



VERSION 8.0.1.XXX - DEVELOPED BY: AVOLITES MEDIA LTD. POWERED BY: SALVATION (TM)
HARRISON DIGITAL MEDIA INC. DRIVEN BY: DAVE GREEN, TREY HARRISON, STEVE WARREN, J.B.
CIARAN ABRAMS, SELVIN COOPER, ARRAN ROTHWELL-EYRE, SIMON GRAHAM, MARY, PRADS,
GREG, ADAM, OLIE, ALL AT AVOLITES AND YOU. SPECIAL THANKS MARK CALVERT. LETS ROCK!

Mapping Moving Scenic Elements

VERSION 8.X

By Arran Rothwell-Eyre December 2014

with Additions by Dave Green & Ciaran Abrams

Mapping Moving Scenic Elements

With a little preparation and accuracy, complex video mapping of moving scenery can be quickly and easily achieved from within Ai – a good example of Ai doing this is can be seen on Bon Jovi's 2013 “Because We Can” tour.

For each element of moving scenery we have in the project, we use a module called “3d Morph Model”. This module lets you load multiple models at the same time and then use a controller or system of your choice to interpolate between the models within a 16 bit 0 to 1 range. If the models used are saved with different positions or XYZ coordinates, the resulting interpolation will give a smooth movement between the different positions.

Working with modules instead of using an independent fixture for each component allows us to work with a higher amount of moving components, provides much better system performance and becomes much easier to perform live playback control.

First of all, we add a screen fixture to our project. The way we construct this project means that we don't get a fixture icon for the moving components within Ai, so we use the screen fixture itself to supply the texture and retain the benefits of the media tiles and layers and then send the resulting video to our components. Once the correct size and resolution has been set for the screen fixture, we can just push it off to the side – a value of -100 on X should be adequate - its still within the project, we just don't need to see it. We also need to add a projector to the project and set its values according to the specification of your real world projector.

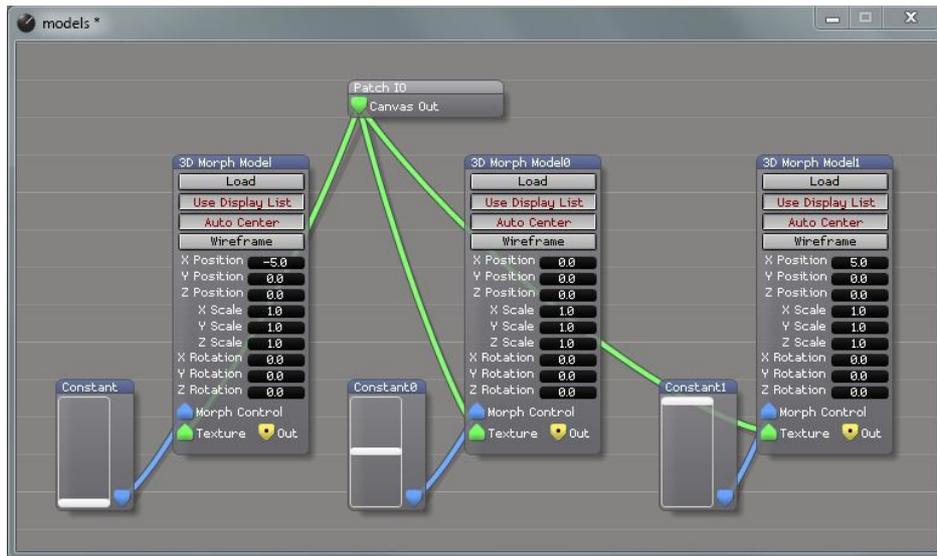
Now open the Stagepatch. firstly, we need to create an empty patch – do this by right clicking the mouse in an empty area and choosing “Insert Empty Patch”. Once placed, rename this new patch to something more memorable such as “models”, then drag a connection from the canvas out of the main screen fixture onto “models”.

Next, expand the patch, and within the new window, right click and choose “Insert Patch IO”. This will allow us to take the texture directly from the screen fixture and apply it to the moving scenery components.

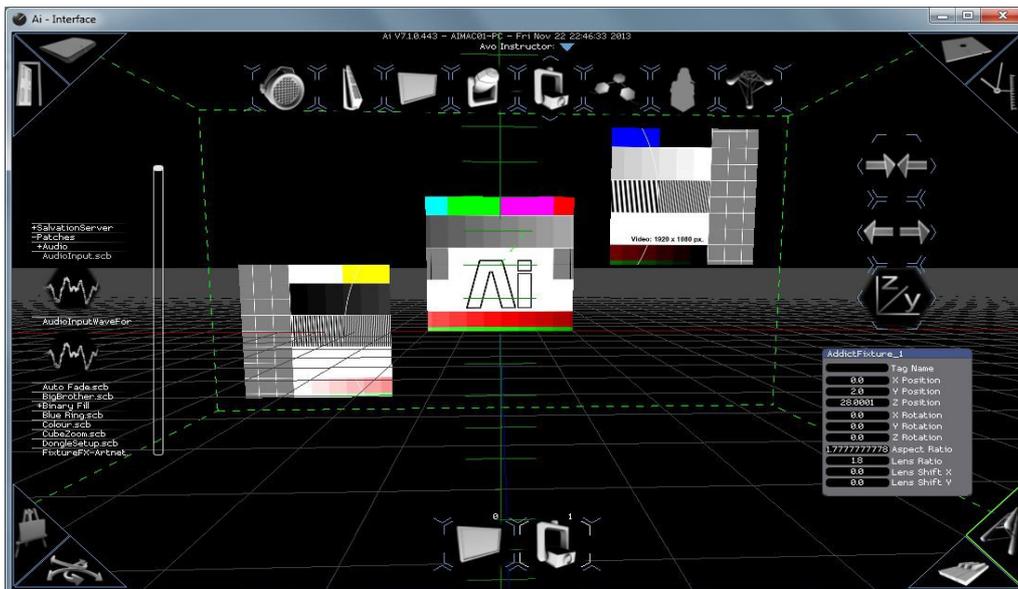
Insert the first 3d Morph Model – right click and choose Insert Module/Import/3d Morph Model. Place the module and connect the Canvas Out port on the Patch IO to the Texture port on the 3d Morph Model. Then add a controller – for now we will use a vertical fader – right click Insert Module/Controller/ Constant, place the module, right click it – choose Load Alternate Skin and select Vslider – connect this to the Morph Control port on the 3d Morph Model.

Select both the 3d Morph Model and the vertical fader and duplicate them as many times as needed to allow for every moving component in your project. Press the Load button on the first module and navigate to your saved models and multi select the first pair of models by holding CTRL and left clicking and then press Open.

Here we can see three 3d Morph Model modules within the patch each with a different value on the vertical fader so that we can see the lower , mid and upper positions of the 3 projection targets.



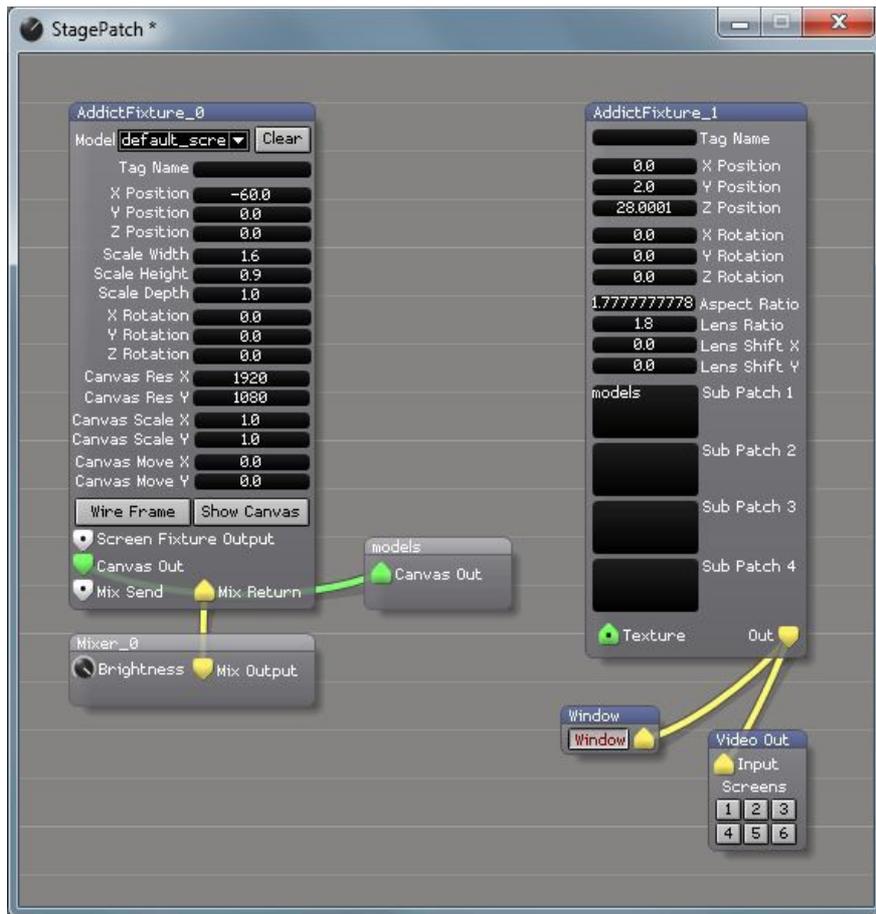
This view of the Stage Construction Page shows the models, again showing their lower, mid and upper positions. Note that the projector has been positioned so that the full range of the interpolated movement always falls within the projector's beam.



When a change in value is received at the Morph Control port, the model can be seen to move within the Stage Construction and Performance pages, the result cannot however be seen on any of your system outputs yet. Normally a fixture would be connected directly to the projector using the Screen Fixture Output / Input ports, but as we have worked outside of the fixture, we do not have these ports available to us. In order to resolve this, we need to tell the projector where to look within our project to find the models it needs to show.

In the Stagepatch, right click on the projector fixture, choose Load Alternate Skin and choose SubPatches. The SubPatches skin allows us to specify the name of a patch that can be found within the Stagepatch, so that the projector can then see the models inside of it. In the first box, enter the name defined earlier - "models". It's important to remember that the naming in these boxes is case sensitive.

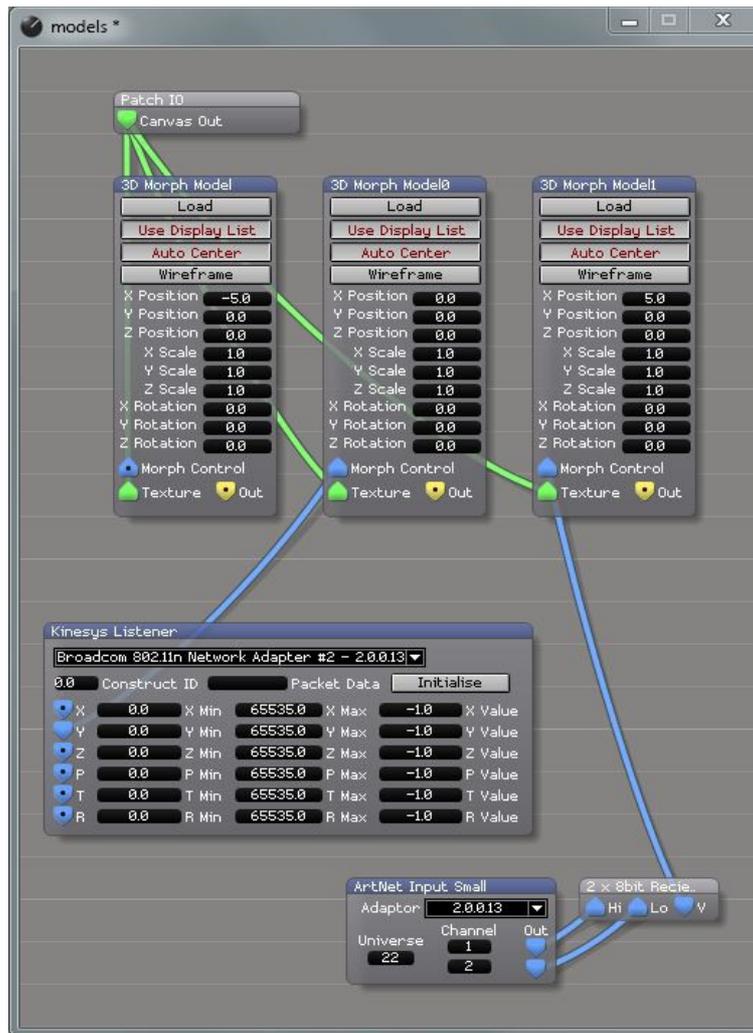
The Stagepatch should now look something like this:



Fundamentally, the project is now setup and ready to work by using the vertical faders assigned to the 3d Morph Models, however, much better results can be made by using an external control system to send a value to the Morph Control port to ensure a higher level of synchronisation of movement between the projection target and the projection itself.

Systems such as Kinesys or Navigator are designed for tasks such as these – they are essentially motor systems used to move stage elements which also give out digital values so that it becomes possible to know very accurately just how much movement has been made and on which axes, then feedback this information to Ai so that its own output reflects those changes in real time.

A major benefit these systems have is a 16 Bit value ranges (0 – 65535) unlike the 8 Bit range of DMX (0 – 255) or 7 Bit MIDI (0 - 127) - in terms of smoothly moving projection mapping, the higher the Bit value of the controller, the better. A Bit depth which is too low will make the projection stutter as the distance between steps is greater and much more obvious to the eye and you have far less accuracy in terms of precise placement of your projections. This project specific and really comes down to the distance over which the movement will make, some projects will require a more fluid control– especially when dealing with large scale projections or curved movement paths.



Other control options could be combining 2 streams of DMX to create a 16 Bit value range to send to Ai – a patch for making use of values created like this this can be found in System Patches/ 2 x 8bit Receive, extracting data from a UDP based network message using Insert Module/Network/Ethernet Receive. With a highly rehearsed show it is possible to work without real time feedback and use well programmed cues and chases from a console or sequencer – this approach can be problematic if the projection targets have much scope for deviation from the planned movement.

Key points to remember:

- All models loaded within an instance of 3d Morph Model must have the same amount of vertex points.
- Every 3d Morph Model will need at least 2 differently positioned models loaded at the same time, otherwise it will appear to do nothing.
- 3d Morph Model's can have more than 2 models loaded at the same time. The total amount of files loaded are always spread across a value range of 0 to 1 and will be interpolated between in a linear progression.

- Try to model as accurately as possible – correctly sized screens in real world positions all set in metres can be very easily dealt with, out of scale models can still work, but it can become very difficult to find the right values. Make sure that metres is your chosen unit when exporting your 3d models from your modeller of choice.
- How you UV map your different models can open up creative results when morphing between them.