



# PEGASUS

By Procedural Worlds

Pegasus is a system that enables the creation of fly throughs and cut scenes.

Version 1.5.0.0 – December 2016

# Contents

|                                     |    |
|-------------------------------------|----|
| Installation.....                   | 2  |
| Online Support .....                | 3  |
| Concepts .....                      | 4  |
| Workflow .....                      | 5  |
| KeyBoard Shortcuts .....            | 7  |
| Chaining Pegasus.....               | 9  |
| Pegasus Triggers & Extensions ..... | 10 |
| Control Pegasus Trigger .....       | 11 |
| Control Animation Trigger.....      | 12 |
| Control Helios Fade Trigger .....   | 12 |
| Pegasus Manager Settings.....       | 13 |
| Pegasus POI Settings .....          | 15 |

# Installation

Installing Pegasus will create the following folder structure:

**Pegasus:**

**Demo:** A simple demo scene

**Documentation:** Pegasus documentation

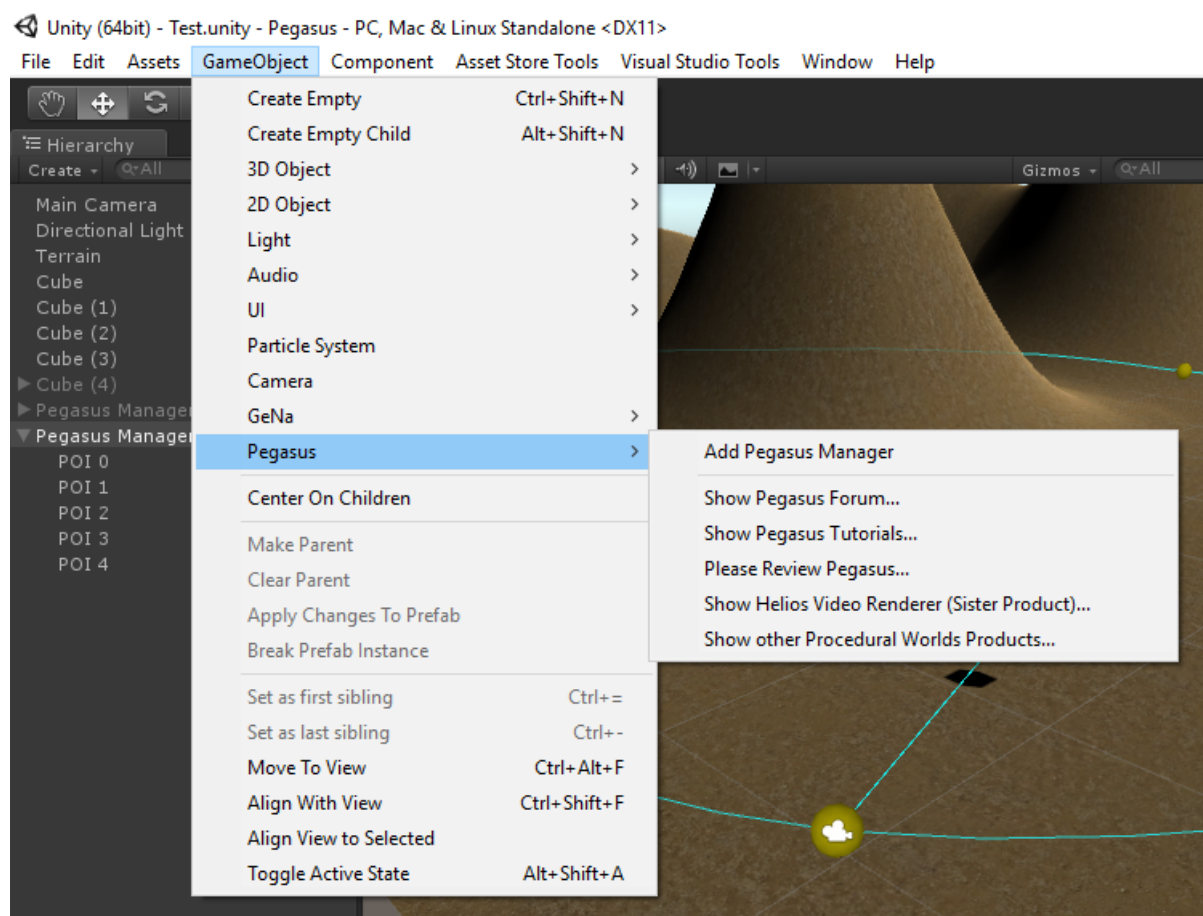
**Scripts:** Pegasus source code

## Online Support

Nothing annoys me more than purchasing a product and then not being able to get the help I need. I pride myself on the level of support I deliver and make it easily available. To access please select the GameObject -> Pegasus menu and then one of the sub-menu's. This will take you directly to the relevant web pages.

If you have a question, then please post it in the Pegasus forum. Quite often the community will get to it more quickly than me as chances are I live in a different time zone than you. In fact, living in Australia almost guarantees this ☺ Please allow a few days for a response, and if it takes a little longer then don't hesitate to ask again. I try to ensure that all questions have a response within 72 hours, but sometimes life gets in the road or emails get marked as spam, and it may take a little longer.

Like all my products, Pegasus took a significant amount of time and love to develop, and after 30 years of commercial development it still surprises me how difficult it can be to make complex things seem easy. If you do like find that you like Pegasus, and would like to see further development then please take the time to review it. Reviews impact the decision of others to purchase, and sales are what makes it financially possible for me to keep on bringing better products to you.



## Concepts

Pegasus allows any object to be driven through a scene and provides fine grained control of its speed, location and what it is looking at.

While Pegasus was designed to create fly throughs and cut scenes, it can drive any object that drop into its Target Object slot.

The main Pegasus components are:

Pegasus Manager – Pegasus manager controls the position, location and direction of an object over time within the scene.

Pegasus POI – Pegasus POI or points of interest are the locations that the flythrough will pass through as it is executed.

## Workflow

To get the quickest workflow with Pegasus update your screen so that you can see both the Scene and Game windows at the same time. Pegasus will update Game view in real time so that you can see how the camera view changes as you make changes to POI and their LookAt targets. This allows you to compose your shots. One way to do this is to select Windows->Layouts->2x3.

1. Add a Pegasus Manager to your scene by selecting GameObject->Pegasus->Add Pegasus Manager.
2. Drag your camera or target object into the Target Object slot on the Pegasus Manager – this is the thing that will be controlled by Pegasus when Play is pressed.
3. Add Pegasus POI to your scene by clicking Ctrl + Left Mouse Button at each location you would like your target to pass through.

You will see a Gizmo place at every location on which you click, and when you place two or more, they will be joined up by a path visualisation spline.

The path visualisation spline allows you to visualise where Pegasus will drive your target.

4. Press Play to start your scene. Pegasus will start playback when you start your scene.

From here:

- Change the settings in your manager to change the way the overall flythrough operates.
- Customise individual POI segments to change the way the flythrough operates through that POI segment.

### Banking Around Corners

If you want to simulate speed, try rotating your POI in the x or z axis as you go around corners. Make sure you visualise it in the game view. Works a treat and is a super nice feature!

## **Bad GameObject rotations**

Sometimes you have a game object that does not have the rotation configured properly. You through each POI and update the rotation at that point to correct it. Pegasus lerps rotation through each POI, so this can be used to counteract the issue.

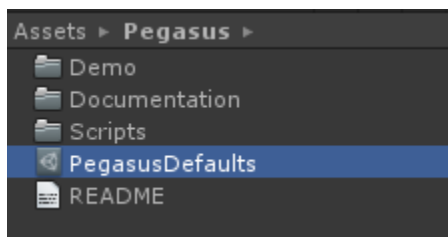
## **Keyboard Shortcuts**

If you use the keyboard shortcuts in conjunction with your Scene and Game view windows, you get the ability to visualise and compose what your audience would see in Game view at run time, even in edit mode. Check out the KeyBoard Shortcuts section below and the tutorial videos to see this in action.

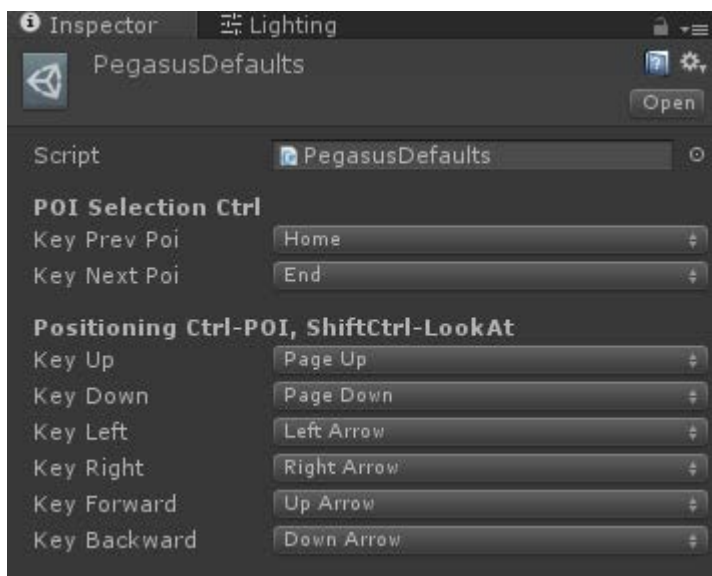
## KeyBoard Shortcuts

The keyboard shortcut system allows you to fine tune and visualise your flythrough. If you take the time to master it (it's not that hard), it will provide a massive speedup to your workflow and very precise control of the flythrough.

These shortcuts are editable, and are stored in the "PegasusDefaults.asset" file in your Pegasus directory in the hierarchy window.



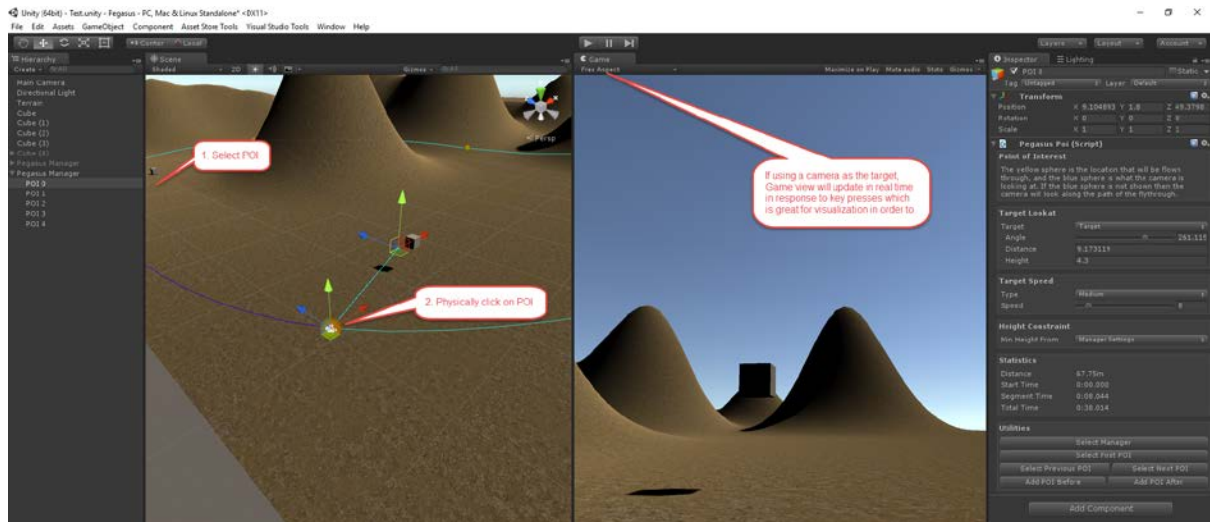
When you select and view it, you can the edit the keypresses that Pegasus listens for:



To physically use them double click on a POI, and then physically click on it in your scene editor window. Hitting CTRL and one of the key clicks will move the POI, and SHIFT+CTRL and one of the key clicks will move the Lookat Target.



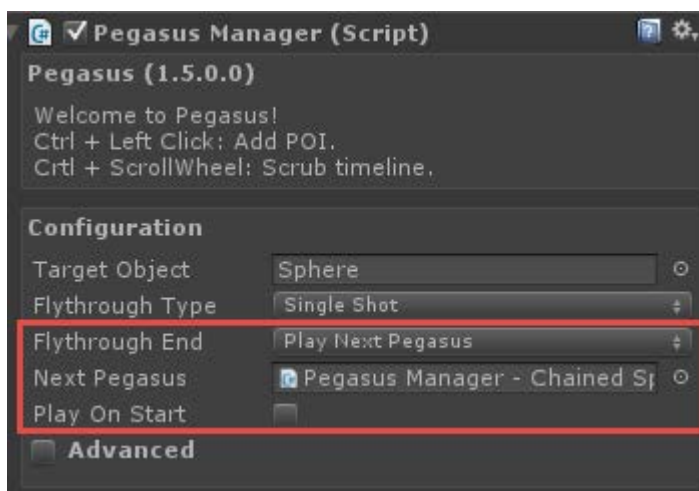
If you are driving a Camera with Pegasus a nice trick is to open both the Scene and the Game window side by side. You can visualise the impact of these settings as the player would see them in the Game window and it's a great way of quickly and accurately composing your screens:



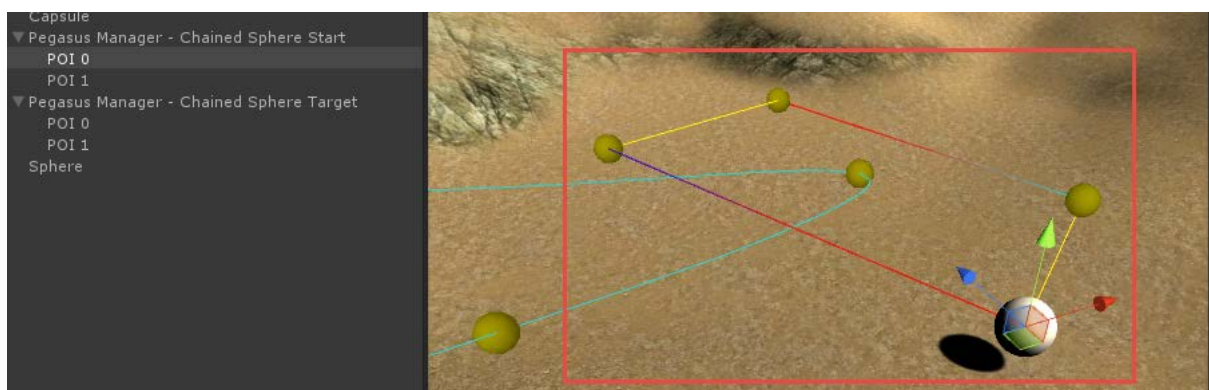
## Chaining Pegasus

Often when you are creating a flythrough, you will want to change the camera angle as if to show off a new perspective, or just to start a radical new motion. This can be done by chaining multiple Pegasus together.

Set each Pegasus up as a Single Shot Pegasus, and then chain them together. Make sure that the child / dependent Pegasus do not have "Play on Start" selected, as they will play when the scene is started instead of when the Previous Pegasus has been completed.



Take a look at the demo scene for an example of chained Pegasus. The yellow line shows the linkage between the two Pegasus. In the example below the Pegasus have been chained together to form an infinite loop.



## Pegasus Triggers & Extensions

Pegasus has a trigger and extension system that allows you to use Pegasus as a way to trigger location and time based behaviours in your scene. You can add as many triggers as you want to a POI, all will get executed at run time.

Triggers are derived from the TriggerBase class and attached as GameObjects to your POI's. Pegasus calls 3 key methods on the trigger script over the duration of the segment playback to allow you to create sophisticated behaviour.

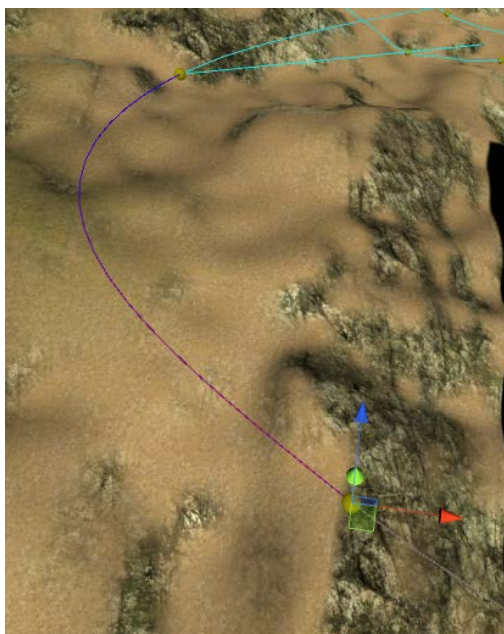
The trigger events are:

On Start: Called as soon as playback starts on the POI segment. Used to do things at the start of the segment playback.

On Update: Called every frame of the playback, and passed the progress through the segment. Used to do things during playback.

On End: Called when playback ends on the POI segment. Used to do things at the end of the segment playback.

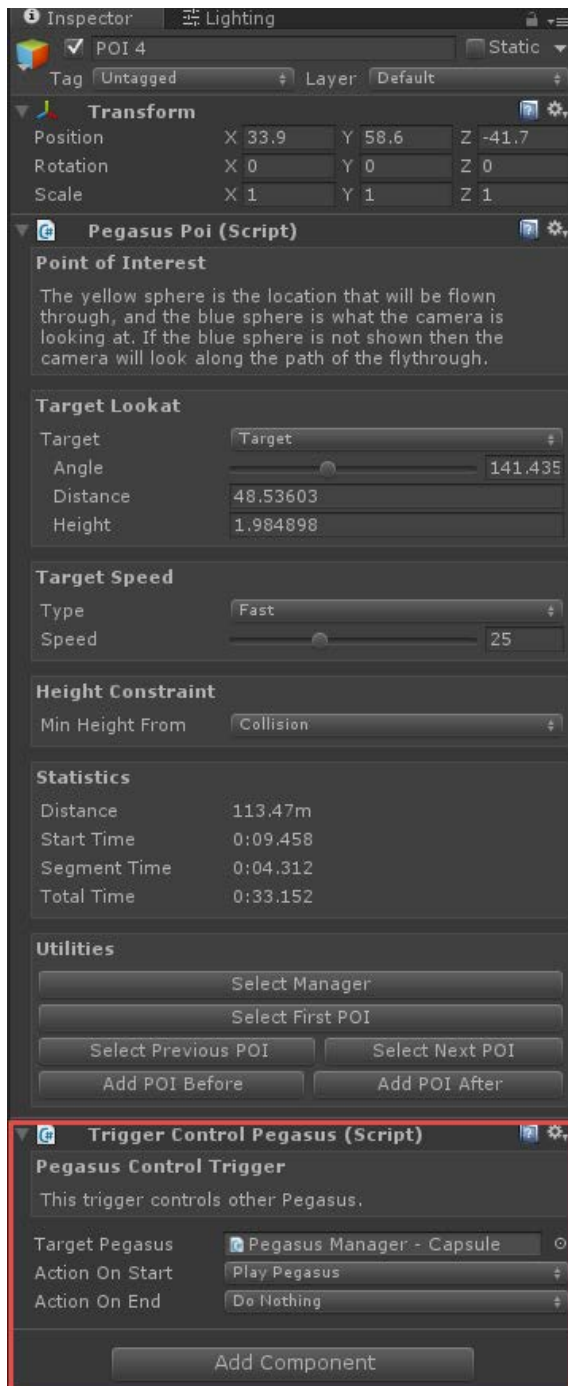
Have a look at the code in the demonstration triggers provided to get a sense of how you could use triggers to extend the capability of Pegasus.



*NOTE: When you select a POI, the segment it controls is shown in a different colour so that you can visualise its trajectory.*

## Control Pegasus Trigger

This trigger is attached to a POI and can be used to control the playback of other Pegasus when the segment is active.



## Control Animation Trigger

This trigger is attached to a POI on a Pegasus that is controlling something that can be animated such as a person or animal. It can be used to control animation over that segment.

## Control Helios Fade Trigger

This trigger is attached to a POI on a Pegasus that is controlling a Helios camera, and can be used to control playback fades to and from a colour e.g. dip to black. It's a great way to visually join chained Pegasus together, and can be used to export video that needs minimal post processing.

# Pegasus Manager Settings

The Pegasus Manager orchestrates and controls a Pegasus flythrough. You can have as many managers as you like in your scene.

These settings can be viewed by hovering over the setting in the editor.

NOTE: Make sure that each Pegasus manager has the correct framerate setting selected and that all of them are the same. If you are using Pegasus with Helios please also make sure that these settings is in alignment with the frame rate settings for Helios.

*Target Object:* The object that will be controlled by Pegasus manager. You would typically drop a game object with a camera attached to it - but this could just as easily be any game object you wanted to drive through the scene.

*Flythrough Type:* The type of flythrough - a single shot or a connected loop.

*Flythrough End:* What to do at the end of the flythrough. Stop - Stop the flythrough. Quit application playback or Play the Next Pegasus - to start another Pegasus flythrough (great for changing camera angles).

*Next Pegasus:* Plays the next Pegasus flythrough then this one has ended, only available when Play Next Pegasus has been selected.

*Play on Start:* Plays the flythrough on start up when selected.

*Advanced:* Shows advanced options when selected.

*Framerate:* The framerate that the game will be controlled at. Set V Sync Count to Don't Sync in your project Quality settings or Unity will ignore this setting.

*Check Height:* Used to control how poi, lookat target and flythrough path heights are constrained. Collision - use whatever it collides with, Terrain - use the terrain height, None - don't constrain.

*Min POI Height:* The minimum height that POI and collisions will be tested for.

*Rotation Offset:* An offset that will be applied to all rotations. Used to fine tune rotation on objects being driven, and quite useful for fixing broken rotations on game objects.

*Rotation Damping:* The amount of damping or smoothing to apply to the rotation of the target. Larger values mean slower rotations.

*Position Damping:* The amount of damping or smoothing to apply to the position of the target. Larger values will do smoother flythroughs, but with less precision through POIs so it should be used with care.

*Gizmo Size:* The size of the Gizmos. Larger Gizmos are easier to see.

*Statistics:* Handy statistics about the current flythrough.

*Distance:* The distance of the flythrough.

*Duration:* The duration of one loop of the flythrough.

*Visualisation:* Allows the trajectory of the flythrough to be visualised in editor. If using a target object that has a camera then click on the game view to see what the camera looks at and how it progresses through the scene.

*Scrubber:* Drag this control to move the target along the timeline - designed for edit mode visualisation. Select the Game View to get the best effect.

*Step Backward:* Step backwards through the flythrough.

*Step Forward:* Step forwards through the flythrough.

*Show Debug:* Show flythrough debug messages.

*Utilities:* Some simple and handy utilities to aid in flythrough creation and visualisation.

*Go To First POI:* Select the first POI in the scene editor.

*Set POI To Min Height:* Sets all POI to the Min POI Height shown in the editor.

*Show Debug on POI:* Displays a cross centred on the POI.

*Hide Debug on POI:* Hides the cross centred on the POI.

# Pegasus POI Settings

The Pegasus POI controls where and how the target will travel through the segment controlled by the POI.

When selected in the editor, the segment changes colour so that you can see its path.

These settings can also be viewed by hovering over the setting in the editor.

*Target Lookat:* These settings control what the target will look at when passing through this segment.

*Target:* Where the target should look. Path - the target will look along the path of the flythrough. Target - the target will look at a custom target.

*Angle:* The angle from the POI to the camera target.

*Distance:* The distance from the POI to the camera target.

*Height:* The height of the POI above the terrain or collider at the target location.

*Target Speed:* These settings control how fast the target will travel along this segment.

*Type:* Change the flythrough speed in common units.

*Speed:* Manually control the flythrough speed.

*Height Constraint:* How the target will be height constrained. Stops the target from going under the things it shouldn't e.g. the terrain.

*Min Height From:* Used to control how poi, lookat target and flythrough path heights are constrained. Manager - use the managers settings, collision - use whatever it collides with, terrain - use the terrain height, none - don't constrain.

*Statistics:* Handy statistics about the current POI's segment.

*Distance:* The distance of the segment.

*Start Time:* The time after the start of the playback that this segment will start.

*End Time:* The time after the start of the playback that this segment will end.

*Total Time:* The duration of one loop of the flythrough.

*Utilities:* Some simple and handy utilities to aid in flythrough creation and visualisation.

*Select Manager:* Selects the POI manager in the scene editor.

*Select First POI:* Select the first POI in the scene editor.



*Select Previous POI:* Select the previous POI in the scene editor.

*Select Next POI:* Select the next POI in the scene editor.

*Add POI Before:* Add a POI before this POI in the scene editor.

*Add POI After:* Add a POI after this POI in the scene editor.

Deleting POI can be done by simply deleting the POI object that has been parented under the Pegasus Manager object in the scene hierarchy. To complete the deletion, click on the Pegasus Manager.

Re-Ordering POI can be done by simply dragging the POI object to a different location underneath the Pegasus Manager object in the scene hierarchy. To complete the re-organisation just click on the Pegasus Manager.