into a new preset.

Note: Store only works when the “Name” field is filled.

Restore

Will Restore the state of controllers stored on the selected Preset. (Did you need to know that, really?)

Delete

Will Delete the selected preset in the presets dropdown list. (really? This one too?)

Reset

Will Eliminate all the presets and controllers defined for the plugin. (Come on!)

IMPORTANT NOTE: THIS ACTION IS NOT UNDO-ABLE. USE WITH PRECAUTION.

Select Preset Objects

Will select all the objects that have been gotten until the point the current state was created.

Note: the Position/Rotation/Cust Att check boxes under this group affect the type of objects to choose. For example, if you have an object stored only with position controller assigned and other objects stored with rotation controllers , you can select only the objects will position assigned to the plugin by Checking the “Position” check box and leave other check boxes inactive.

Key Current Preset

Will set an animation key for the controllers included in the current preset.

Note: To set a key, the 3Ds Max “*Animate*” button must be active.

Note: the Position/Rotation/Cust Att check boxes under this group affect the type of which controllers to set a key for. For example, if only the “Position” check box is active , by pressing “Key Current Preset”, you will only set a key for Position controllers of the current state. If all the check boxes are off, it will set no key for the controllers.

Name

Use this field to enter the name of your preset.

States (drop down list)

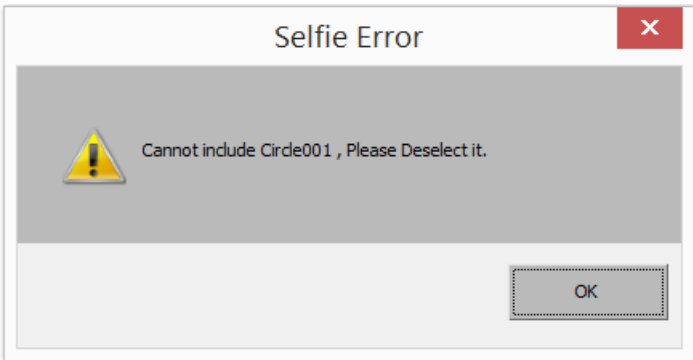
Defines which state to work with.

Merge (Slider)

Will restore the selected state by percent instead of restoring that state straight off. It can be used to merge different states. It is more useful when you are storing the position of a bunch of face controllers that have many presets like “Happy” , “Angry” and “surprised” , so it can transform between various states and create unique expressions.

Warning messages

- **Selfie Error**



It shows when : the object containing “Armin Attribute holder” is in selection while you press the “Get” button

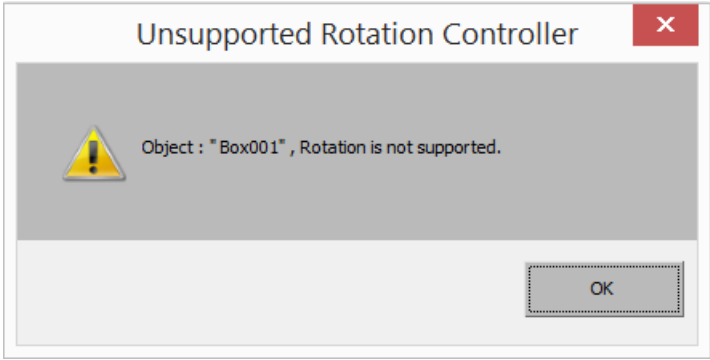
To solve: Just deselect the object you have applied the “Armin Attribute holder” which you are using now.

- **Unsupported Position Controller**



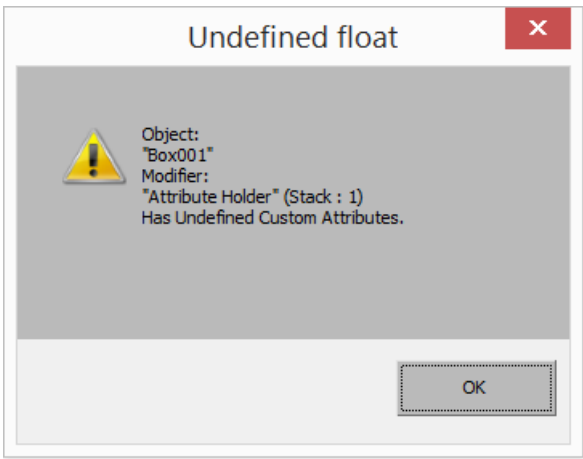
It shows when : The position of at least one of the objects in the selection set is not “Position XYZ” or “Position List”.
To solve: Just locate the named objects in the message and give them a valid Position Controller.

- **Unsupported Rotation Controller**



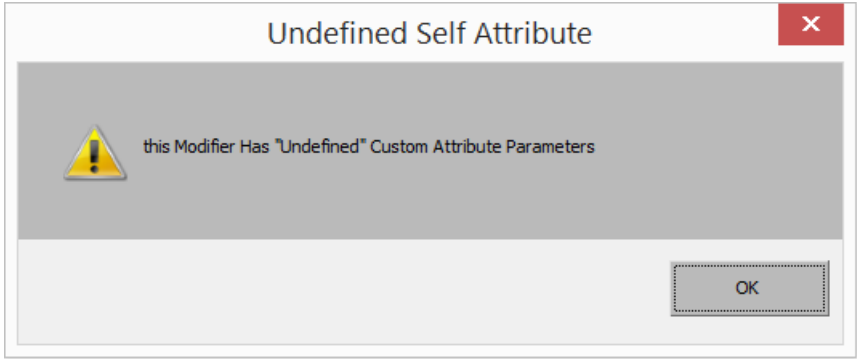
It shows when : The Rotation of at least one of the objects in the selection set is not “Euler XYZ” or “Rotation List”.
To solve: Just locate the named objects in the message and give them a valid Rotation Controller

- **Undefined Custom Attribute**



It shows when : One of the objects has a custom attribute modifier but its parameter’s controller is undefined. Note that when adding custom attributes to an attribute holder modifier, the default controller of the added parameter is not defined at the time.
To solve: Locate the parameter pointed by the error message and simply set a keyframe for that parameter once and it will be defined. Then you can delete that keyframe. If you have other parameters wired to your custom attribute or already assigned a “Bezier float” controller to it, you should not get this message at all.

- **Undefined Self Attribute**



It shows when : You have custom attribute added to the current modifier – “Armin Attribute Holder” – but it is not defined yet. Note that when adding custom attributes to an attribute holder modifier, the default controller of the added parameter is not defined at the time.
To solve: simply set a keyframe for parameters added to the plugin once and it will be defined. Then you can delete that keyframe. If you have other parameters wired to your custom attribute or already assigned a “Bezier float” controller to it, you should not get this message at all.

Final word

I hope you find this free plugin useful so it become a part of your every project. Feel free to share it with others and let me know if you have any problem with the plugin or suggestions to make it better.

Armin Soleiman Boroujerdi

www.ArminSoleimani.net

Armin.Soleimani.87@gmail.com