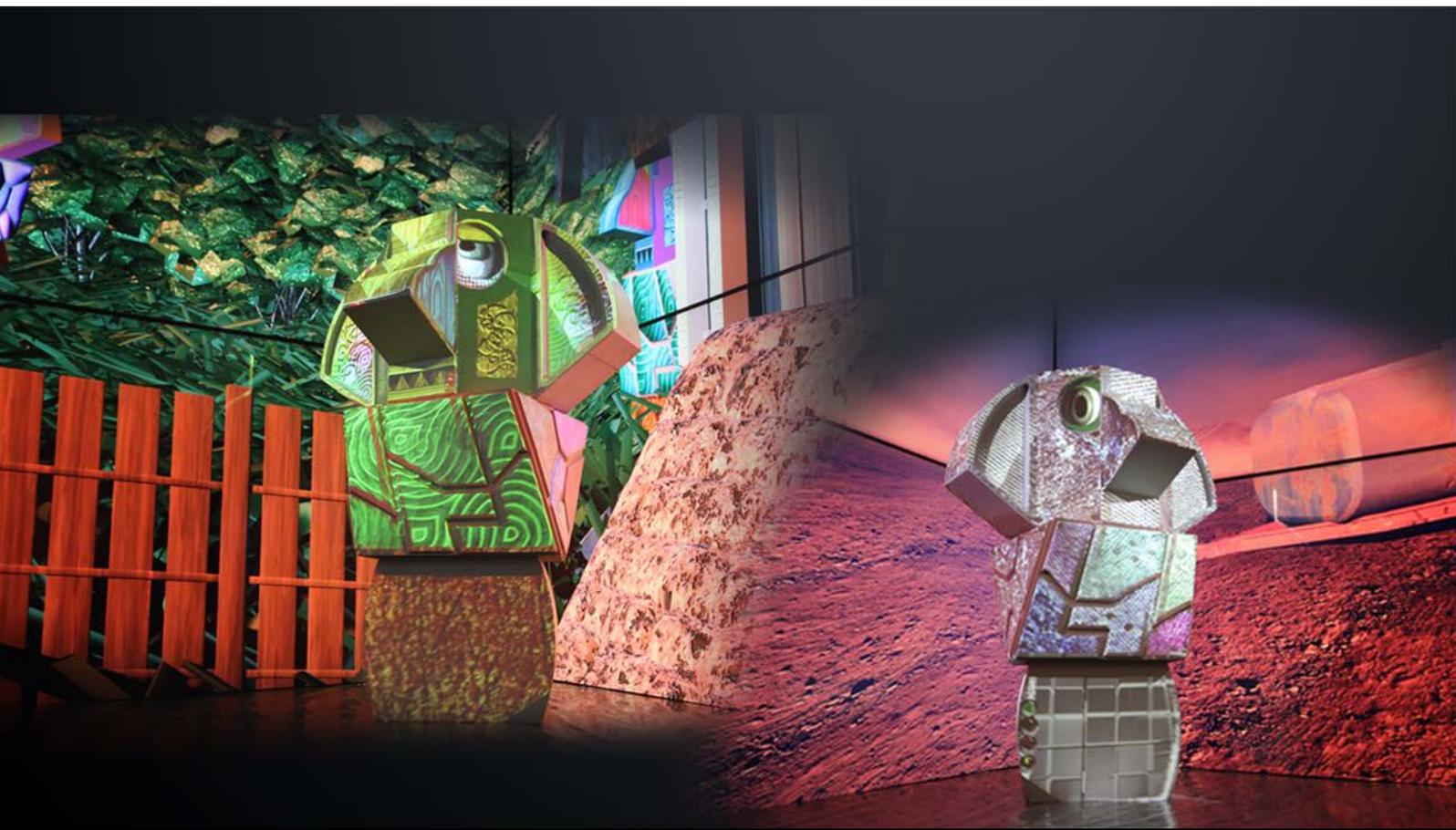




Delta Media Server

DeltaGUI Mesh Mode

User Guide



DeltaGUI Mesh Mode : User Guide

Trademark Information

Delta Media Server is a trademark of 7thSense Design Ltd. Brand or product names may be registered trademarks of their respective owners. Windows is a registered trademark of Microsoft Corporation in the United States and other countries.

Copyright Information

All Rights Reserved. This document is copyrighted © by 7thSense Design Ltd and shall not be reproduced or copied without express written authorisation from 7thSense Design Ltd.

The information in this document is subject to change without notice. 7thSense Design Ltd assumes no responsibility for errors, and/or omissions contained in this information.

Printed: July 2021

This edition is for software version Delta 2.6, 2.7
Document ref.: M078-7

E: info@7thsense.one
W: 7thsense.one

7thSense Design Ltd
2 The Courtyard, Shoreham Road
Upper Beeding
Steyning
West Sussex
BN44 3TN
UK

T: +44 (0) 1903 812299

7thSense Design LLC, Michigan
332 E Lincoln Ave
Suite 100
Royal Oak, MI 48067
USA

T: +1 248 599 2717

7thSense Design LLC, Orlando
4207 Vineland Rd
Suite M1
Orlando, FL 32811
USA

T: +1 407 505 5200

Introduction	4
Projection and Textured Modes Explained	5
Mesh Mode Setup	9
Mesh Setup Basic Sequence	11
Setting an Eyepoint in Delta	12
DeltaGUI Mesh Mode Controls	13
3D Projector Calibration	20
Warping a Mesh	30
Blending in Mesh Mode	32
Troubleshooting Projection Mode	33
Working in Textured Mode	36
Multiple Meshes and Media	37
Active 3D Mesh Mode	40
Editing a Mesh	45
Movement Nodes	52
Mesh Optical Tracking	55
Document Information	58
Index	59

Introduction



Mesh Mode is a Delta Licence option intended for projection onto complex objects or surfaces. This practice is commonly referred to as 'Projection Mapping'. Really, Mesh Mode could be utilized for even the simplest of traditional applications (e.g. a single projector pointed straight on to a flat projection screen), but most users would feel it introduces unnecessary complexity in applications where our Flat Plane mode plus 2D warp and blend tools can handle the job more simply.

DeltaServer uses a 'model-first' approach to projection mapping, meaning a 3D model (or 'mesh') representing the projection surface is prepared in advance, and then imported into DeltaServer for media texturing and projector alignment. This is opposed to a 'projection-first' or drawn-on-site workflow used by some other tools (generally more VJ-type 'rock up and do something neat' tools). The 3D model is generally instrumental to the media creation phase of a project as well, which often happens months in advance of going to the site. So, depending on whether the physical surface already exists yet in the world (e.g. a historical building) or not (e.g. some new sculpture that isn't built yet), the initial model may be software-originated or produced by laser-scanning. Often with a software-originated model, a laser scan is performed once the physical object does exist, and the planning model is then corrected against the laser scan 'point cloud' result by overlaying it in the modelling software.

This guide will take you through the process of matching your mesh model and media to the real life environment. The two major stages of projection mapping in DeltaServer are:

1. texturing content onto your 3D model in the virtual world
2. resolving alignment of the real world projectors relative to the model.

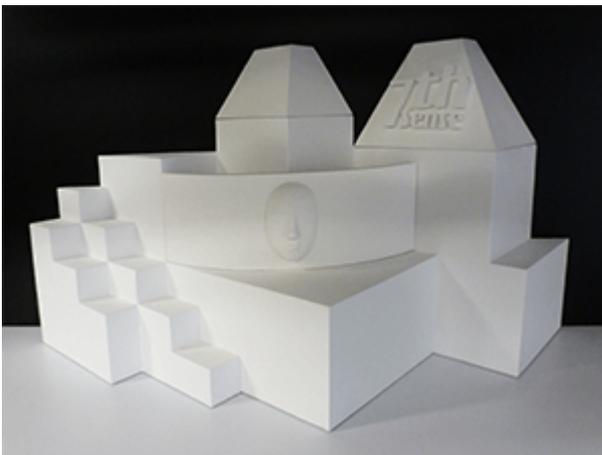
Projection and Textured Modes Explained

Projection Mode

Projection Mapping is the art of using projectors to paint media onto complex surfaces. Most projector lenses are designed for tangential mapping, meaning that when a projector's light source produces an array of pixels and passes that through, the lens is designed to produce a uniformly lit, uniformly focused rectangle of light on a flat plane surface. Success depends on the projector being pointed square at the surface, and the surface reflecting the light uniformly back to the viewer's eyes. The key challenges of projection mapping are therefore dealing with geometry and luminance (the intensity of light reflected/emitted from a surface), compensation or compromise, when projecting onto non-planar objects with varied surface properties, generally from multiple blended projectors shooting from disparate angles, and often in environments where viewer position varies greatly.

All these factors should be considered in planning a projection mapping. Simple, right? Tools like DeltaServer hide much of the challenge by predicting geometry and light behaviour in a 3D space. The resulting image relies on a sensitive static relationship between projectors and surface, so care should be taken in installing the projectors in relation to the target, in a stable manner that will change as little as possible in the future. Any changes will result in the need for alignment adjustment.

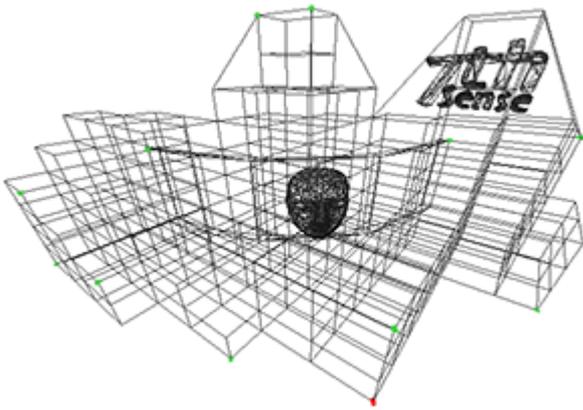
The 3D Object/Surface



Geometry and light predictions are accomplished by placing a virtual camera representing each real world projector in the software's 3D space, whose X, Y, Z, heading, pitch, roll and lens characteristics closely match that of the real world projector lens. So long as the geometry of the 3D model accurately matches that of the real world object, you will accomplish an accurate reflection of that camera's view of the textured software model back onto the real world object.

Texturing the model means defining how 2D content will be applied to a 3D mesh of faces. DeltaServer supports both explicit UV maps (a list embedded in the .obj model file, defining relationships between 2D canvas x,y percent positions to 3D points on the model), as well as eyepoint-based virtual projection.

The 3D Model



A 3D model is a system of co-ordinates ('nodes' or 'vertices') joined in a mesh. The co-ordinates take as their reference a zero-point (or **origin**) within (or outside) the model of the reference object. The model can be rotated and viewed from any angle, but when used for projection mapping, the designer will take a chosen viewpoint (or **eyepoint**) that 'projects' an orientation from which the object will be viewed.

The 2D Media



Because the designer's reference point is the origin of the model, the media effectively represents what the origin of the model would see in a mirror. For better and sharper results, the media can be displayed over several projectors, and it may be better to split a complex mesh model into more manageable pieces for those projectors, but projection mode still assumes that the designer and viewer are both looking at the media-painted object from the same view of the model.

Textured Mode

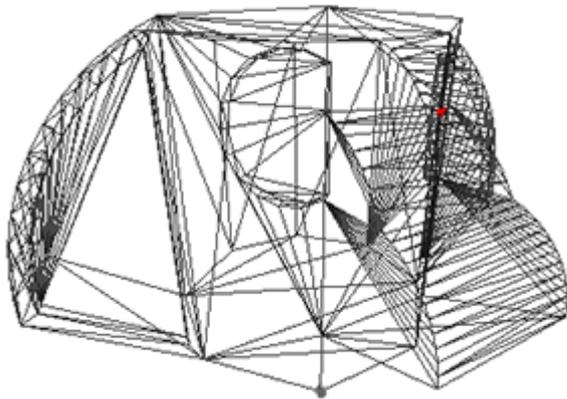
The 3D Object/Surface



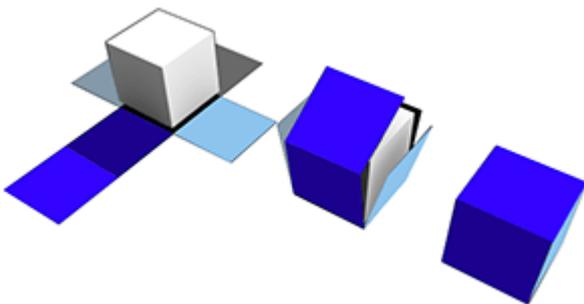
Just as the 3D model can be rotated and viewed from any angle, so the viewer can either move around the model, or the object can be rotated, and the media will 'stick to' and move precisely with the object.

Here is a demonstration model of a totem head with easy-to-identify surfaces.

The 3D Model

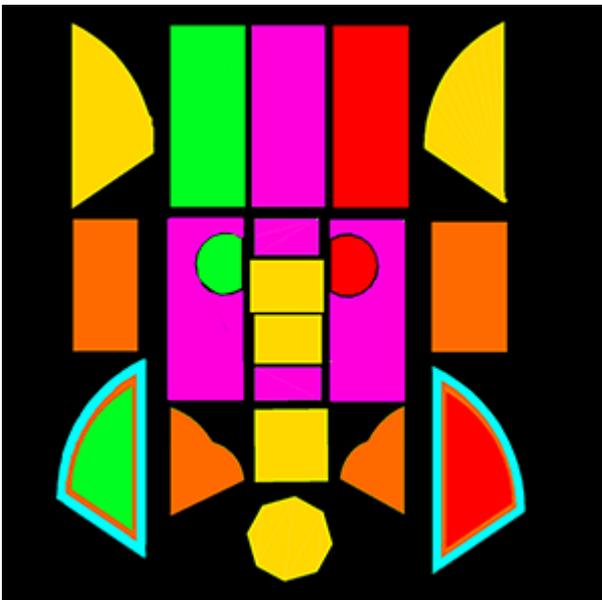


The 3D model is constructed in the same way, but its entire external surface is notionally 'peeled off' or unfolded onto a flat surface. This surface is 'painted' by the designer to create specially rendered 2D media. Whilst the object and model are described in $X \times Y \times Z$ axes, the surface is described in the 2D $U \times V$ axes, hence UV-rendered or mapped media.



In this illustration a simple cube demonstrates the theory of what can be actually a more complex mapping, as in the 'totem head' example used here.

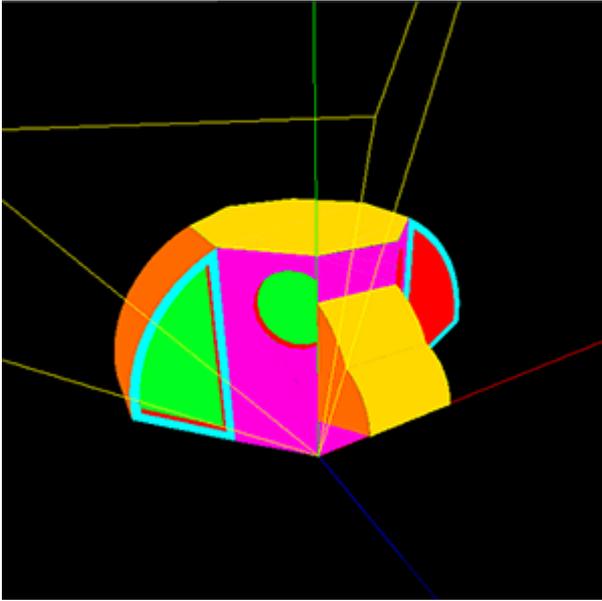
The 2D UV-Mapped Media



For the designer, all viewpoints now are perpendicular to the model/object surface, rather than, say an oblique view of the side of a cube. In textured mode, the characteristics of the mesh model surface are 'baked' into the flat media, which is then folded back onto the model in the projection.

Compare this UV-mapped media frame with the one used above for projection mode. The top row of faces are projected on the other side of the object, either by rotating the object into projector view, or by providing all-round channel coverage.

The reason this media does not look simply unfolded as in the cube above, is that this would create a lot of empty space in every frame. Instead, the elements are packed into the 2D space, with mapped coordinates that are used to place them on the target object or canvas.



This is how the UV media looks in the Playback Window when 'folded' or positioned back onto the model, as seen from a single channel. The yellow lines show the model origin (X,Y,Z = 0,0,0); red, green and blue are the X, Y, Z axes.

Mesh Mode Setup

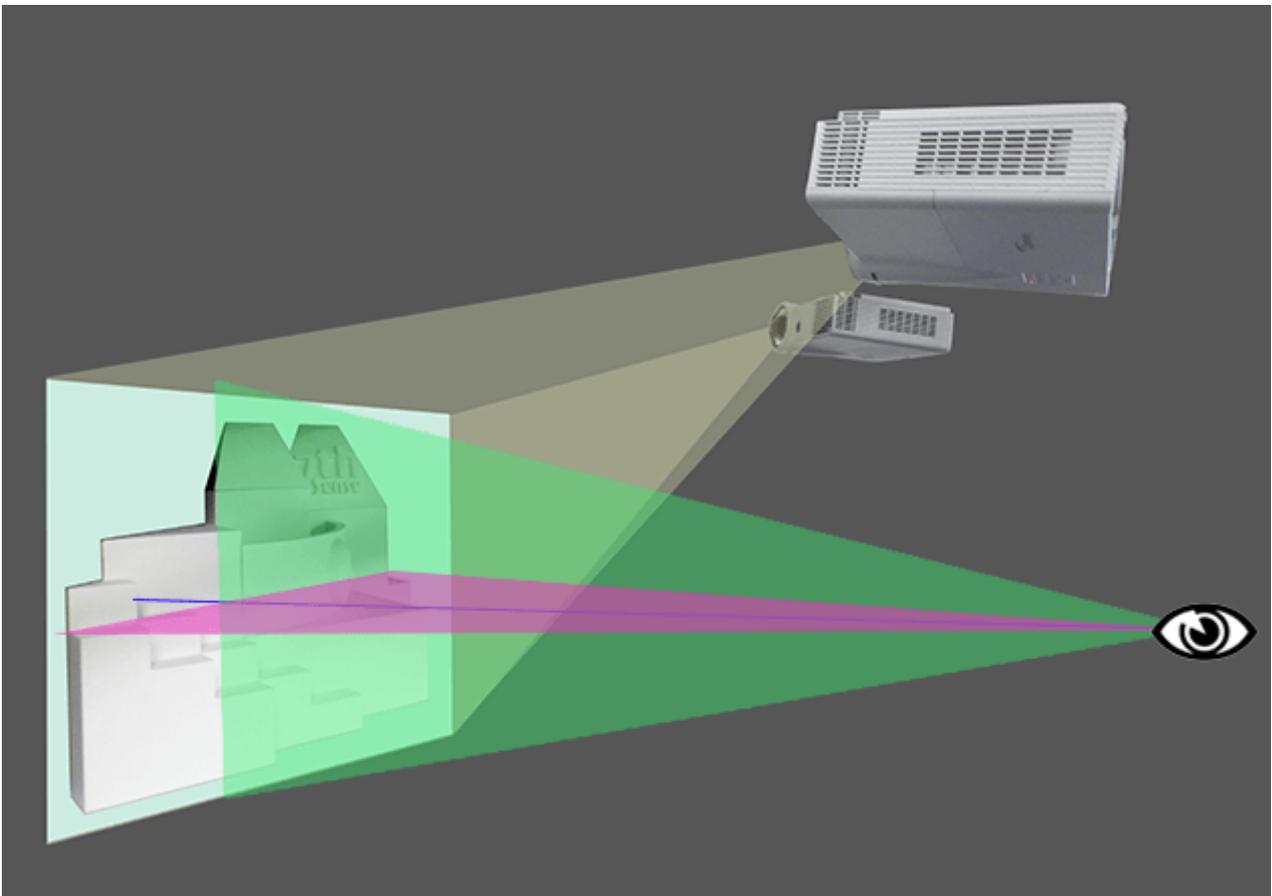
How Mesh Mode works

In mesh mode generally, there are a number of key elements:

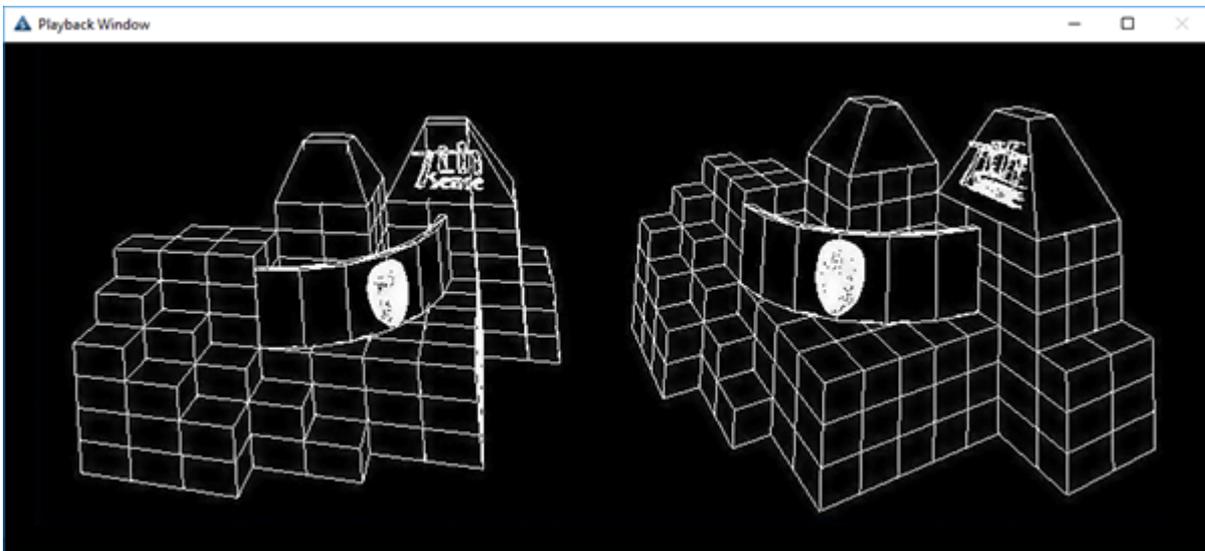
- Representing a real-world object as a 3D mesh of points (saved in a .obj file).
- For projection mode situations, designing flat media with a defined eyepoint.
- For texture mode situations, 'unwrapping' the 3D mesh model onto a 2D UV surface, and mapping the media onto it.
- Calibrating the virtual positioning of projectors in Delta, to their precise real-world positions.
- Using DeltaGUI to relate the software mesh model in the server, back to the real world object and projector positions.
- Using DeltaGUI to apply warping and blending for seamless joins and perfect fit.

Flat Projection Mode

Here is an example of a theatre for display in mesh mode, with a 3D canvas and two offset projectors:



This test model features flat and curved surfaces, detail in the 7thSense logo, and a face with a highly detailed mesh, in front. Whilst the projectors are pitched down, the design eyepoint (red and green axis planes) is horizontal. Here is how each virtual projector in DeltaGUI will see the canvas (in this case a 3D object), and note that they may see surfaces the viewer is not designed to see, or even for which there is no media coverage (Playback window, media disabled, grid displayed at 100%):



- In Delta, we must set up the render camera eyepoint of non-UV media, as defined in the media. This we do know exactly.
- We also need to know the projector locations for pitch, heading and field of view – but we can't know these exactly.
- By giving Delta enough reference points from the .obj file, placed in real space, Delta can work out how to warp the flat media from any angle, into exact position.

Already we can see that projecting non-UV-mapped media onto very oblique faces may reduce sharpness, and in this model it may even be better to have a separate mesh for the face on the front, with a separate head-on projector to avoid shadows. Delta can handle up to 16 meshes per show, and 32 channels.

Textured (UV-mapped) Mode

The theatre setup may be the same as above, but the UV-mapped media will be wrapped to the surface exactly the same way as the .obj model grid shown. Alignment is much easier, though the media construction is more complex.

Mesh Setup Basic Sequence

Assumptions

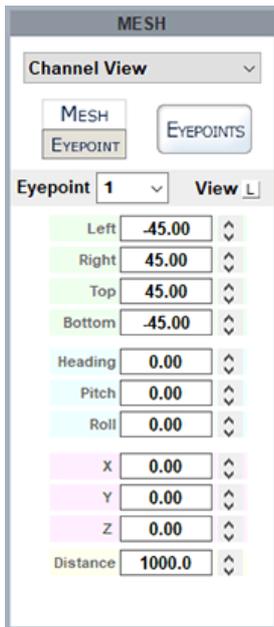
- You have a theatre set up with projectors and display canvas / objects.
- You have .obj mesh files that cover every part of your display.
- You have media that is either for flat projection, or is UV-mapped to the mesh, for textured mode.

Sequence in DeltaGUI

1. Check your canvas size and render settings : *Display > Mesh Mode > Mesh Canvas*
2. Set your active channels : *Display > Mesh Mode > Channel*, and
3. use the *Defaults Wizard* to distribute your virtual projectors.
4. Add your mesh files : *Display > Mesh Mode > Mesh > Mesh Files*
5. Set your parameters for each required eyepoint (not required for UV-mapped media in textured mode) : *Display > Mesh Mode > Mesh > Eyepoint Files > Eyepoint Details + Parameters* (see [Setting an Eyepoint in Delta](#)⁽¹²⁾), and
6. assign eyepoints to the meshes where they are to be used.
7. Set your eyepoint mode to 'Use Texture Coordinates' (not required for flat media in projected mode) : *Display > Mesh Mode > Mesh > Eyepoint Files > Eyepoint Details*
8. Use [3D Projector Calibration](#)⁽²⁰⁾ with any timeline media disabled and full grid display on, for **each channel** to select and align **each required mesh**, with the projectors and real-life canvas. For this stage, keep all channels active, but turn beams off except for the channel being aligned.
9. Apply (or enable) timeline media and view for where correction and adjustment is needed.
10. [Edit the mesh](#)⁽⁴⁵⁾, apply channel distortion, or media warps to correct for real-life canvas, and minor .obj or media errors.

Setting an Eyepoint in Delta

Eyepoints are always associated with meshes. UV-mapped media for textured mode effectively has an infinite number of eyepoints, but for projected mode, there will always be one precise eyepoint in the media. If you are not using UV-mapped media in textured mode, then you must obtain the eyepoint data from the content creator, or from within the media (e.g. Cinema 4D).



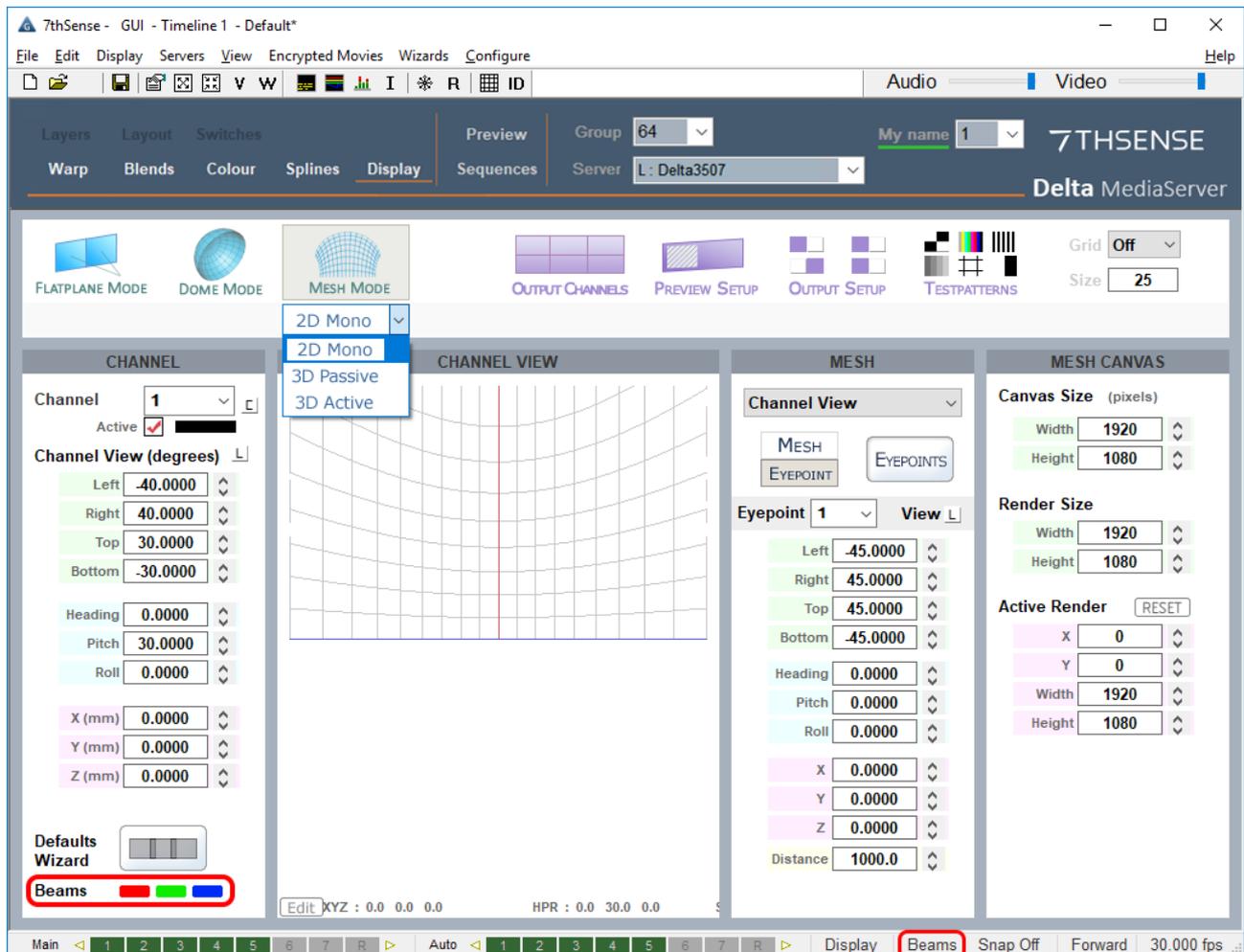
In Channel or Interactive view, click Eyepoint to view the Eyepoint parameters. Using the small L button, make sure the field of view (FOV) parameters are displayed as Left / Right / Top / Bottom (rather than lens aspect and zoom). Each set of parameters is just a translation of the other, but for translating render camera data from Cinema 4D, we need this view.

Enter the eyepoint data, as translated from C4D render camera details:

DeltaGUI eyepoint	Cinema 4D FOV
Left	(negative) of horizontal FOV × 0.5
Right	(positive) of horizontal FOV × 0.5
Top	(positive) of vertical FOV × 0.5
Bottom	(negative) of vertical FOV × 0.5
Heading	(negative) of C4D value
Pitch	(positive) of C4D value
Roll	(positive) of C4D value (most likely 0)
X	(positive) of C4D
Y	(positive) of C4D
Z	(negative) of C4D focal length

DeltaGUI Mesh Mode Controls

With DeltaServer running and the Playback Window in full screen, select Mesh mode in DeltaGUI. Choose your required 2D or 3D media mode from the Mesh Mode drop-down:



There are three control panels here:

Channel Panel: Setting up your Projectors

Here you set the projector positions as in any mode. The process of calibration will automatically make fine adjustment to these.

As for any mode, select the active channels you will be using, with the small 'C' button.

Note the two places where **channel beams** can be switched on and off: in the foot of the channels panel (click the colours), and bottom right in the DeltaGUI status bar. You will need to switch beams off for calibrating individual channels. Do not deactivate channels for this operation.

The **defaults wizard** dialog behaves the same as Dome Mode, but be aware you are typically arranging channels around an object at a centre, so use Projectors Outside Screen. Note that 'Include Cap' is not relevant to this mode.

Channel Size

Select from the two options. Use Existing Size (if the display model is the same as previous show) or Force Size and type in the settings according to the design guidelines.

Channels Used

Select the number of channels (projectors) in 'Force n Channels', or 'Use Existing Channels' if active channels already set up.

Screen

Set screen radius here if using 'Projectors at Screen Edge' in channel arrangement, or Use Existing Screen if using 'Projectors outside Screen'.

Spacing

If set to **Calculate**, projectors will be assumed to be all-round, in equal divisions of 360°. Changes to each projection can then be adjusted accordingly at a later stage.

Channel Arrangement

For Projectors at Screen Edge set a screen radius above; for Projectors outside Screen set a radius here.

At Radius

This is the visualised distance from the object origin.

Mesh Canvas Panel

Canvas Size

Your dimensions here for projected (not textured) mode will be defined by the media.

Render Size

This is for tuning performance by reducing resolution to decrease the load on the graphics card. (Preserve the aspect ratio.)

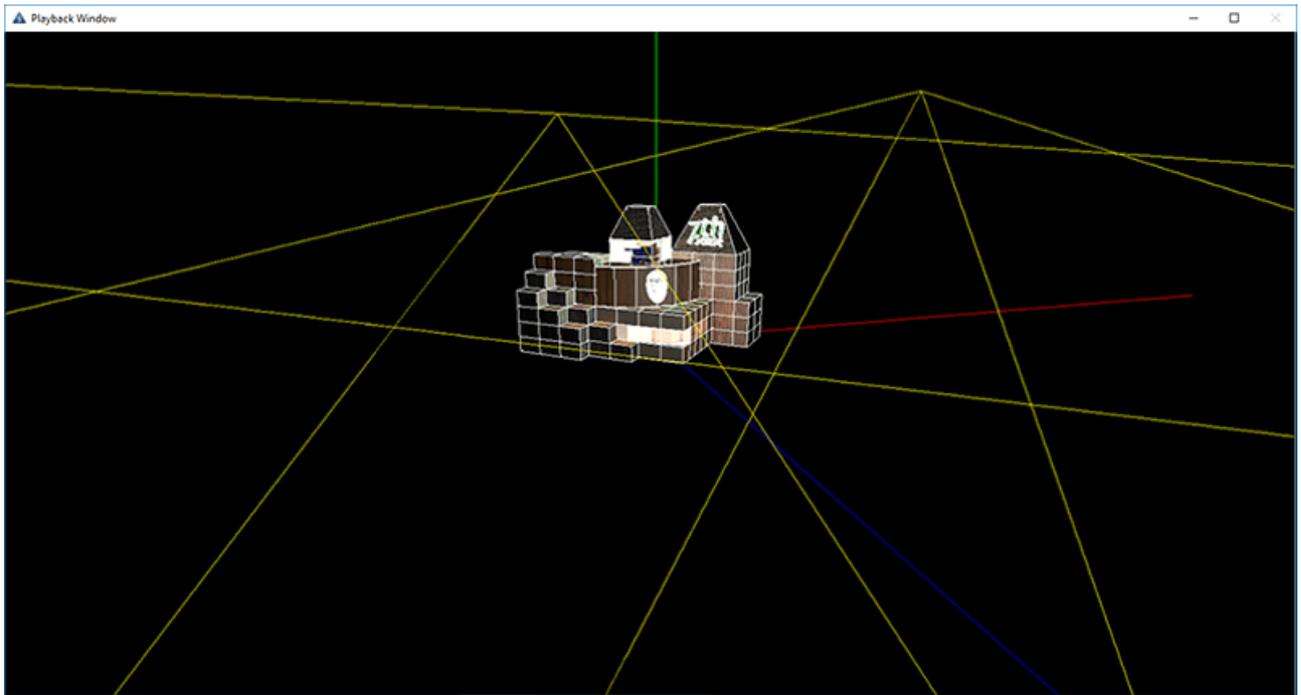
Active Render

This is the location of the render on the overall canvas.

Mesh Panel

Here, you have five different **views**:

- **Channel**: shows the view of the mesh surface covered by (as 'seen' by) a single channel in the GUI. It is here (or Interactive view) that you set a fixed eyepoint.
- **Interactive**: allows the whole model, axes and projector frustums (yellow lines) to be rotated and viewed in the playback window:



- [3D Projector Calibration](#)⁽²⁰⁾: place the mesh vertices on the actual canvas points
- [Mesh Editor](#)⁽⁴⁵⁾: alter the mesh by overlaying adjustments
- [Movement Nodes](#)⁽⁵²⁾: synchronise media movements between eyepoints and between meshes.

For projection mode (not textured) you must know the design eyepoint for the media, for each channel.

Mesh Dimensions

3D models are 'ideal', and even if used as the basis for constructing the actual real-world object (our canvas) there will be differences. DeltaGUI provides for all reasonable adjustments, but the mesh model must be accurate and to scale. Beyond this, the mesh itself may require editing.

- First of all, the eyepoint of the designer/viewer must be related to the **origin** (X, Y, Z = 0, 0, 0) of the mesh model.

Typically left as 0, 0, 0, you can reposition the object away from its inherent origin if necessary.

- If there is a difference in the **scale** of the model and the actual object, this must be adjusted.

Set to 1.0 by default. You can adjust the size of the model if the object dimensions are not fully to scale, but of course the projector locations may then have to be altered to account for the new object size.

- If there is any difference in **heading, pitch** or **roll** of the actual object from that expected by the projected media, these too can be adjusted.

Heading, Pitch and Roll values are in degrees. The 3D object may come in rotated about the wrong axis, so you can correct this here.

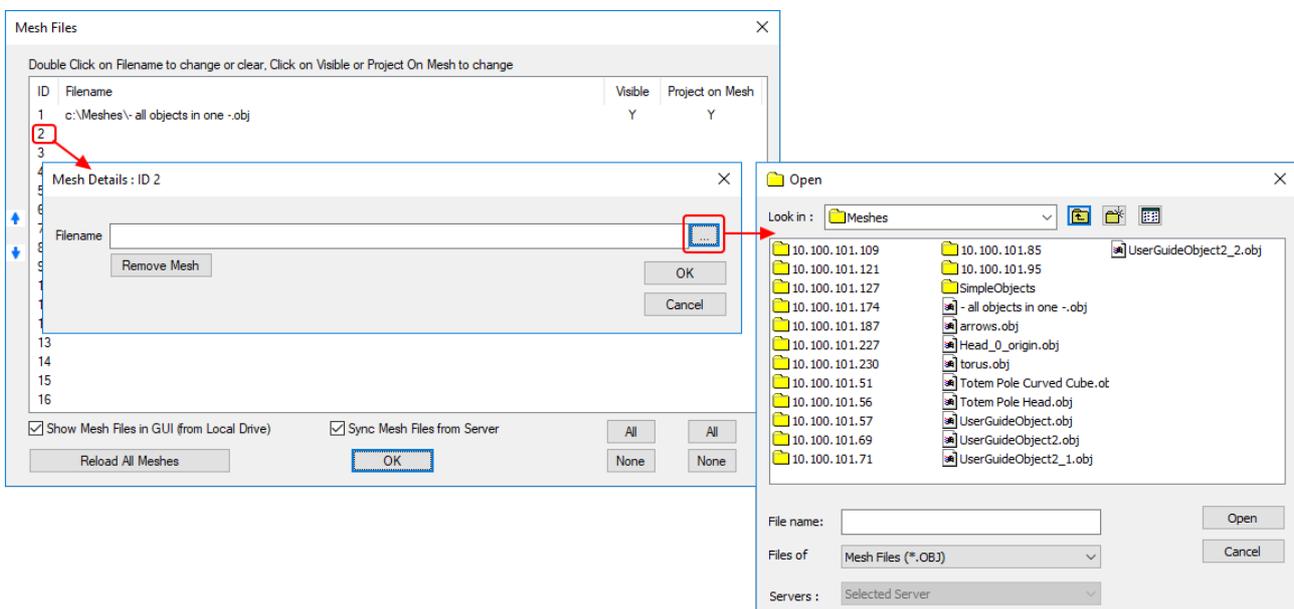
Mesh (.obj file) Selection

For DeltaGUI and DeltaServer to work together, each mesh **.obj** file must be on both the DeltaGUI PC and the server, in **C:\meshes**. When you assign a mesh in DeltaGUI, a copy of the server mesh will be synced to the DeltaGUI PC. Their filenames must stay the same in both locations.

In Channel view, select both mesh and eyepoint for each channel:



Double-click an ID line to add or to remove a mesh from a channel. You can apply multiple meshes per channel, up to 16 per show.



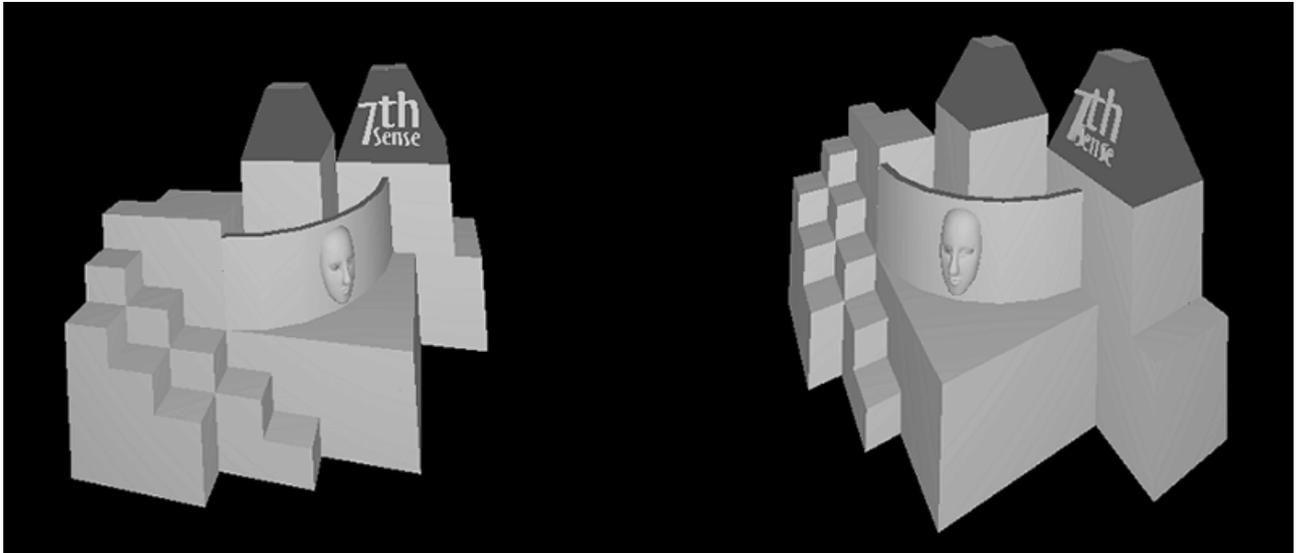
Note in this illustration that meshes synced from other servers have been acquired in folders with those servers' IP addresses.

Visible

A mesh can be assigned but only be visible in the GUI, not playback, by selecting No.

Project on Mesh

If you do not want media to appear, but you still want to be able to see and illuminate the object with the grid off, a mesh can be rendered in greyscale.



Eyepoint Selection

In this example (*not a textured mode example*), it is assumed that this eyepoint will apply to all meshes in all channels. Each mesh can have a different viewpoint or more than one viewpoint. Different meshes can share the same viewpoint. Eyepoints are available to all, or selected channels only.

Eyepoints ✕

ID	Enabled	Mode	Ortho	Depth	Alpha	Meshes	Outputs	Offsets
1	Y	Projected	n	Y	n	1 - 16	1 - 32	0.0 x 0.0
2	n							
3	n							
4	n							
5	n							
6	n							
7	n							
8	n							
9	n							
10	n							
11	n							
12	n							
13	n							
14	n							
15	n							
16	n							

Eyepoint Details : ID 1 ✕

Enabled
 Orthographic

Use Texture Coordinates
 Repeat Texture

Use Depth
 Enable Alpha
 Clamp To Border

Pixel Offset:

Note - If Repeat Texture is off, large offsets can result in black

Line Offset:

Enabled Meshes

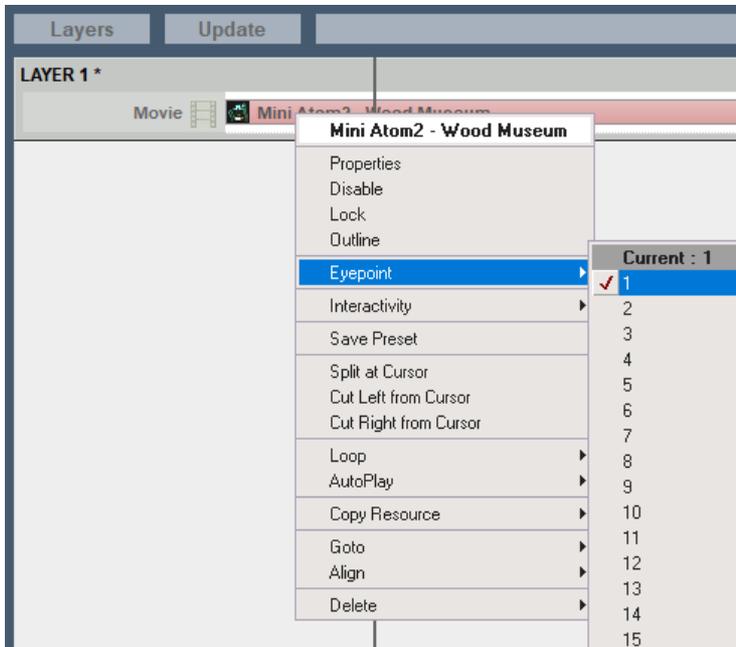
1 <input checked="" type="checkbox"/>	9 <input checked="" type="checkbox"/>
2 <input checked="" type="checkbox"/>	10 <input checked="" type="checkbox"/>
3 <input checked="" type="checkbox"/>	11 <input checked="" type="checkbox"/>
4 <input checked="" type="checkbox"/>	12 <input checked="" type="checkbox"/>
5 <input checked="" type="checkbox"/>	13 <input checked="" type="checkbox"/>
6 <input checked="" type="checkbox"/>	14 <input checked="" type="checkbox"/>
7 <input checked="" type="checkbox"/>	15 <input checked="" type="checkbox"/>
8 <input checked="" type="checkbox"/>	16 <input checked="" type="checkbox"/>

Enabled Output Channels

<input checked="" type="checkbox"/> 1	<input checked="" type="checkbox"/> 9	<input checked="" type="checkbox"/> 17	<input checked="" type="checkbox"/> 25
<input checked="" type="checkbox"/> 2	<input checked="" type="checkbox"/> 10	<input checked="" type="checkbox"/> 18	<input checked="" type="checkbox"/> 26
<input checked="" type="checkbox"/> 3	<input checked="" type="checkbox"/> 11	<input checked="" type="checkbox"/> 19	<input checked="" type="checkbox"/> 27
<input checked="" type="checkbox"/> 4	<input checked="" type="checkbox"/> 12	<input checked="" type="checkbox"/> 20	<input checked="" type="checkbox"/> 28
<input checked="" type="checkbox"/> 5	<input checked="" type="checkbox"/> 13	<input checked="" type="checkbox"/> 21	<input checked="" type="checkbox"/> 29
<input checked="" type="checkbox"/> 6	<input checked="" type="checkbox"/> 14	<input checked="" type="checkbox"/> 22	<input checked="" type="checkbox"/> 30
<input checked="" type="checkbox"/> 7	<input checked="" type="checkbox"/> 15	<input checked="" type="checkbox"/> 23	<input checked="" type="checkbox"/> 31
<input checked="" type="checkbox"/> 8	<input checked="" type="checkbox"/> 16	<input checked="" type="checkbox"/> 24	<input checked="" type="checkbox"/> 32

Using texture coordinates with UV-mapped media means that the model, grid, media and object can all be aligned in one calibration operation.

Eyepoints can also be assigned to media by right-clicking the resource on the timeline:



Enabled:

Enable or disable this eyepoint. Disabling will hide related meshes.

Use Texture Coordinates:

Enable this if the .obj file contains texture coordinates, if you enable this and the model has no texture coordinates, nothing will show on this mesh. Disable this, and the flat plane canvas will be projected from an eyepoint configured in the GUI.

Use Depth:

Enable (default) to have occlusion between meshes, disable and the meshes will not occlude against each other (determined by the order shown in the mesh files dialog).

Orthographic:

Choose this to have an orthographic (non-lens based) eyepoint, note this only applies to a Projected (non-textured) eyepoint.

Enabled Meshes:

Choose the meshes related to this eyepoint.

Enabled Output channels:

Choose which output display channels will render this eyepoint.

3D Projector Calibration

Now that the physical projectors are set up correctly for the media and model eyepoint, the model can be calibrated to the real-life object or canvas. This is done by projecting the mesh using the grid, with its edges and vertices, rather than the media, onto the object. Each channel is calibrated separately. The media is the same, the mesh is the same, and Delta does all the maths required to adjust for each projector position.

Per Channel Setup

With the Playback Window in full-screen mode, and DeltaGUI in *Display > Mesh*, with the mesh files set up as above, turn the grid on to 100%. (Grid size: the spacing of the .obj mesh depends on the file itself.) The grid will not show until a channel is calibrated.

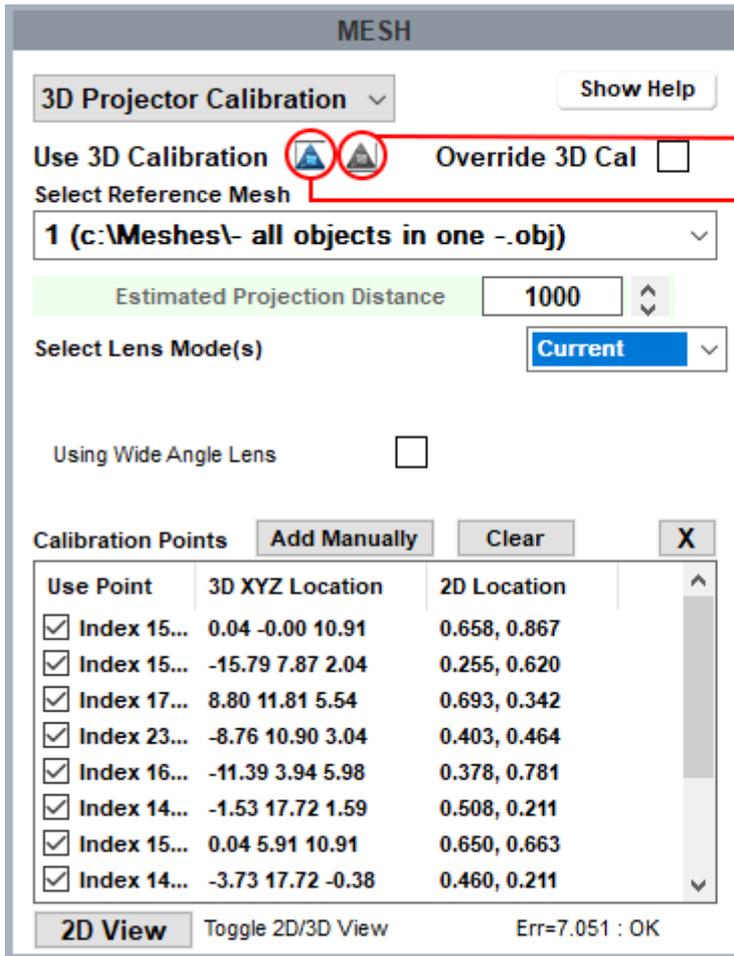
If any warps or blends are on the timeline, you will see a red warning, and a checkbox to disable them. This is because the undistorted mesh should normally be added to the real-world canvas, and any warps and blends applied to the media once calibration is complete.

However, since the 3D Calibration process can interpret radial lens distortion from the resulting calibration points, this can now be automatically retrospectively applied to a timeline background flat geometry, for that channel. This can be further warped manually.

- Turn the beams off all but the channel you are calibrating.
- In the Mesh panel, select 3D Projector Calibration view, click the left triangle button (changes from grey to blue) to select 'Use 3D Calibration', and select or ensure your reference mesh (.obj file) for this channel is correctly showing.
- Enter an estimated projector distance for this channel.
- Select Lens Mode.

Lens Mode

Current is the preferred setting for Delta 2.6 users, and further improves reliability. For users upgrading from Delta 2.5 to 2.6, calibrations previously made in **Manual** lens mode will appear as '**Legacy_1**', and those made in **Default** lens mode, as '**Legacy_2**'. These two modes can continue to be used to modify the calibrations. For Legacy_1 and Legacy_2, see the [Delta 2.5 User Guide](#).



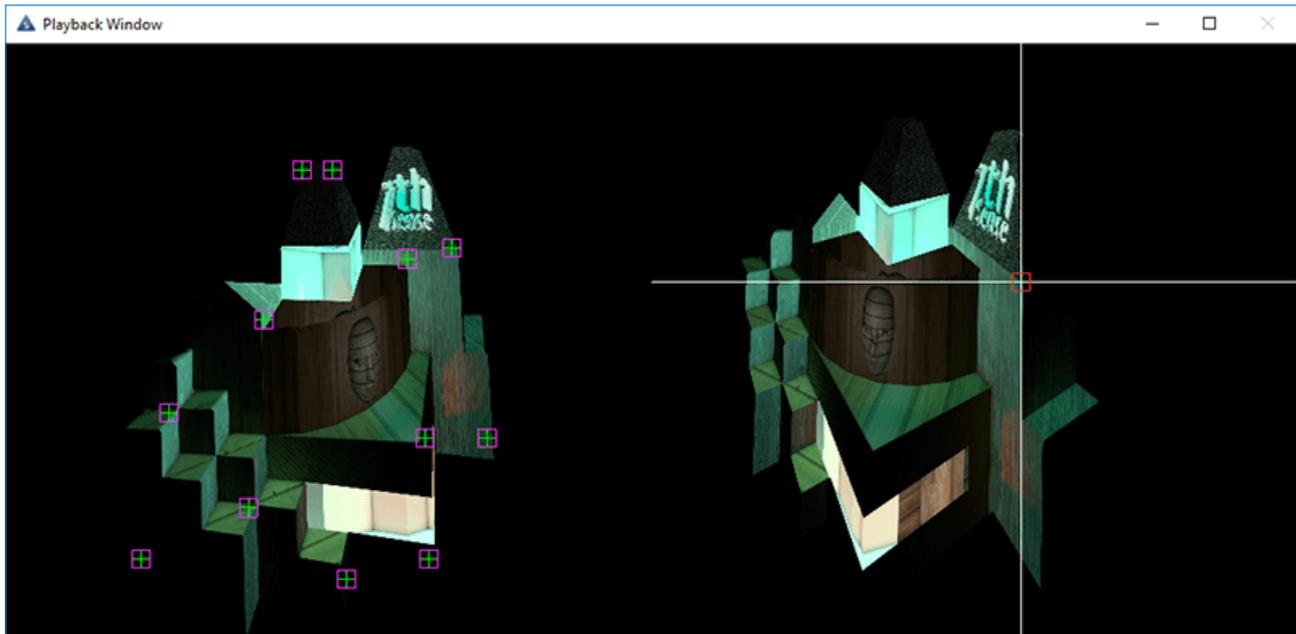
Apply lens distortion

Use 3D Cal

2D/3D View

This is the button you will use to toggle between selecting a vertex from the model and placing it precisely on the object. 2D view is looking at the model from the projector position, 3D shows the reference mesh object.

In this playback window view, channel 1 (left) has 'Show all 2D points' applied. Channel 2 (right) is the playback view when selecting an object node in DeltaGUI in '3D View'.



Calibration Points

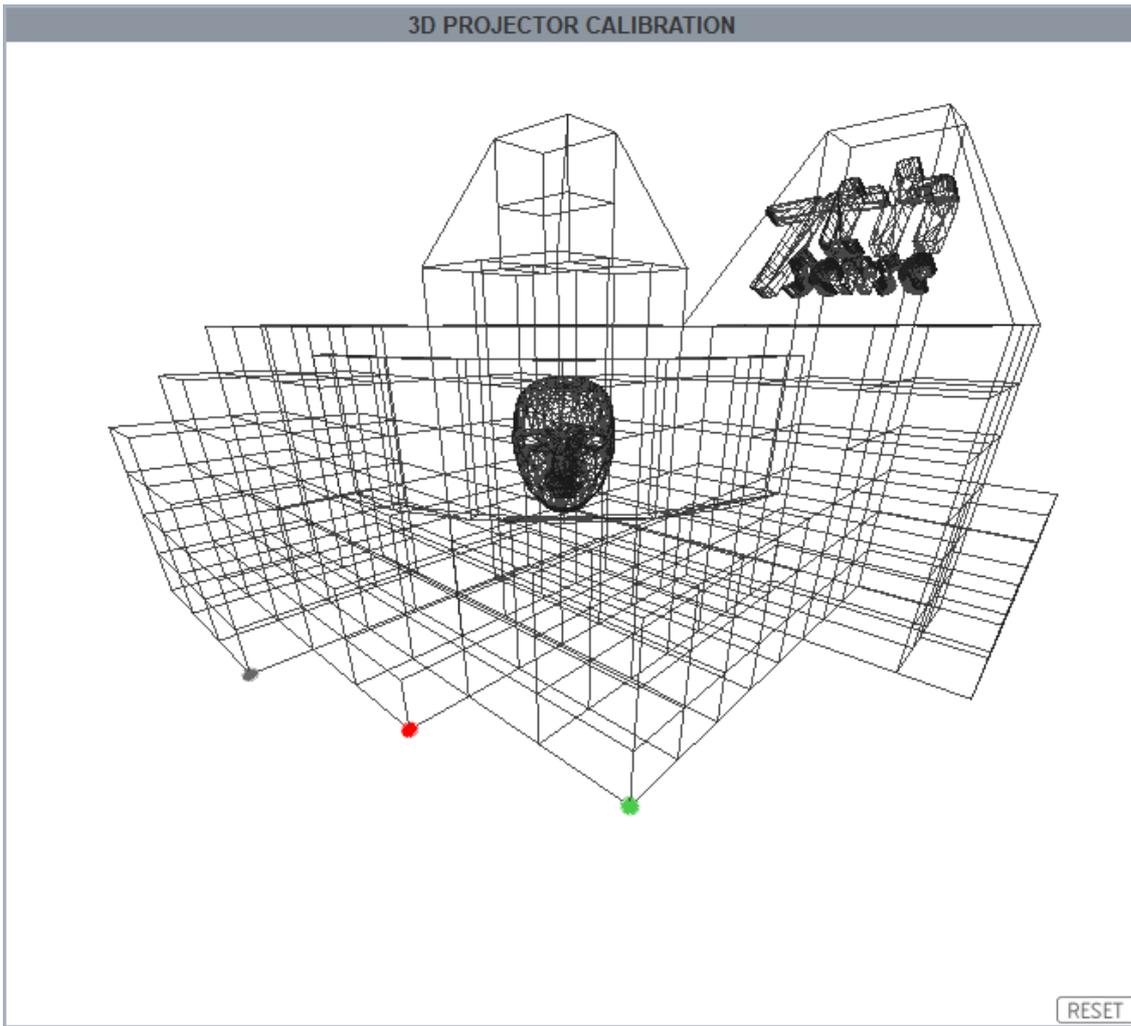
Calibration requires at least 6 distinctive and distributed reference points (mesh vertices) to be accurately positioned, using DeltaGUI, on the object. Delta will infer the relative position of all other vertices, which is why a good 3-dimensional spread of calibration points is needed. Delta will then automatically calculate the mapping of the whole .obj mesh onto the object. This initial calibration may be sufficient, but further reference points can be selected and adjusted thereafter.

In 2D view, position the mesh model (this may be out of view, but the RESET button will recentre it). Manipulation of the model using the mouse buttons is the same as shown in Interactive view (shown right). This takes practice, but will be useful in being able to quickly rotate the model for a clear view of the vertices you want to use (rather than a node right behind it).

Mouse controls:

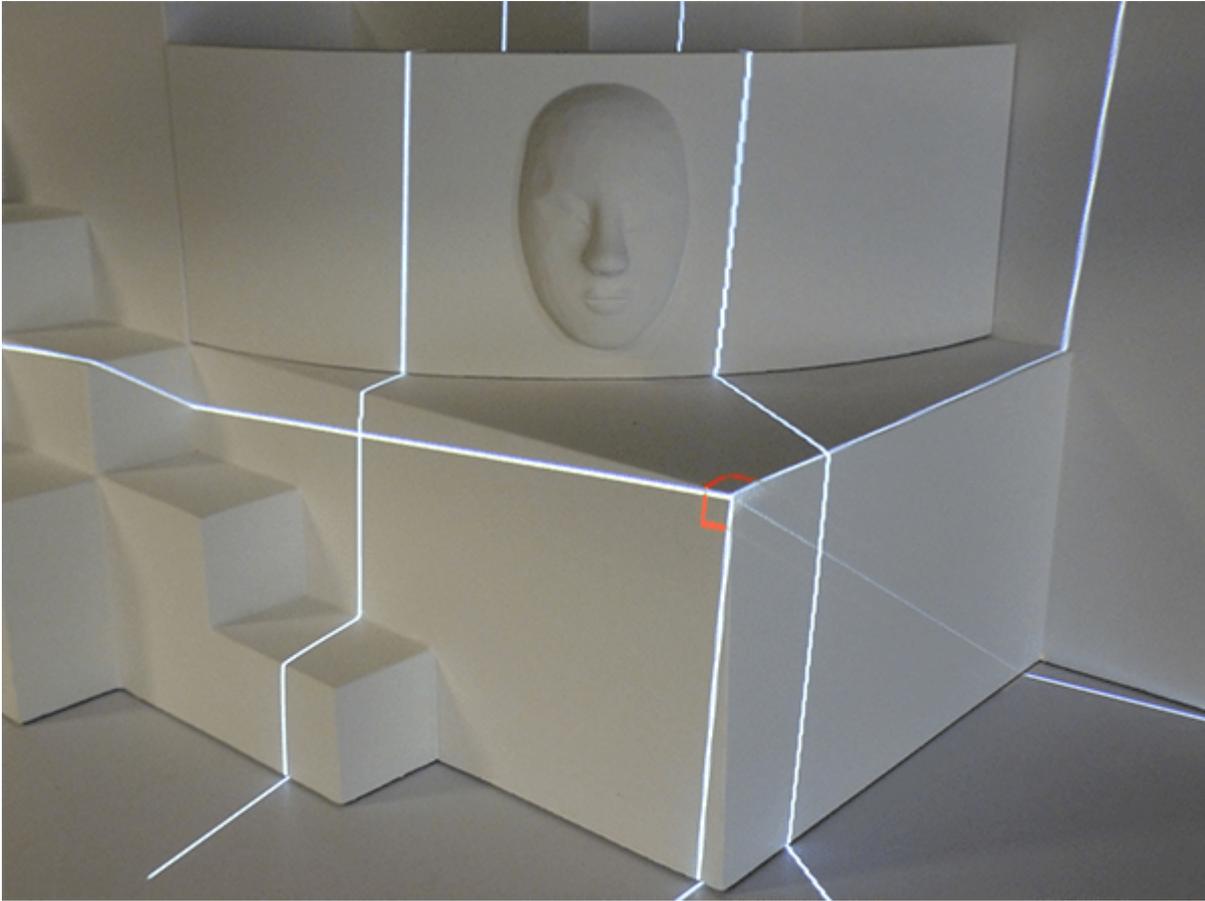
- **Left-click + drag** : pan and tilt the whole mesh
- **Right-click + drag** : shift the whole mesh
- **Centre-click + drag** : zoom
- **Shift+left + drag** : spin the mesh around a set point in 3D space

Hint: in [Mesh Editor](#)⁴⁵ [view](#)⁴⁵, you can change the size of the dots marking vertices with Node Size . The movement of the mouse can also be adjusted, to control the responsiveness of the mesh image.



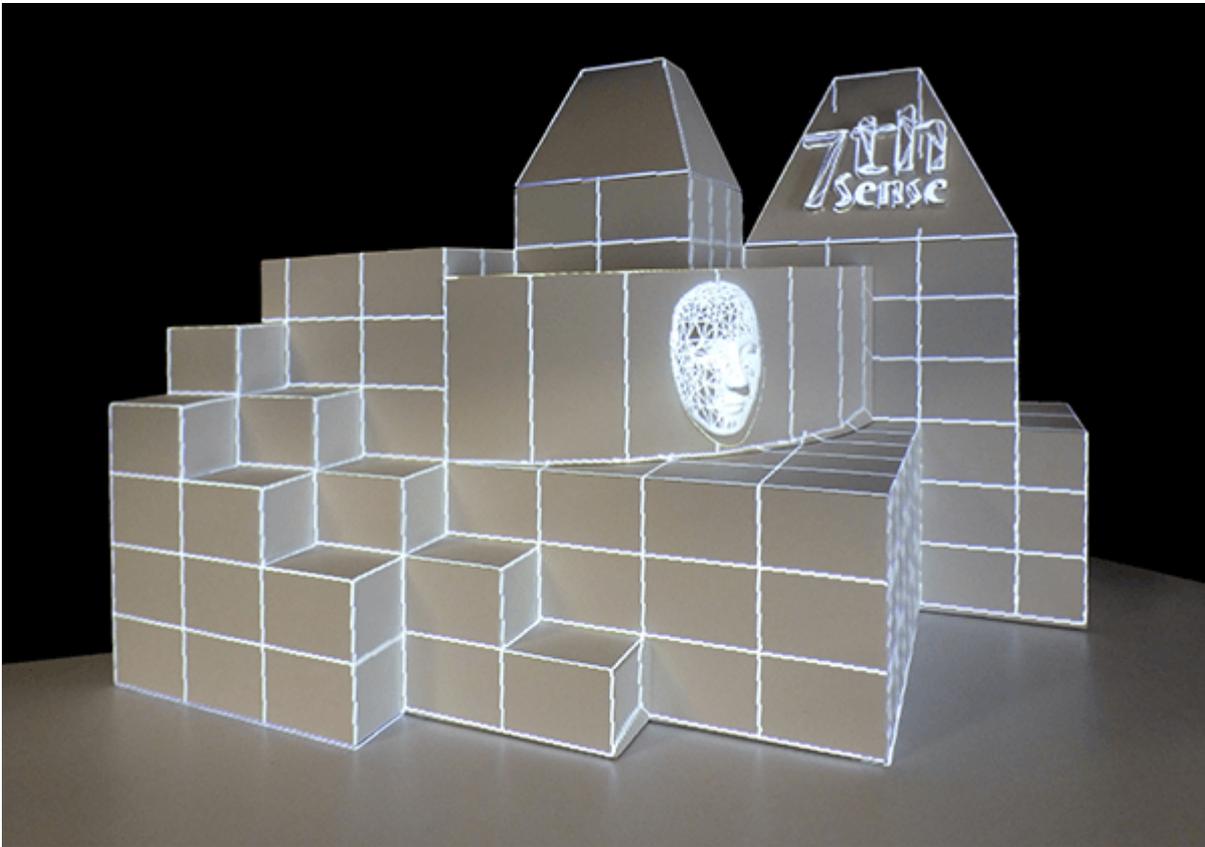
When a node point is found, the dot is grey. When a vertex is selected, it is red. When it has been set, it shows green.

Select a vertex, and switch to 3D view. Now look at the object/canvas and just move the mouse (no buttons) to drag the white cross-hairs in a red box, over the same point in the object:

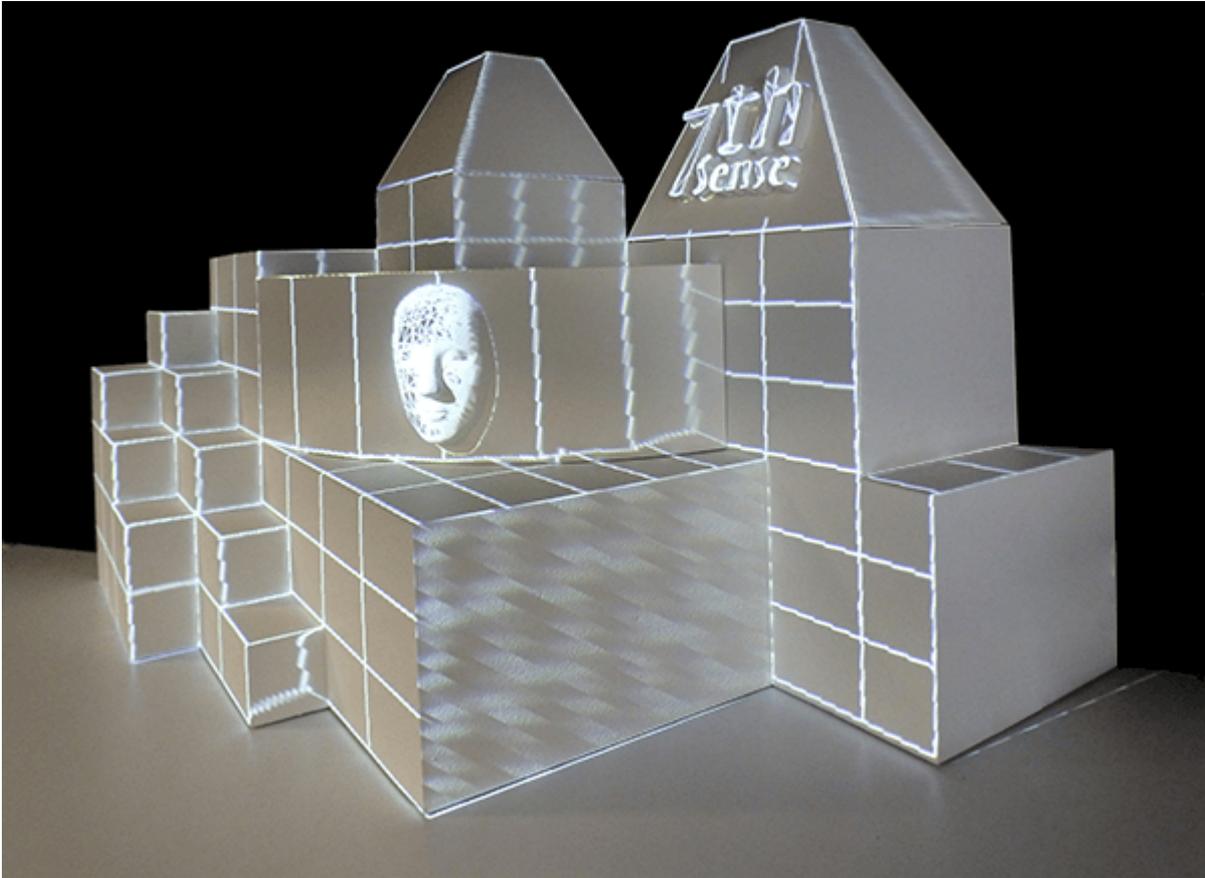


Press **Ctrl + left-click** to register this point, and the lines will turn green. This point will now be listed in DeltaGUI with its 3D XYZ model coordinates and 2D channel-frustum location. **The mouse position need only be approximate:** after registering the point, you can nudge it into more exact position with **Ctrl + arrows** or **Ctrl+Shift+arrows** (fine control). You can choose later not to use it, if for some reason it is not being useful, or you can readjust it by selecting from the list in 3D mode (it will be shown on the object) and nudging it again.

When you have 6 registered points, DeltaGUI will calculate the whole mesh, and the whole grid will be displayed on the object. It should be a reasonably good fit. There are two obvious points here that can be selected to improve the result:

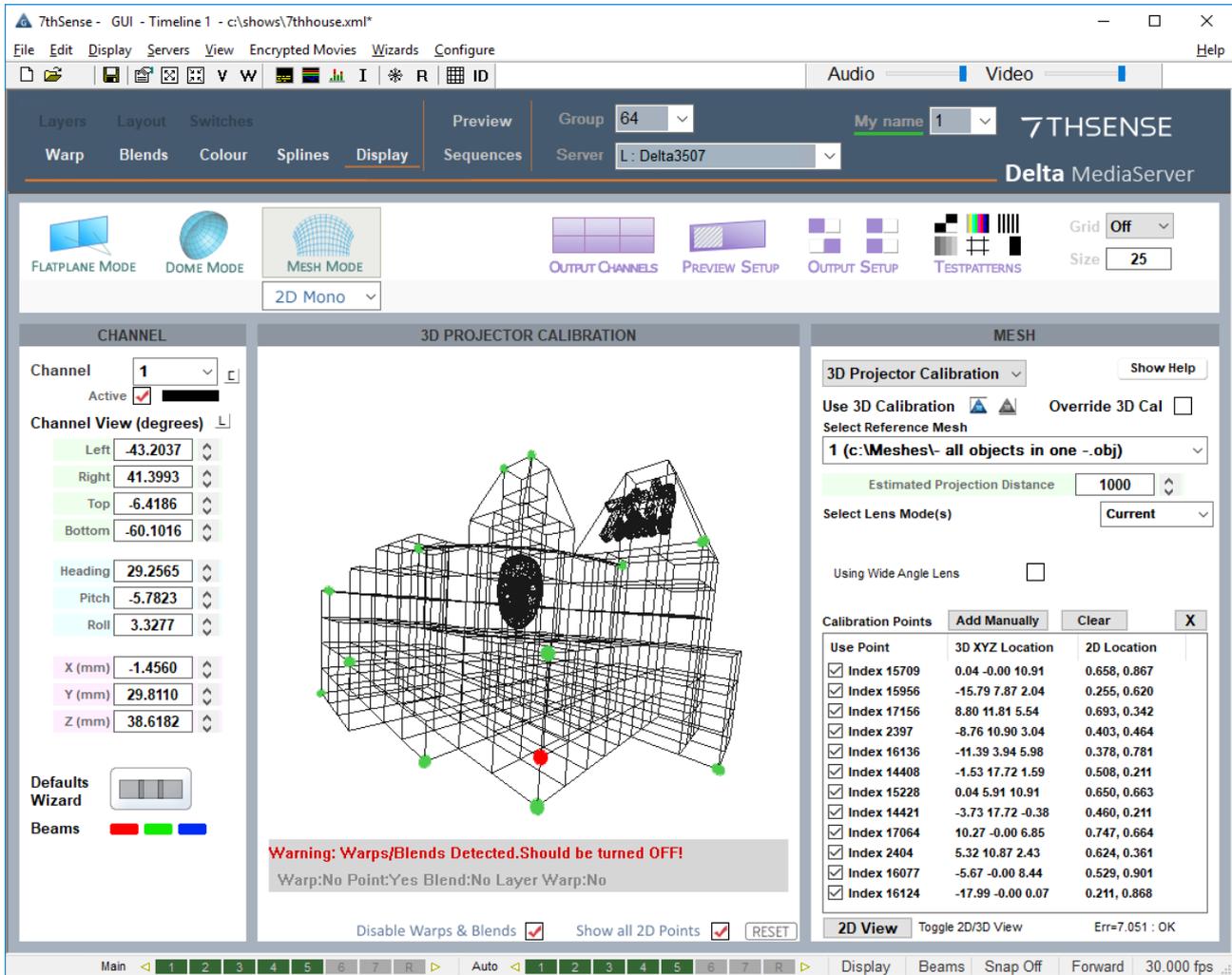


Now look around the other side, which will be covered by the second projector, and we see that it might be better not to include part of the mesh here at all:



We can calibrate the second projector now, and return to the [mesh editor](#)⁴⁵ later to tidy up, and to hide or fade these faces per channel.

Here is one completed channel in DeltaGUI:



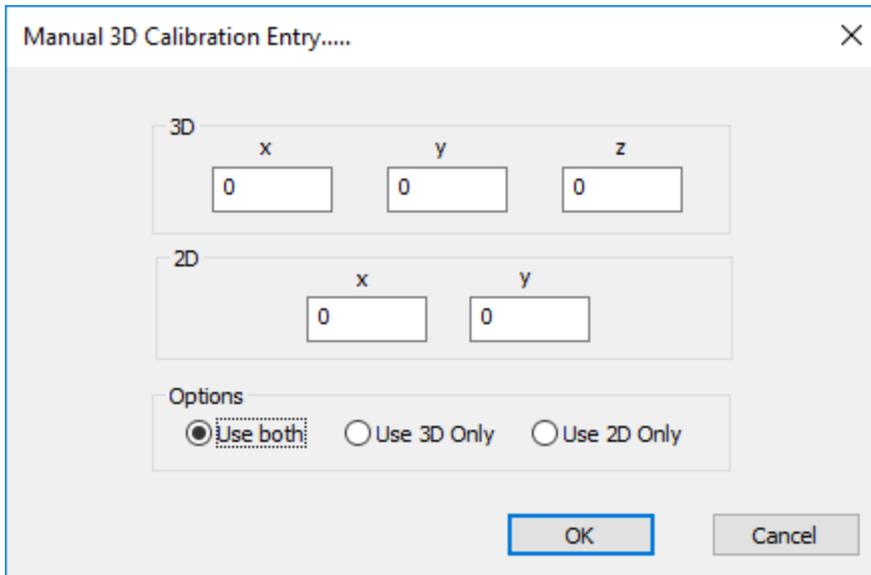
Editing Calibration Points

To **adjust** any point, select it in the list (make sure you don't deselect it in the checkbox) in 3D mode, and it will show a red box around the vertex on the object. Nudge it into a better position with **Ctrl+arrows** or **Ctrl+Shift+arrows** (fine control).

Clear: If you want to start over, click 'Clear'.

: Delete a selected point only.

Add Manually: If you have precise calibration points available from tracking software such as BlackTrax, these can be added manually:



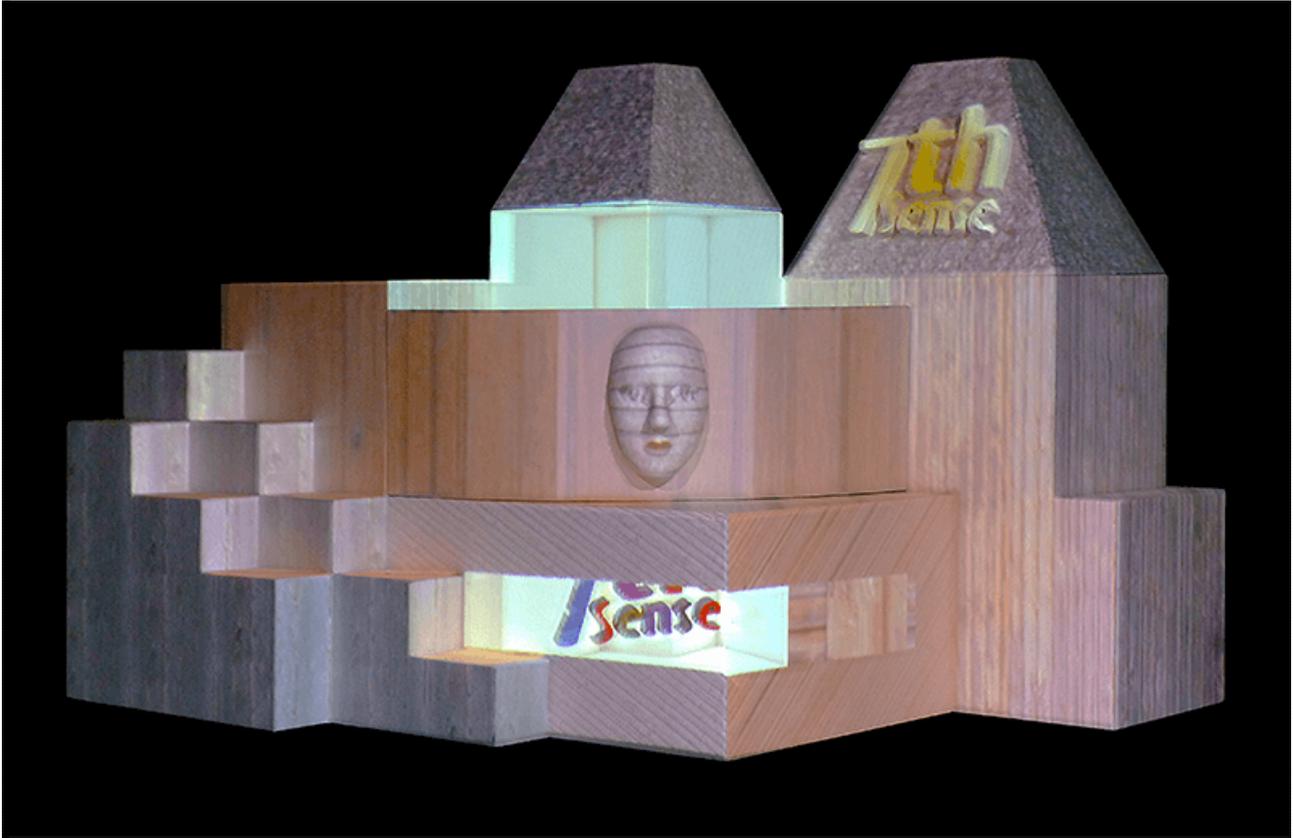
The image shows a dialog box titled "Manual 3D Calibration Entry....." with a close button (X) in the top right corner. The dialog is divided into three sections:

- 3D:** A section with three input fields labeled "x", "y", and "z", each containing the number "0".
- 2D:** A section with two input fields labeled "x" and "y", each containing the number "0".
- Options:** A section with three radio buttons: "Use both" (which is selected), "Use 3D Only", and "Use 2D Only".

At the bottom of the dialog, there are two buttons: "OK" and "Cancel".

Viewing the Result

Now take the grid display off, and go back to DeltaGUI > *Layers* and add or enable your media in the Timeline, and your media should fit the object. UV-mapped media is baked into the .obj model, but flat-projected media may take additional adjustment for a perfect fit.

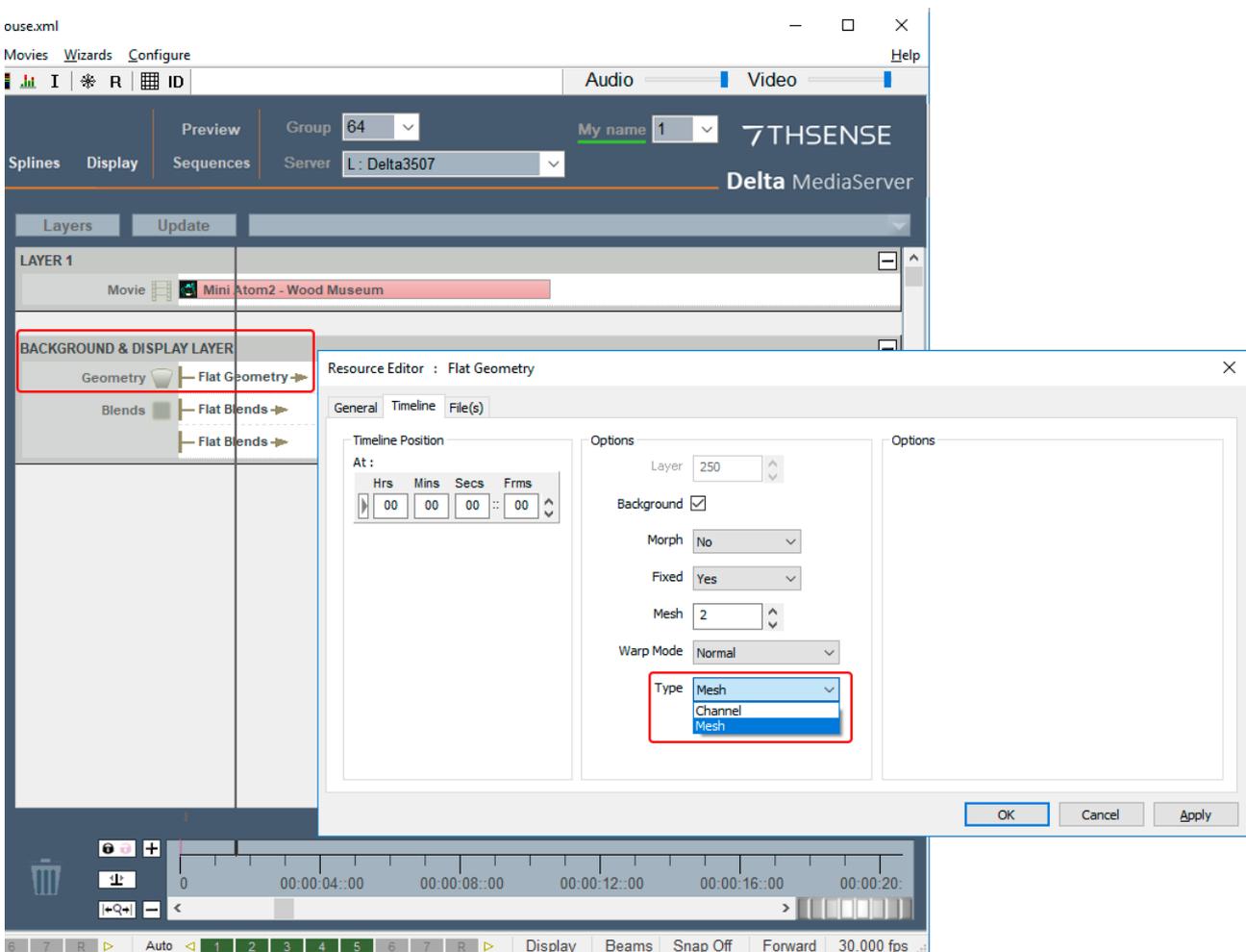


Warping a Mesh

It may be useful to warp the **mesh as a whole**, rather than manipulating the mesh .obj or the canvas, especially if it just involves regular or geometric deformation. Timeline media warps are normally disabled during calibration, but can be added and adjusted afterwards.

From Delta 2.6.48

The warp must be applied to the background layer, and can be applied to a channel or to a mesh. Multiple background warps can be applied to separate meshes and channels. (Selecting the type changes the number selector descriptor.)



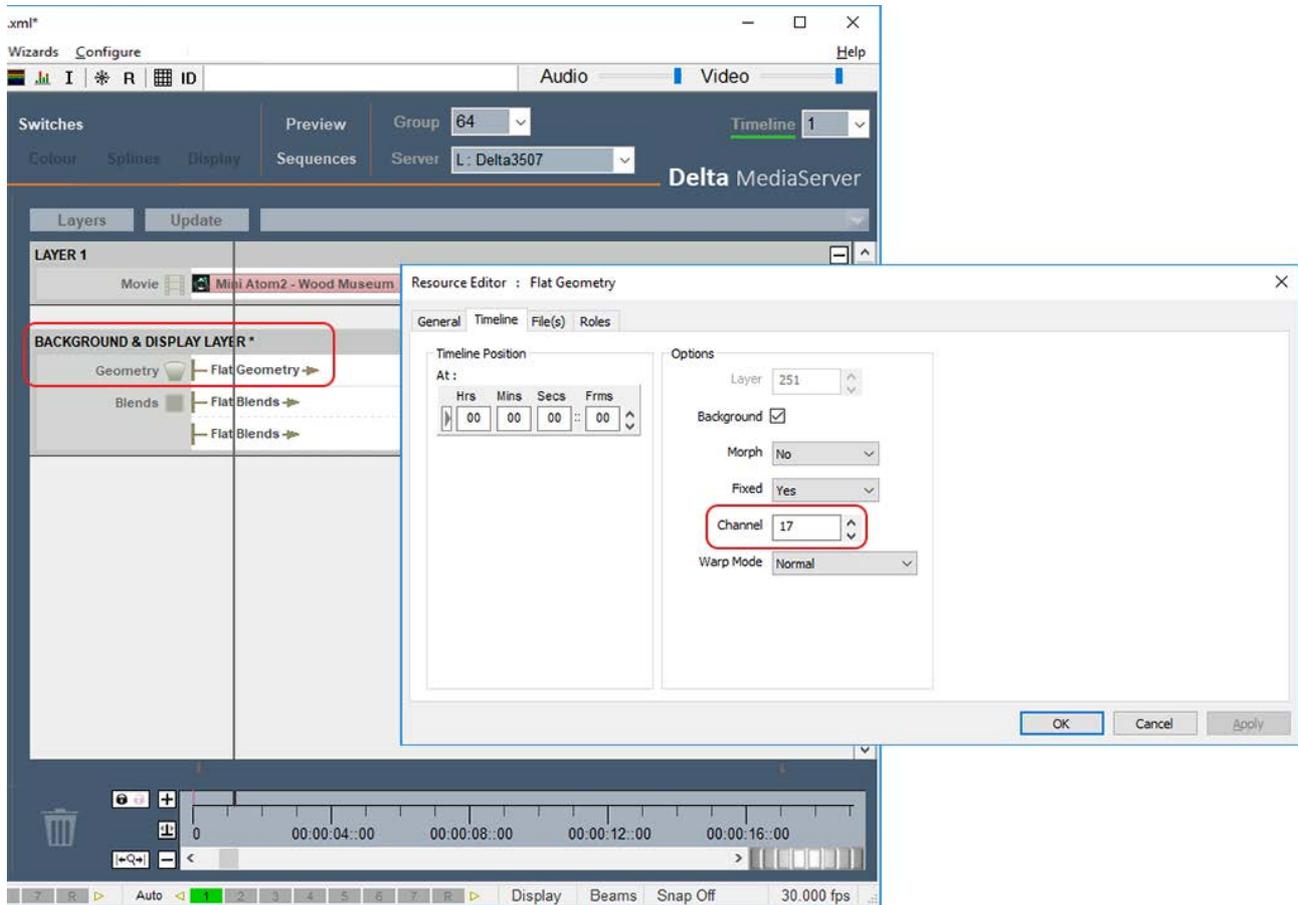
Earlier versions: Delta 2.6.1 to 2.6.47

The warp must be applied to the background layer, and not the media channel but the respective mesh. Delta will allow one or two meshes to have background warps applied:

- to apply a warp to the first mesh (ID=1), assign it to channel 17

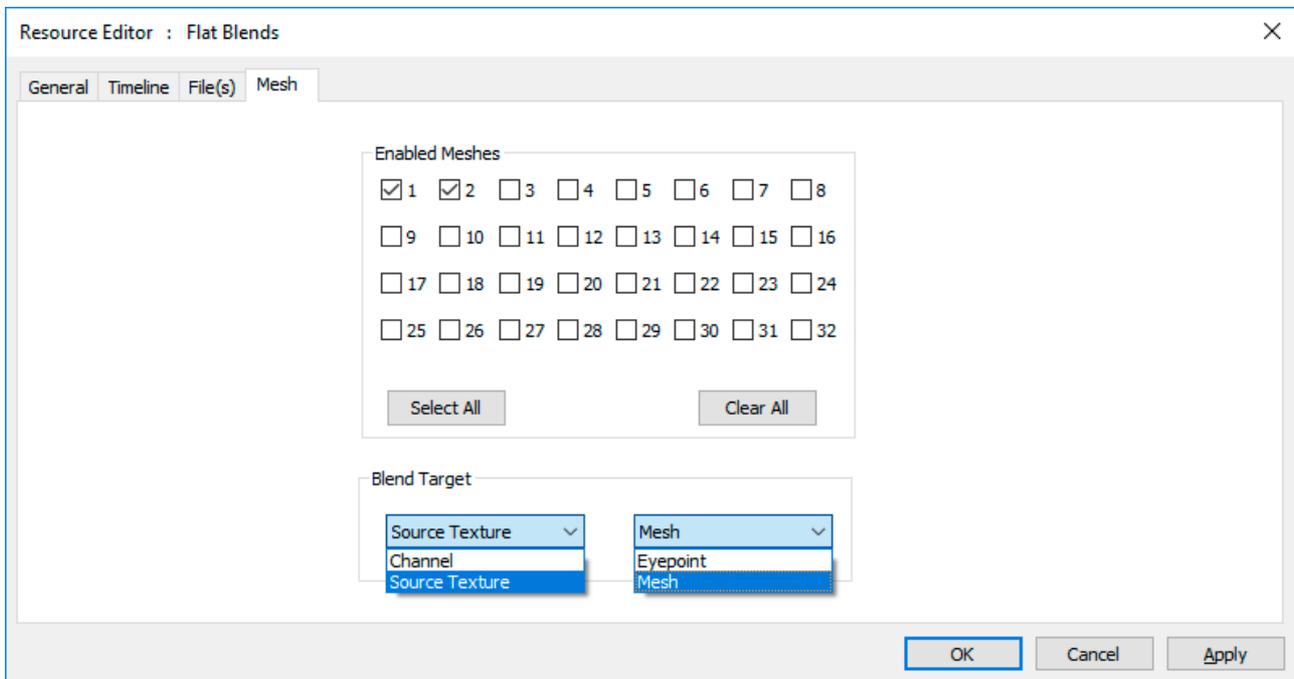
- to apply a warp to the second mesh (ID=2), assign it to channel 18

Note: these are not 'active' channels.



Blending in Mesh Mode

In mesh mode, flat blends offer an additional choice. The Resource Editor for flat blends placed on the timeline (background layer) have this additional tab of settings:



Blend Target

In projected mode, the blend can be restricted to not only channel (also in the Timeline tab) but particular eyepoints, and with the choice of where it applies. 'Channel' will apply it as normal to the periphery of the projector frustum, 'Source Texture' will apply it to the periphery of the media frame.

In textured mode the blend target should be 'Channel'.

Enable target mesh(es) and Eyepoint(s) as required.

This is similar to being able to warp the canvas, or to apply warps to media.

Troubleshooting Projection Mode

Fitting the Mesh

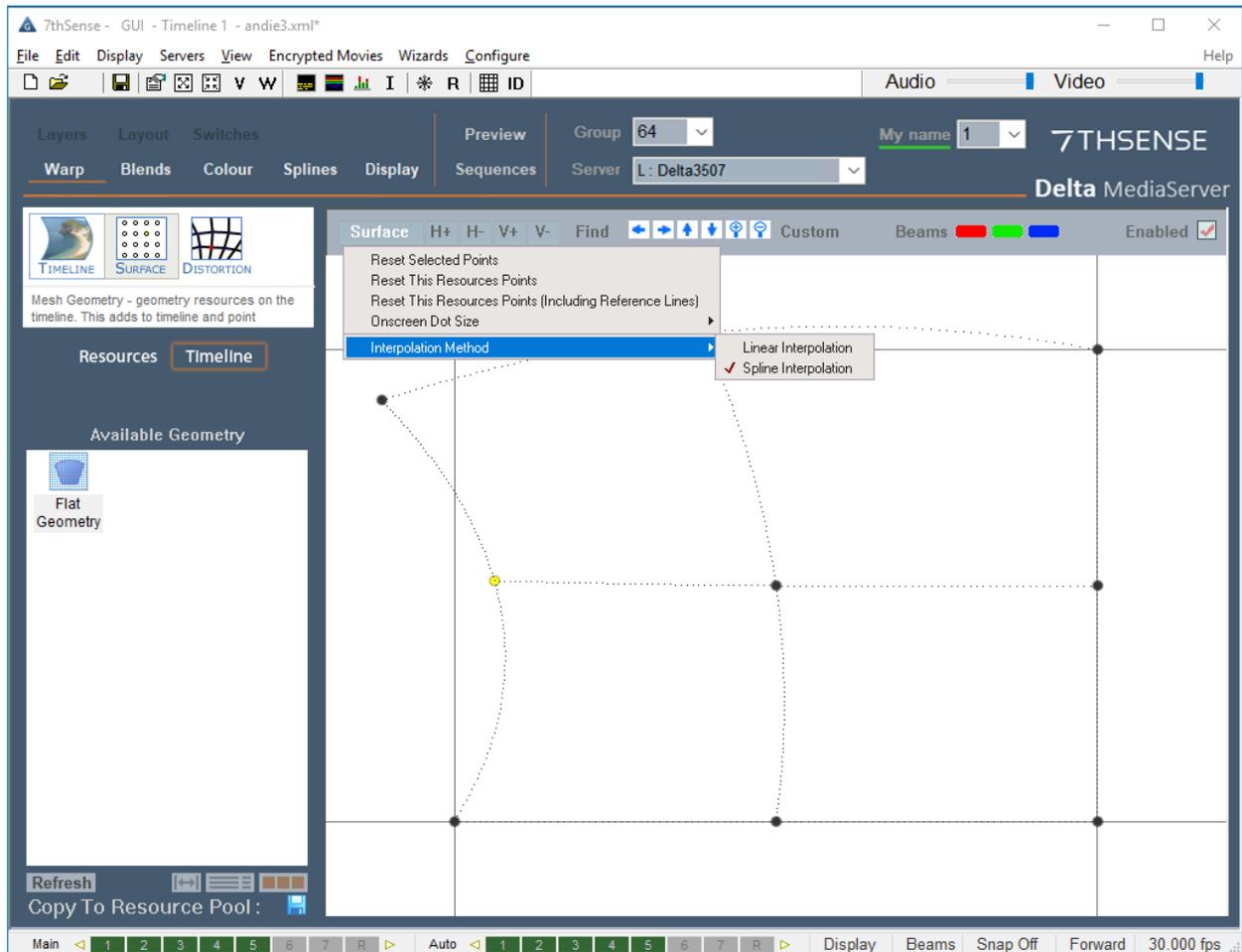
If, having positioned six calibration points, the whole projected grid is very wrong, this could be either due to poor distribution of points or a mistaken vertex between the mesh and real life (perhaps behind the one you think it is).

- Try switching off individual calibration points in case one in particular is bad.
- Try adding one further point, perhaps where the projected grid is most out of position.
- Check the Lens Modes in the 3D Projector Calibration panel. Are you using the recommended defaults? Try unchecking and checking Fixed Aspect, Zero Tangential, and K6, K5, K4 for best improvement.
- Discrepancies between the mesh and the real life surface can also be corrected for by [editing points on the mesh](#) ⁽⁴⁵⁾.

Fitting the Media

Delta works out how the projectors fit the model to reality. If the media designer has worked from the same model and scale, and rendered it correctly, *and* you have compensated for irregularities in the real-life canvas, then the single crucial error may be setting the eyepoint position in Delta, so check this carefully first.

- If the eyepoint is correct, some warping of the canvas may be appropriate. Add a flat geometry to the background layer, assign the channel in the Resource Editor, and use channel point distortion warp (see this section under Display Configuration, in the Delta User Guide).
- If the overall correction required seems geometric, try warping the mesh as a whole (see [Warping in Mesh Mode](#) ⁽³⁰⁾).
- If the mesh and media correspond and the required adjustment is small, then the **media** can also be surface warped into a better fit. Place a flat geometry on the timeline in the same layer as your media. Go to Warp and ease the media into place. Remember that Surface Warp has two different interpolation methods. Spline method moves points in smooth arcs, warping a greater area than the linear method, which moves the point(s) in straight lines, within its own area, without affecting the rest of the image. The H+, H-, V+ and V- buttons increase the grid density (click then 'Apply' or Enter), which affects the area of warping.



For more guidance on surface warp, see this section under Display Configuration, in the Delta User Guide.

Mesh is Clipped or not Showing on Playback

If on playback (projected, displayed or in the Playback window) you either cannot see all your mesh, or cannot see it at all, it could be out of range of the DeltaServer clipping plane settings. This could occur for very large or distant meshes, which you would not notice in DeltaGUI whose clipping planes are separate.

The DeltaServer far clipping plane is held in a Registry key and can be adjusted.

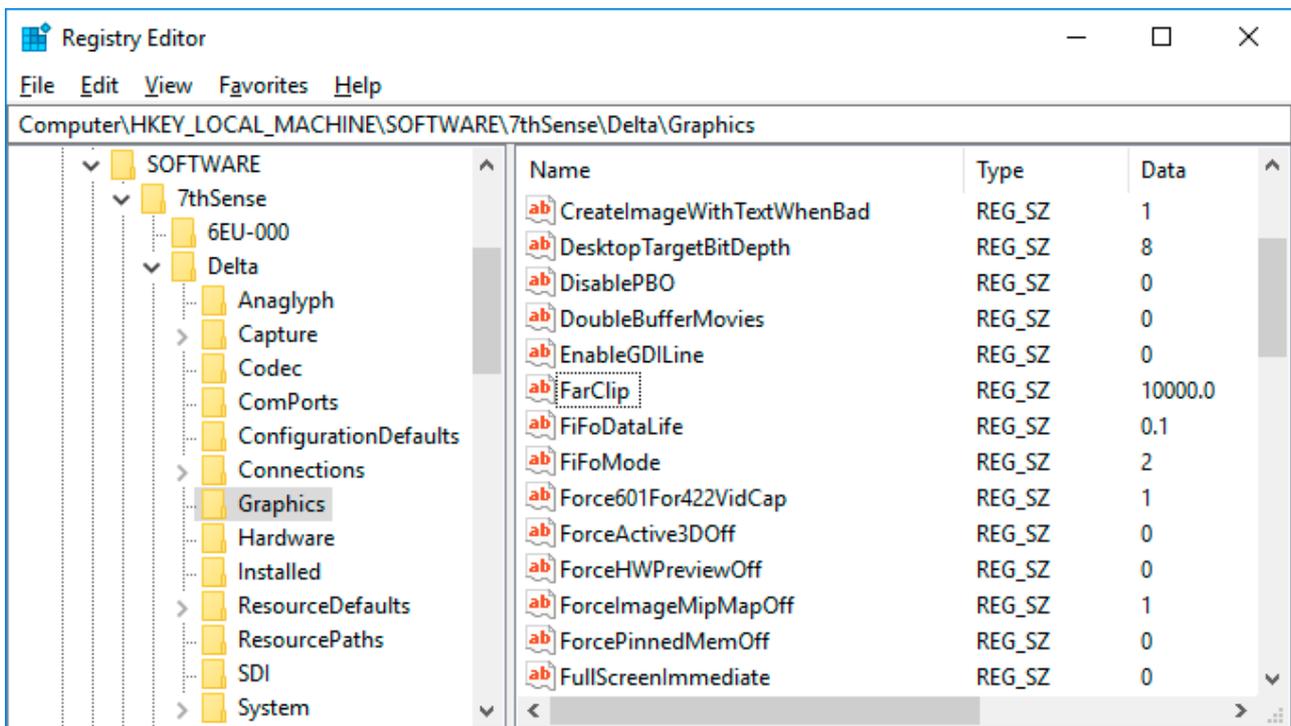
Registry Edits

7thSense software uses the Windows Registry extensively, and on occasion we make recommendations for changes.

- Before making any changes, close DeltaGUI and DeltaServer.
- DeltaServer can take up to 10 seconds to fully close in the background, even if it appears that all the Windows have closed. Ensure that Delta has *fully* closed before opening and/or refreshing, and finally editing. The best way to ensure this, is to watch Windows Task Manager > Details list. You will see 'DeltaServer.exe' in this list if DeltaServer is still running. If it won't close for any reason, you can right click on it in the list and choose 'End Process Tree'.
- Make a note of any setting before you change it, in case you want to change it back.
- Only make the changes as recommended and only to those registry keys.

Go to Windows search > regedit >

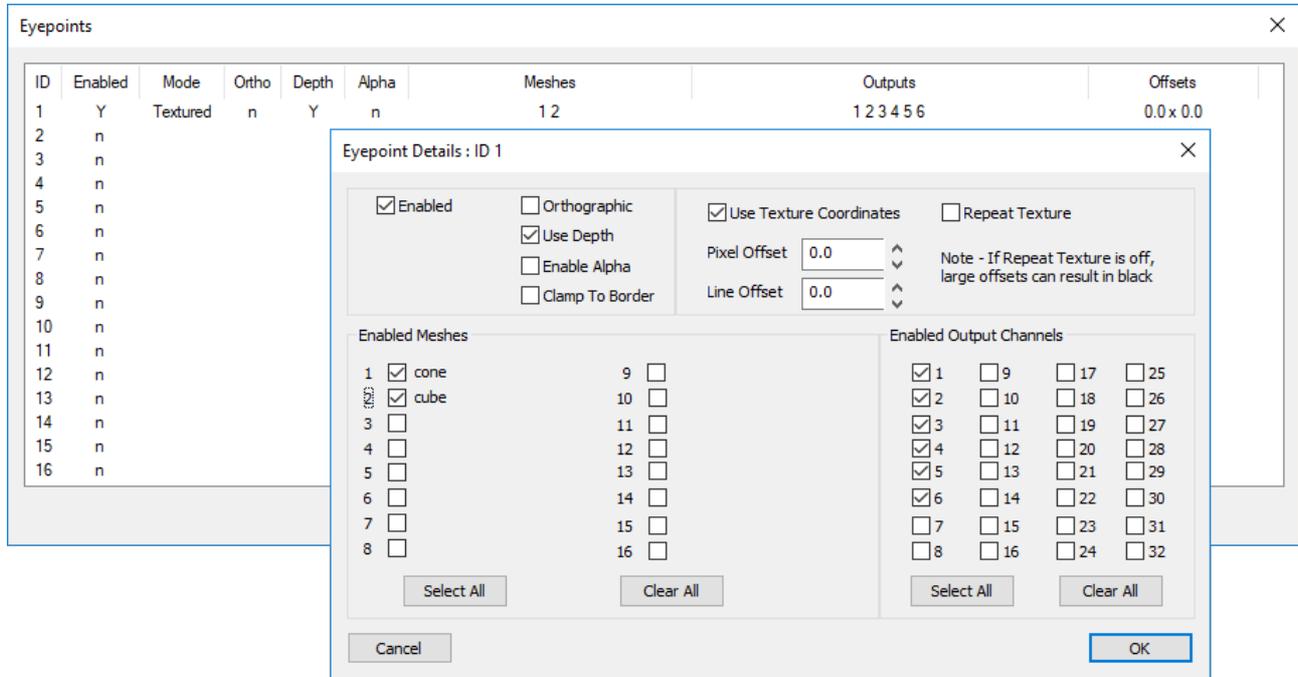
Computer\HKEY_LOCAL_MACHINE\SOFTWARE\7thSense\Delta\Graphics and find FarClip. The default is 10000 and can be increased as required.



Note that the far clip should be set to just beyond the model limits to preserve rendering precision in the 0 – FarClip range.

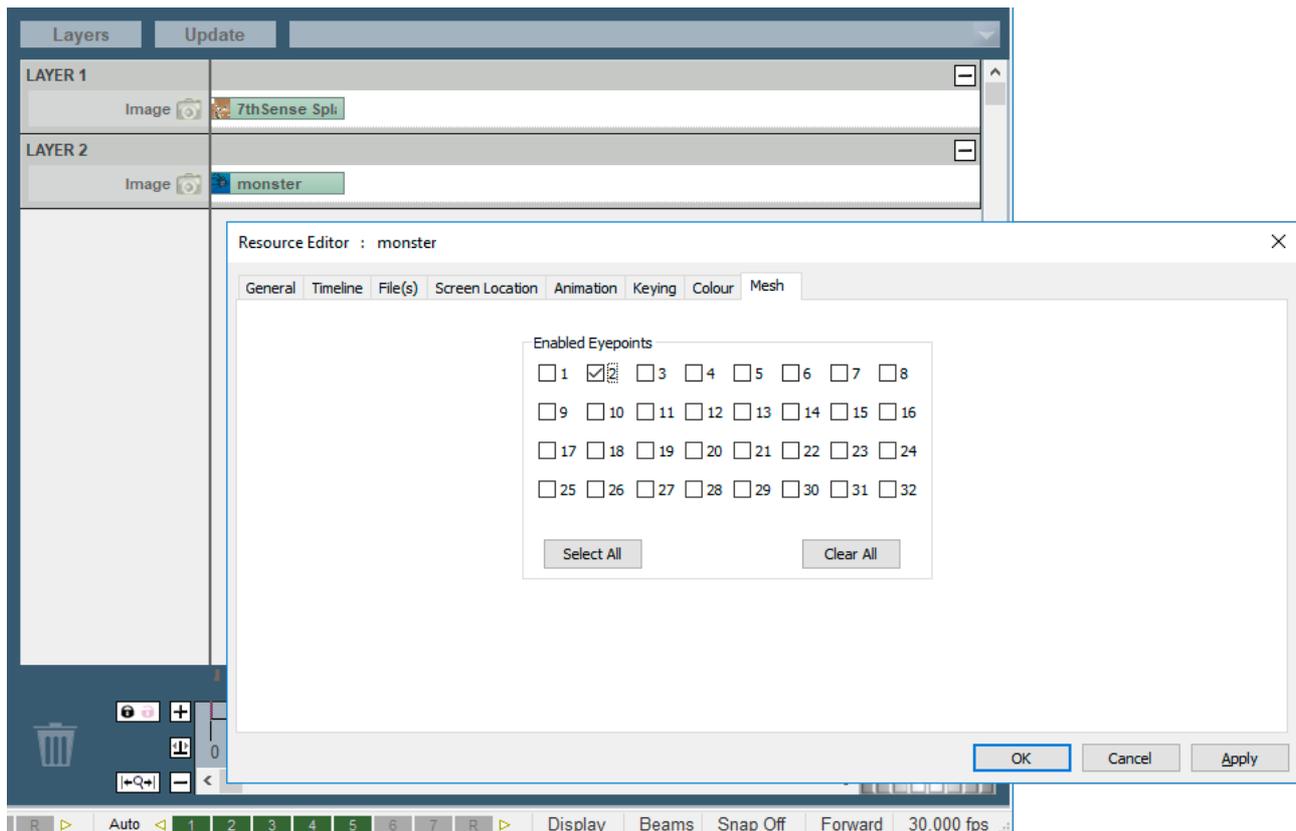
Working in Textured Mode

Working in textured mode is little different, but since there is no single eyepoint, parameters need not be added. When assigning an eyepoint for each channel in *Display > Mesh Mode > Mesh > Eyepoint Files > Eyepoint Details*, make sure you are using texture coordinates:

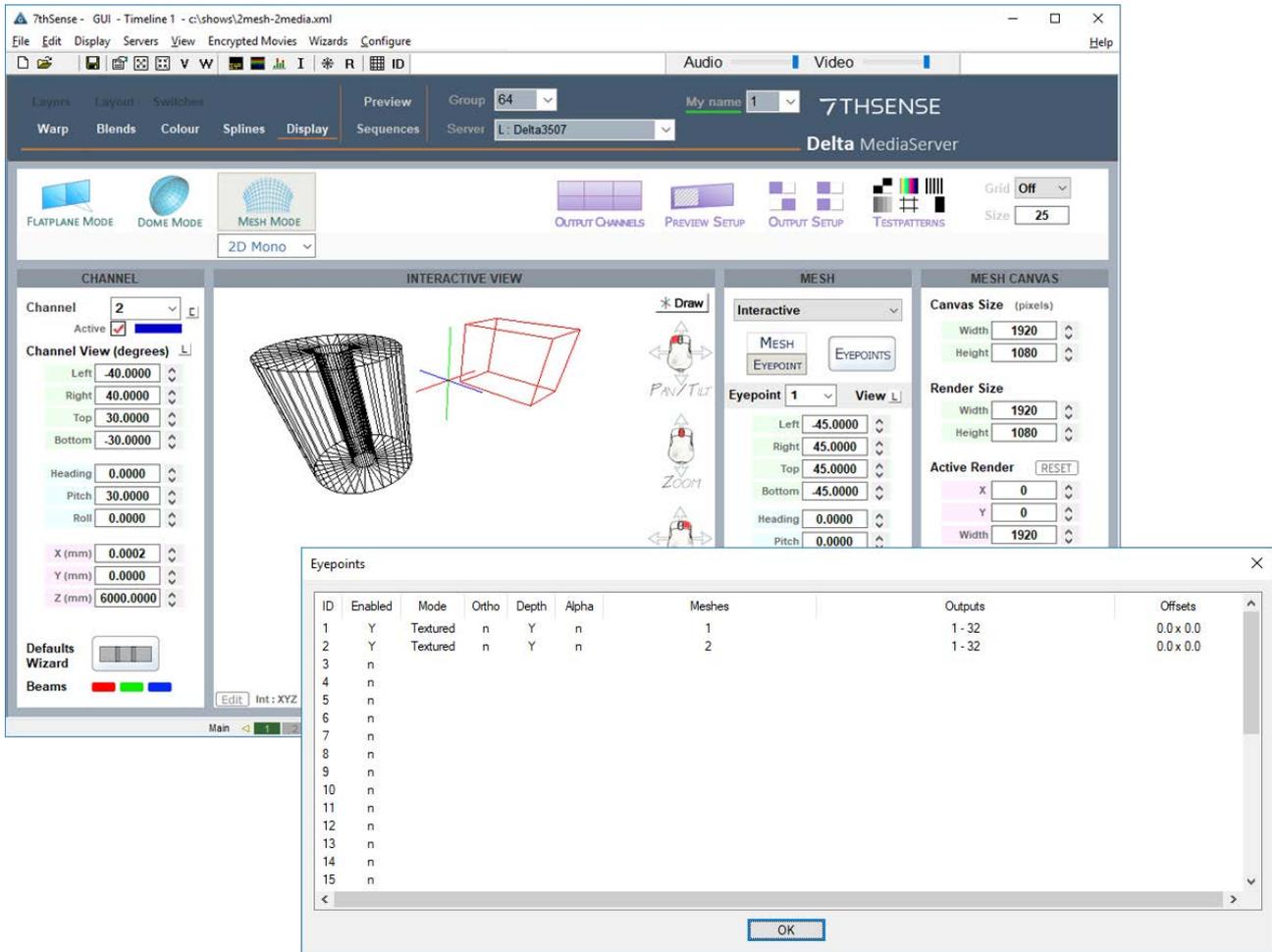


Multiple Meshes and Media

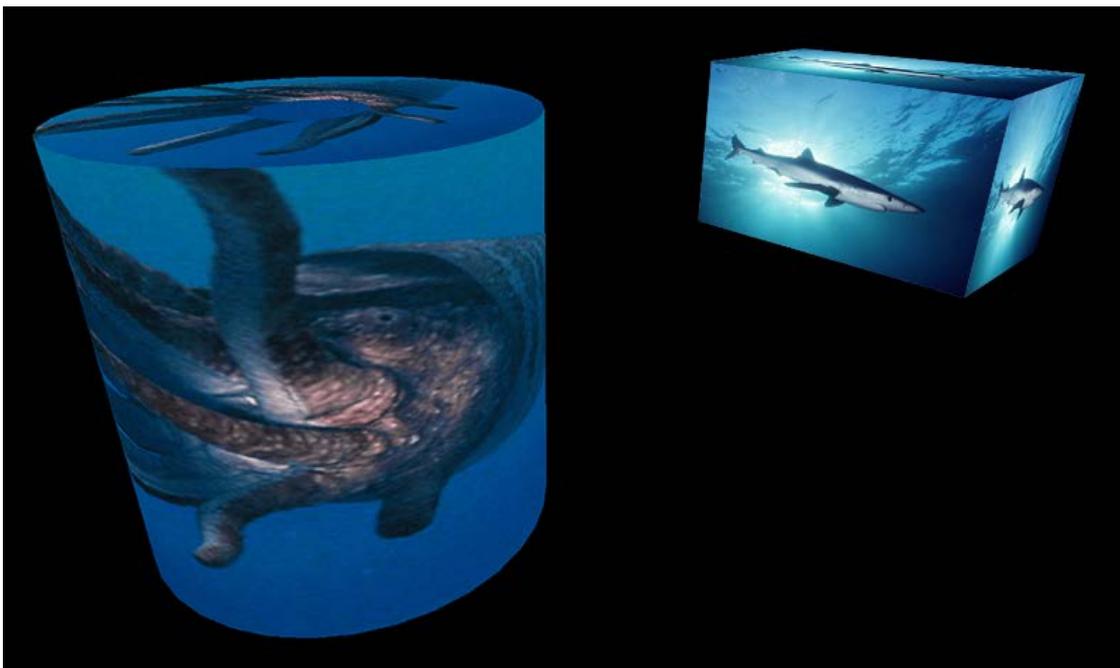
Delta can manage up to 32 meshes (16 prior to version 2.6.48). Since in mesh mode, media can be associated with specific eyepoints, and eyepoints can be associated with specific meshes, we can have simultaneous media on a timeline, to be projected onto separate objects using their own meshes. Here we have two images, and have assigned them to different eyepoints by disabling all others:



Here we have two meshes, for a tube and a cube, set up in DeltaGUI, with different eyepoints assigned to them:



Here in Playback we have media 1, associated with eyepoint 1, which is associated with mesh 1 (the cube), and media 2, associated with eyepoint 2, which is associated with mesh 2 (the tube):



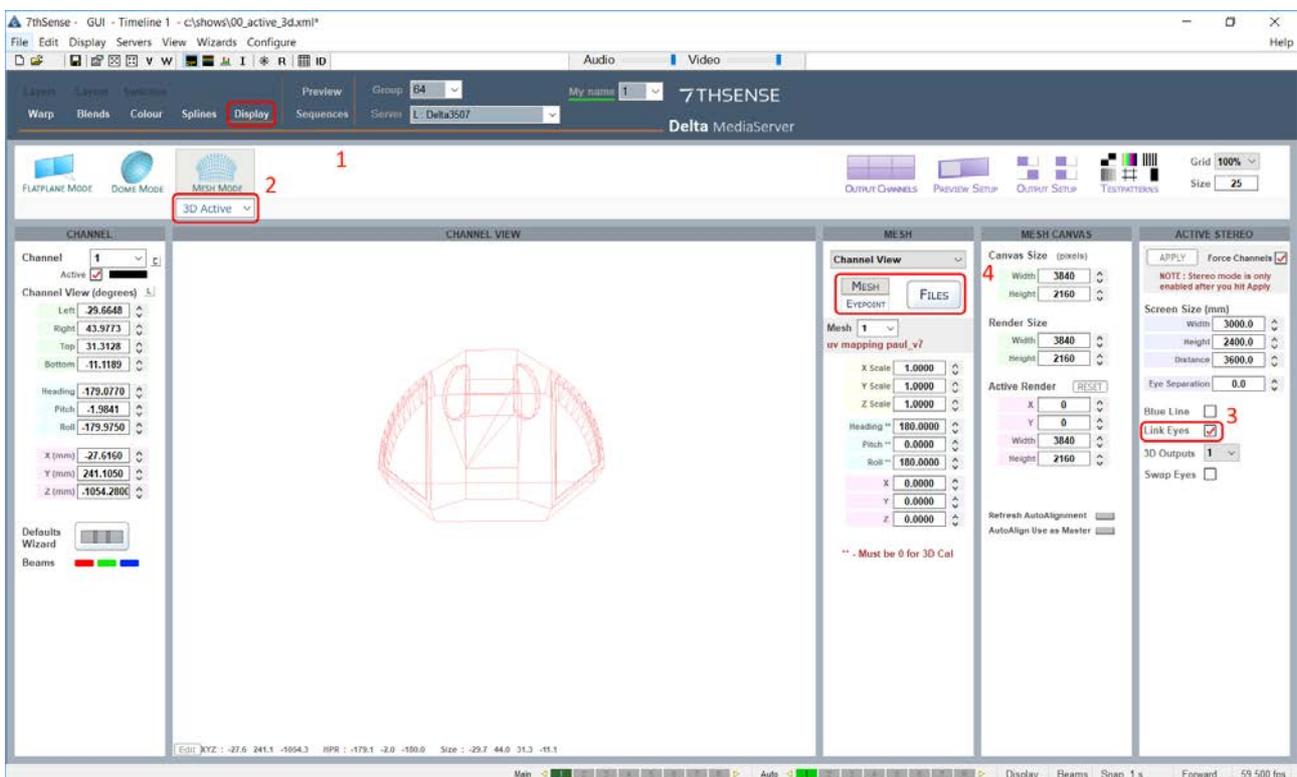
Note that the media eyepoints are using textured coordinates.

Active 3D Mesh Mode

In Active 3D for mesh mode, two copies of a mesh, each with respective left or right eyepoint, are required.

Paired left- and right-eye media are played to separate channels (e.g. 1, 3, 5 and 7 for the left eye, and 2, 4, 6 and 8 for the right eye). By linking eyes, the same warp and blend is applied to both left eye (odd number) and right eye (even number) channels. Mesh files and their eyepoints are assigned to the respective left and right eye media running on the timeline.

Here is the workflow to achieve this.

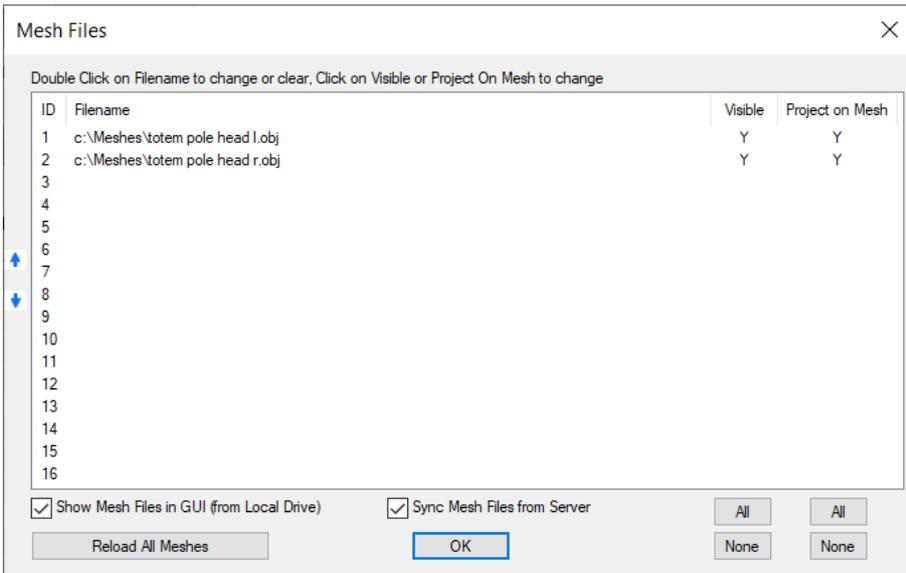


In DeltaGUI, Display workspace (1), select Mesh mode and 3D Active (2). Link Eyes (3) to link any warp and blends applied to left-eye channels also to right-eye channels.

Mesh Files

Now add the left and right mesh files of your model or surface.

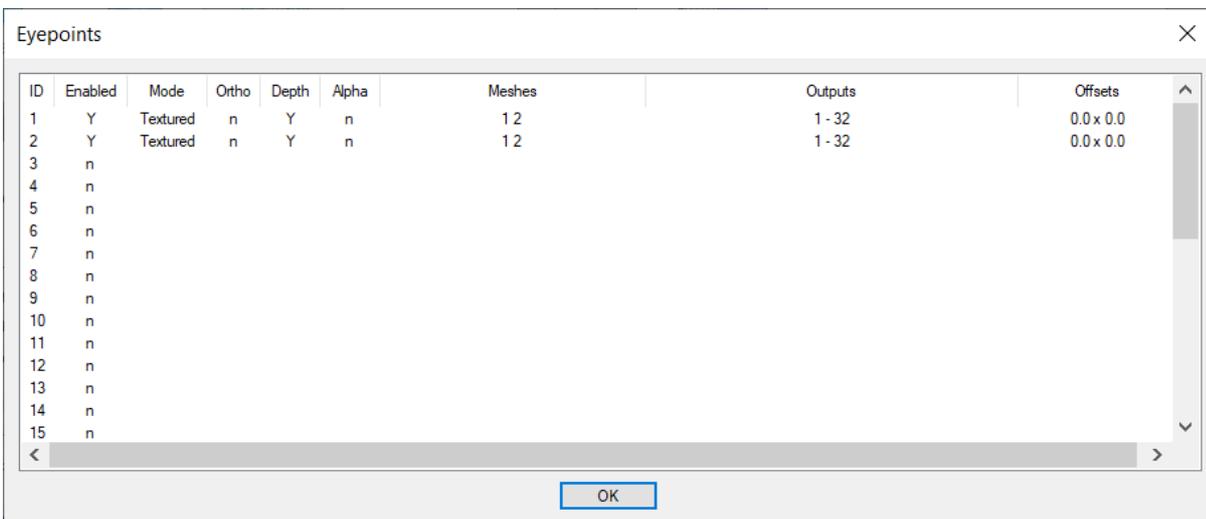
Click on the Files button (4) and in the new dialog, double-click on the first ID row. Use the browse button to locate the left-eye mesh file you want to use. Do the same for row 2 to add the right-eye mesh file.



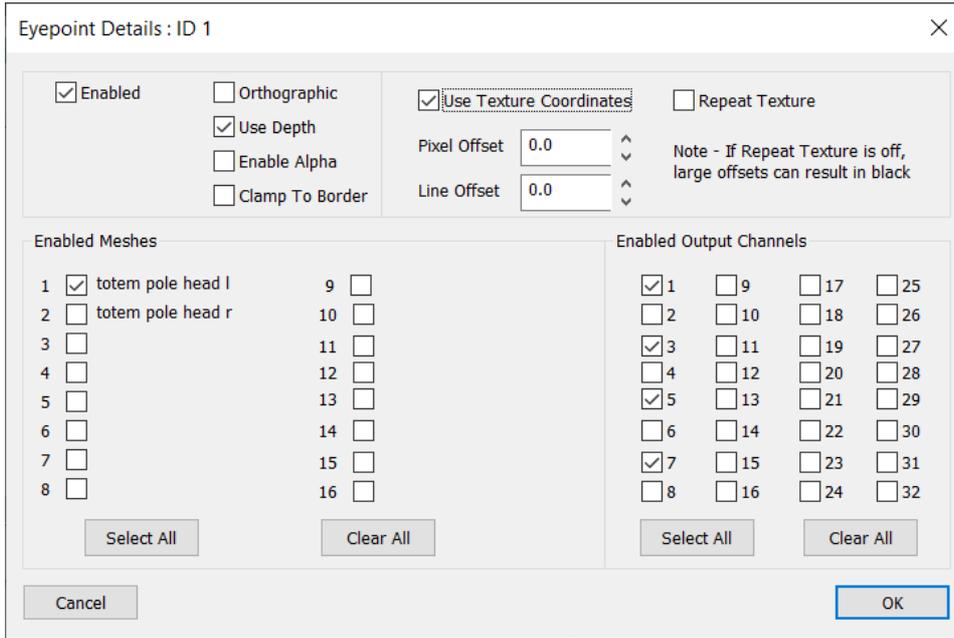
Click 'OK' to close the dialog.

Eyepoints

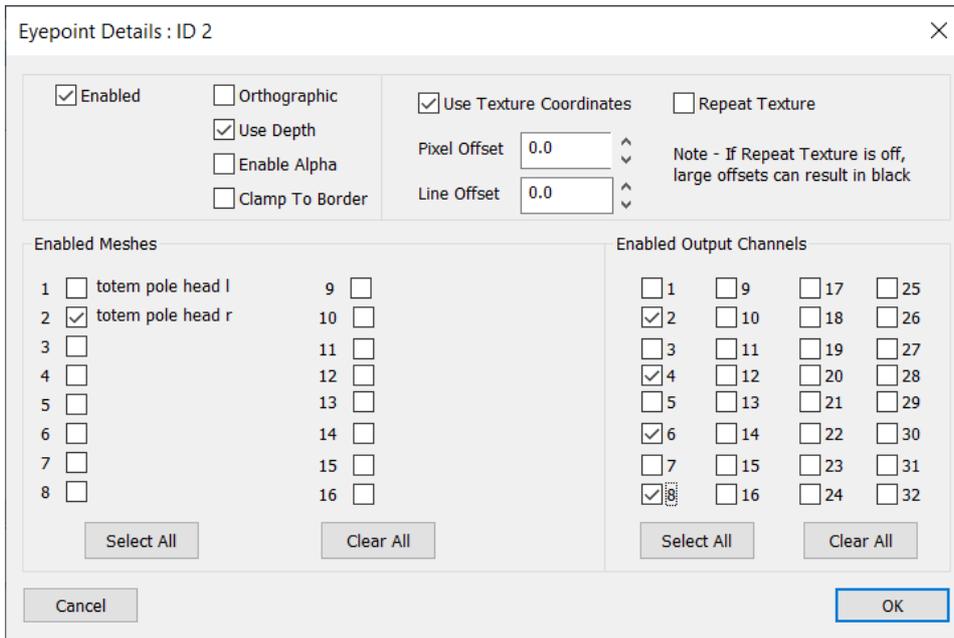
Select 'Eyepoints' below 'Mesh', and then the 'Files' button again (as marked (4) above).



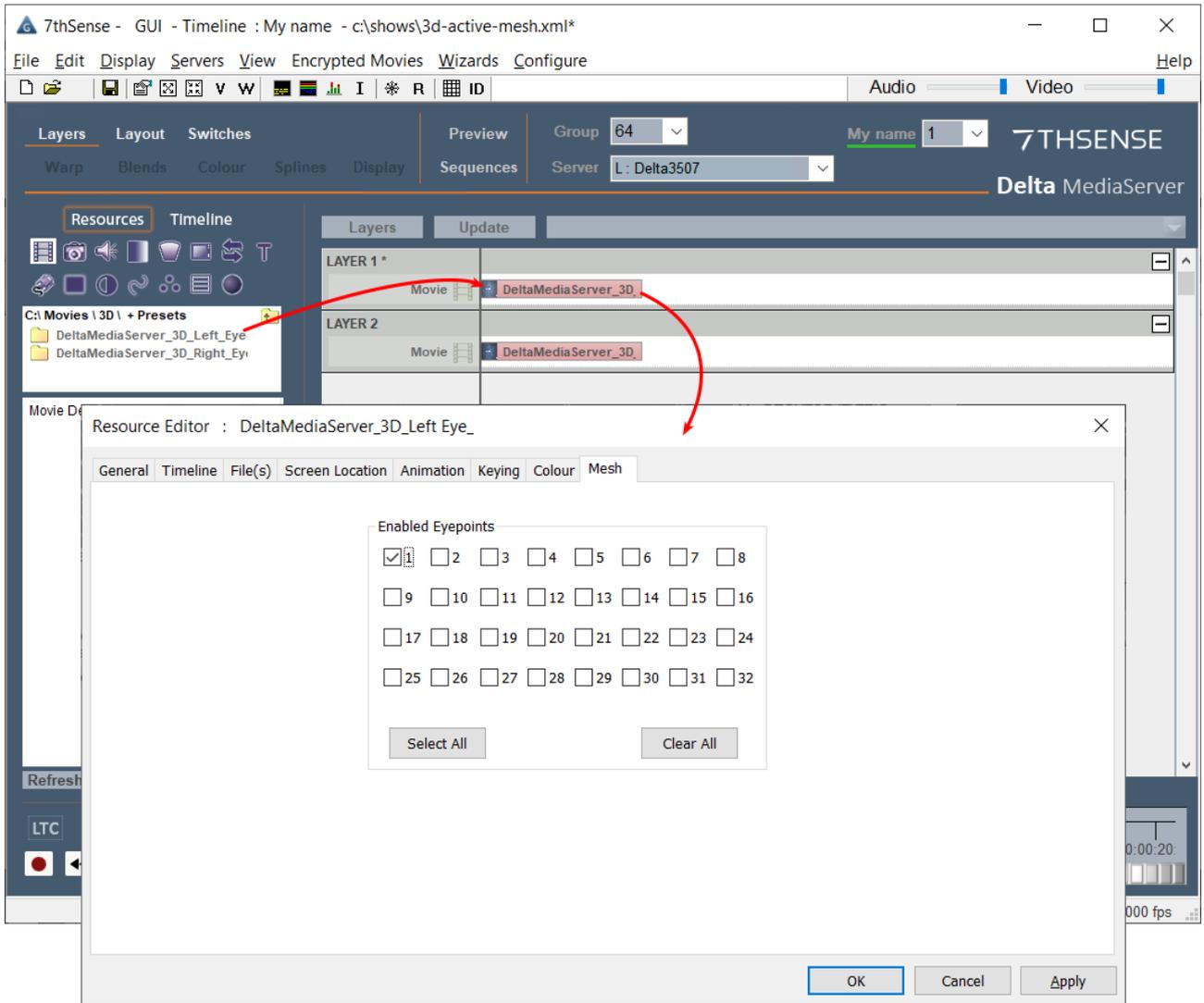
Double-click the row for eyepoint 1 (left mesh), and in the new dialog (below), 'Clear All' meshes, and enable just mesh 1. Then under 'Enable Output Channels', click 'Clear All' and check just the odd numbered channels in the required range (in the example, for an 8-channel, 4-pair setup):



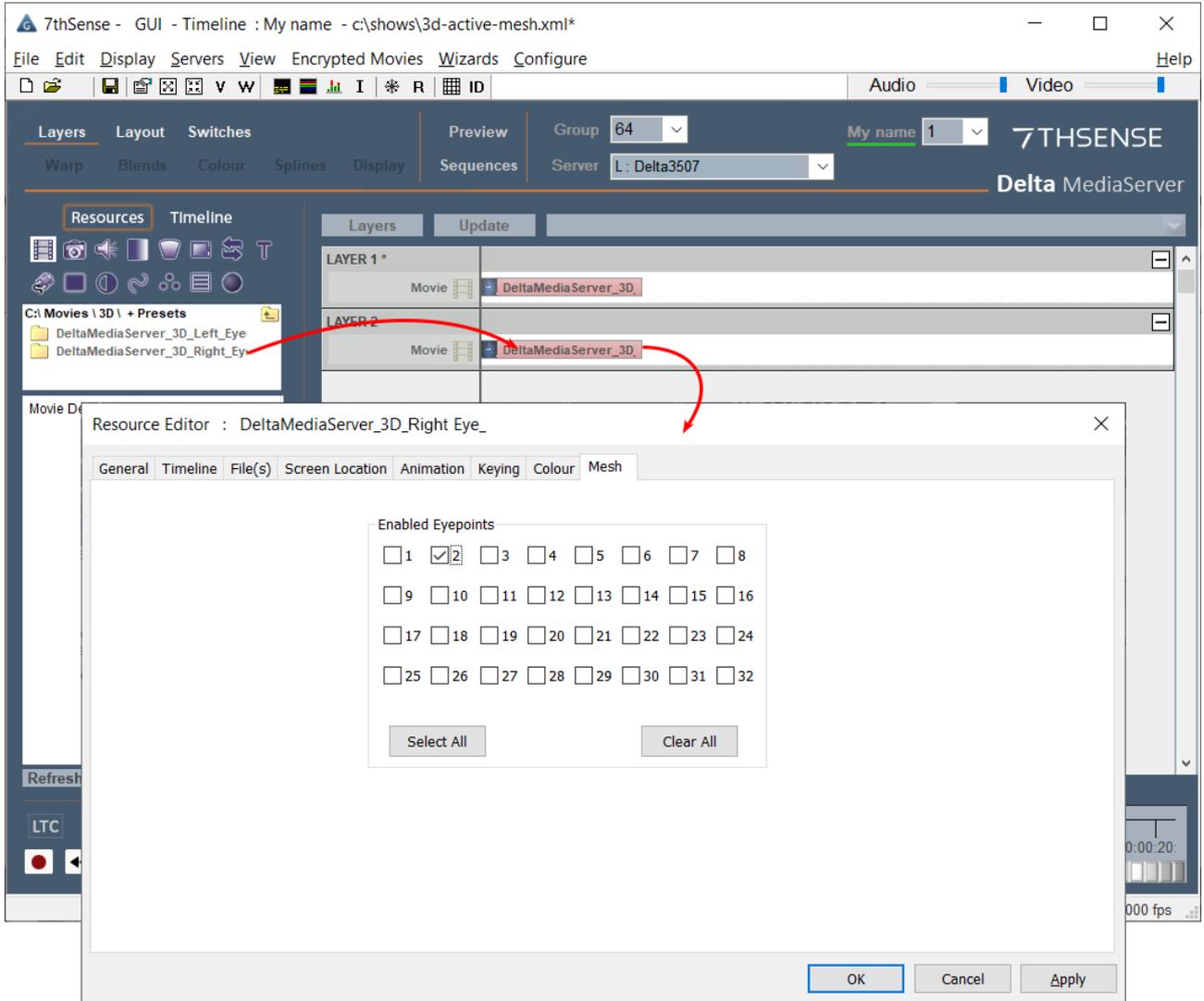
Click 'OK' to close, and this time select eyepoint row 2. Select only mesh 2 and apply only the even numbered channels in the required range:



For the left eye media on the timeline, double click to open the resource editor, select the 'Mesh' tab, clear all eyepoints and enable only eyepoint 1:



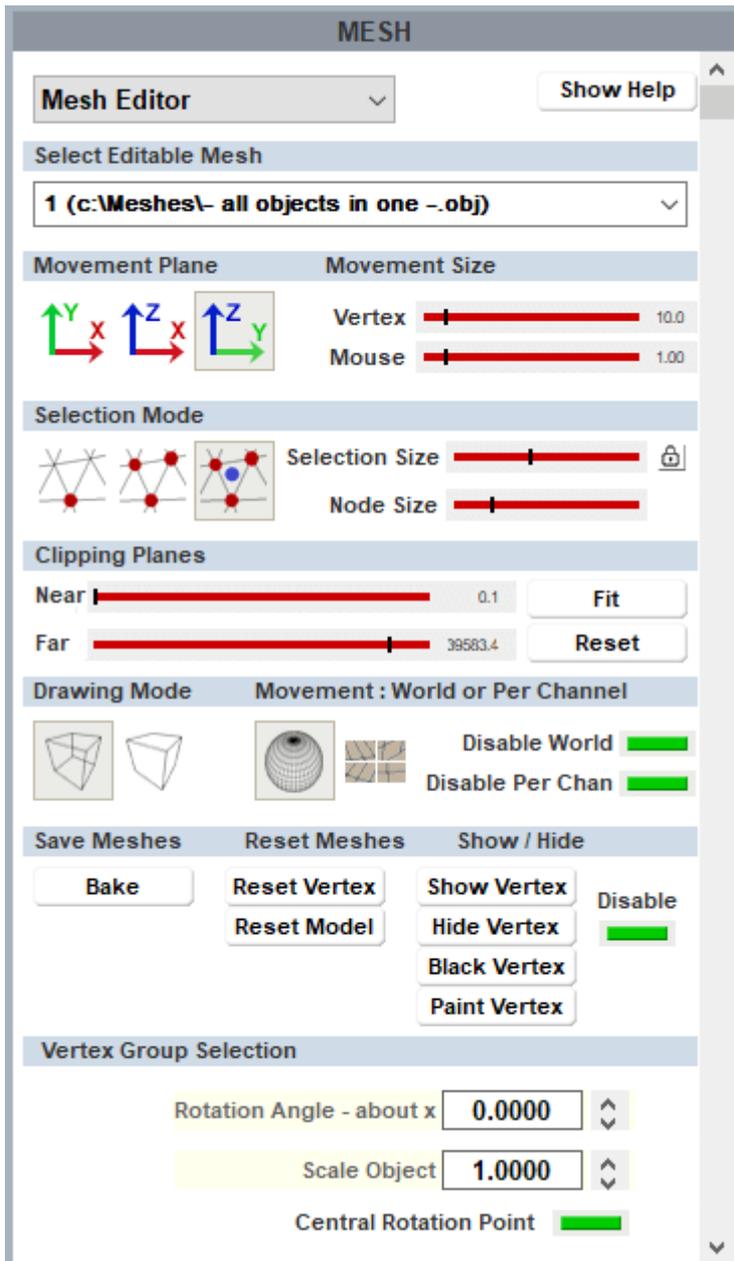
For the right eye media, open the resource editor, clear all eyepoints and enable only eyepoint 2:



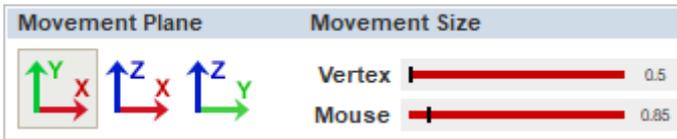
Editing a Mesh

The Mesh Editor allows you to manipulate the mesh as a whole, per channel, by vertex or group of vertices, with control over the editing view. **The X, Y, Z axes throughout are those based on the .obj model's origin.**

Even if you are not editing points, the mesh editor has settings for moving the mesh around in the GUI, with slider controls for Node Size and Mouse movement size.



Movement Plane



Select which 2D plane (axis pair) in which you want to move selected vertices, in the overall 3D space.

Movement Size

The **Vertex** slider controls sensitivity of selected vertex movement, using Ctrl+arrows. The **Mouse** slider controls the speed at which the mouse drags the mesh around in the GUI.

Drawing Mode

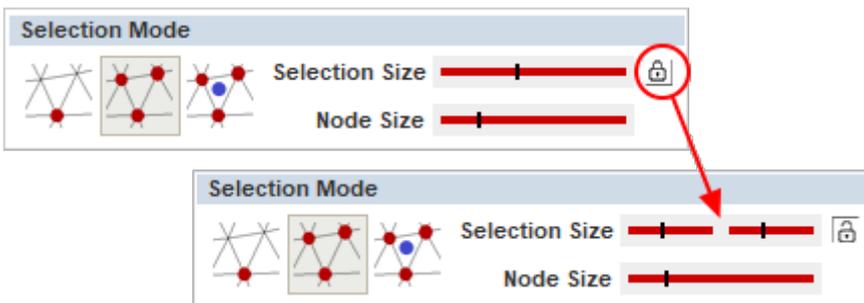


These options enable you to see the entire mesh, or focus on the foreground only. The result will depend in part on the way the mesh has been constructed.

Movement: World or Per Channel

Choose whether you want to manipulate the mesh for a particular channel only, or for all channels (World view). Per channel is generally best for small adjustments, and World if there is a significant error in the media across the whole object. These edits are overlaid on the .obj mesh and will be saved as separate .vto files. The Disable buttons switch the adjustment overlays off and on without resetting the model.

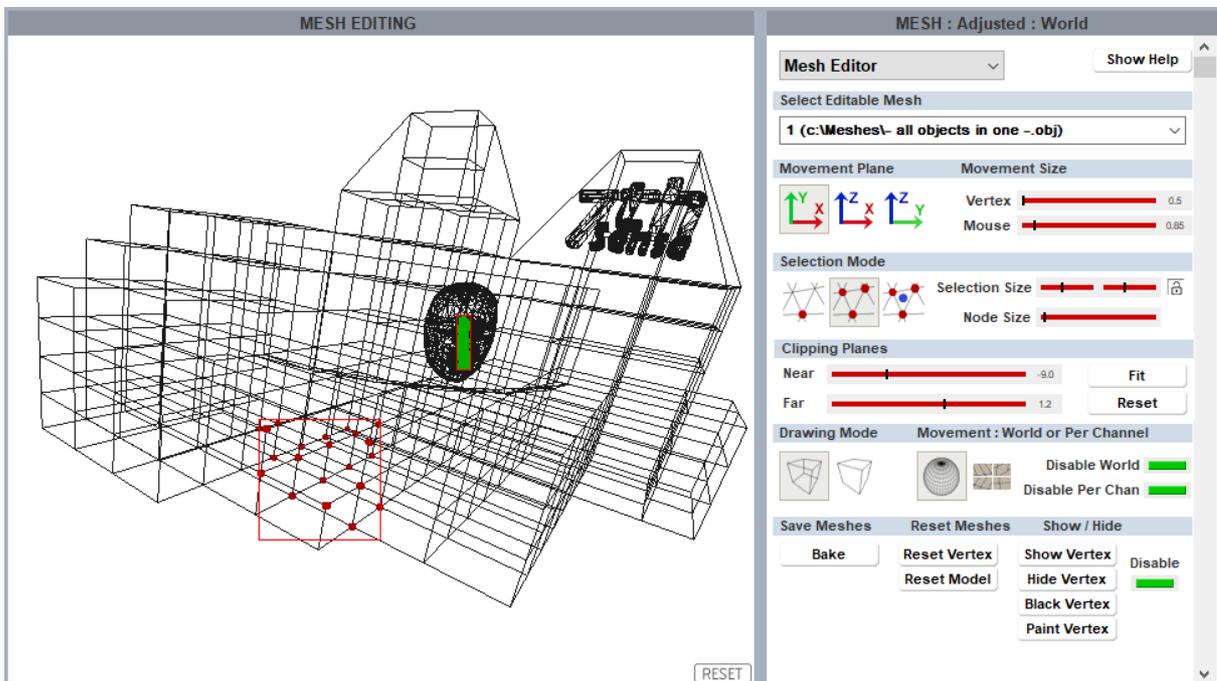
Selection Mode



Vertices can be selected either to reposition them, or to change their visibility (as in the oblique faces of our object example above). Select a single vertex to move it, or a group of vertices, or where a set of linked vertices has been constructed in the .obj file, select the group.

The group selection area finds all vertices within an XY window *as seen from the current viewpoint in the GUI, not relative to the size of the obj mesh*. The size of this window can be controlled by the size slider. Click the **lock** button for separate X and Y axis sizes. *Selection includes all vertices in this rectangle, in all planes*. Add more vertices to the group selection with **Ctrl+left click**. Near or far vertices can be excluded by working in a clipping plane (see below). For multiple selection of individual vertices, use a very small group area.

In this example: red points are a selected group with equal XY selection. Green shows the mouse hovering over another group with a limited X axis and mid-range Y (see settings on the right).



Moving points: Once selected, all red nodes can be moved around *en bloc* using **Ctrl+arrows**. For finer movement, use **Ctrl+Shift+arrows**. Movement is along the X, Y, Z axes of the .obj file, based on the object origin. Selected vertices or the whole model can be reset using the **Reset** buttons (bottom right).

Moving nodes around

Whenever groups of connected nodes are moved, this inevitably distorts the surrounding mesh. A mesh that is not an accurate representation of the real world can, to some degree, be pulled into position, but there are limits beyond which making one part fit makes another worse.

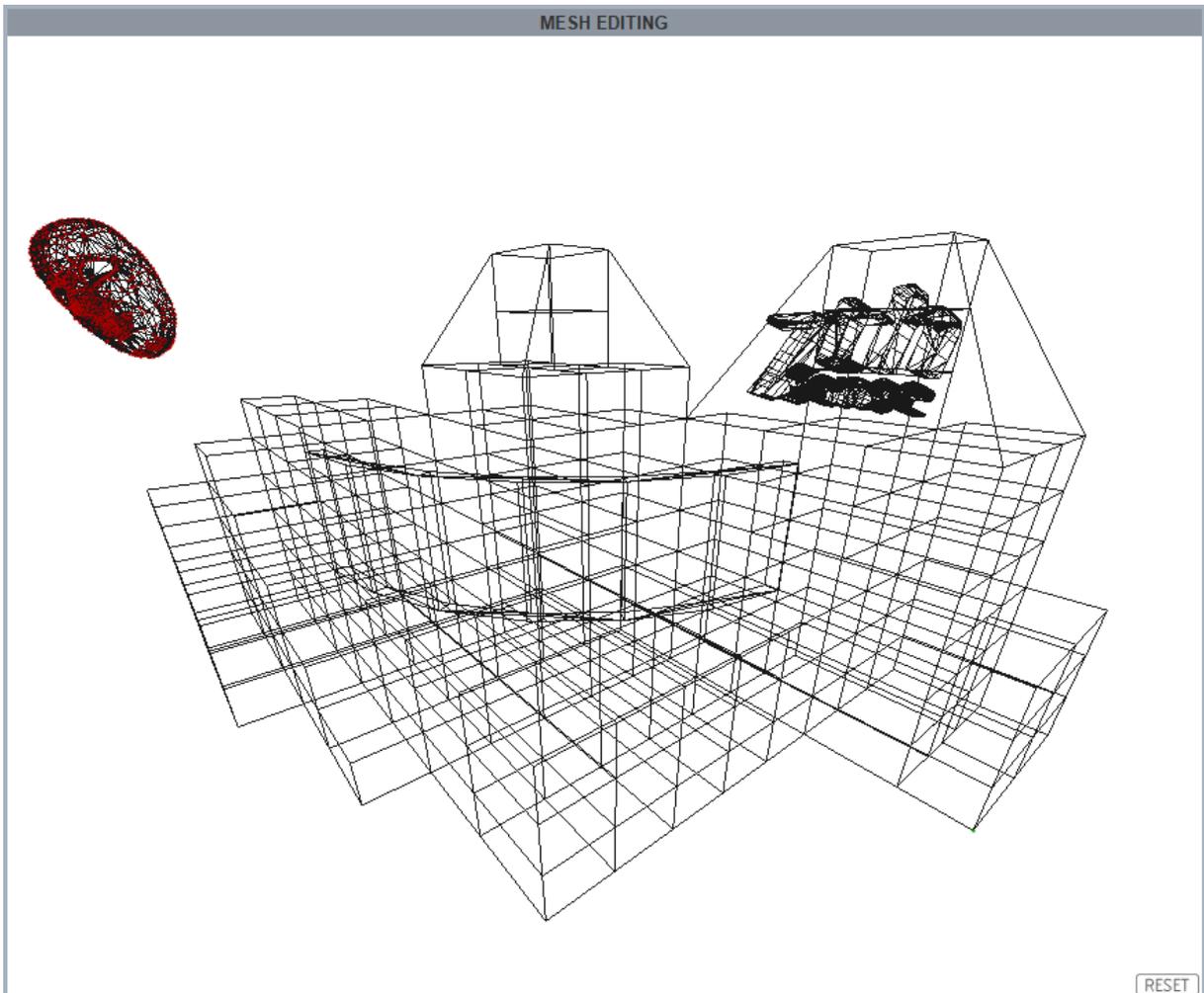
Vertex Groups

In complex models it is helpful to construct the .obj file out of groups of vertices where the groups are not connected directly to each other: for example, the detailed face illustrated above is separable. In these circumstances, the separable section of linked vertices can be selected by use of the third mesh selector in Delta (the one with the blue dot): the 'vertex group selection'. The separable part can now be manipulated, to move it, or rotate it on any axis about a selected point, or its median point.

In the illustration below, the face element from the front of the building has been selected with a single click because it forms a vertex group, unconnected to the main building. Lateral

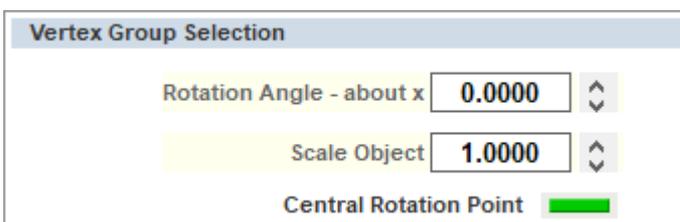
movement is as elsewhere using Ctrl+arrows, or Ctrl+Shift+arrows (finer) according to the movement size set in the editor panel, and the selected movement plane axes.

Please note that depending on the complexity of the vertex group being manipulated, this can be very demanding of DeltaGUI.



Rotation is available, again in the selected movement plane axis, either about the central rotation point, or a selected point (click the green button to switch this).

The vertex group can also be **scaled** about the same point, increasing or decreasing all vertex separations proportionately. Here we can see that by selecting the X-Y movement plane, rotation will be about the Z-axis:



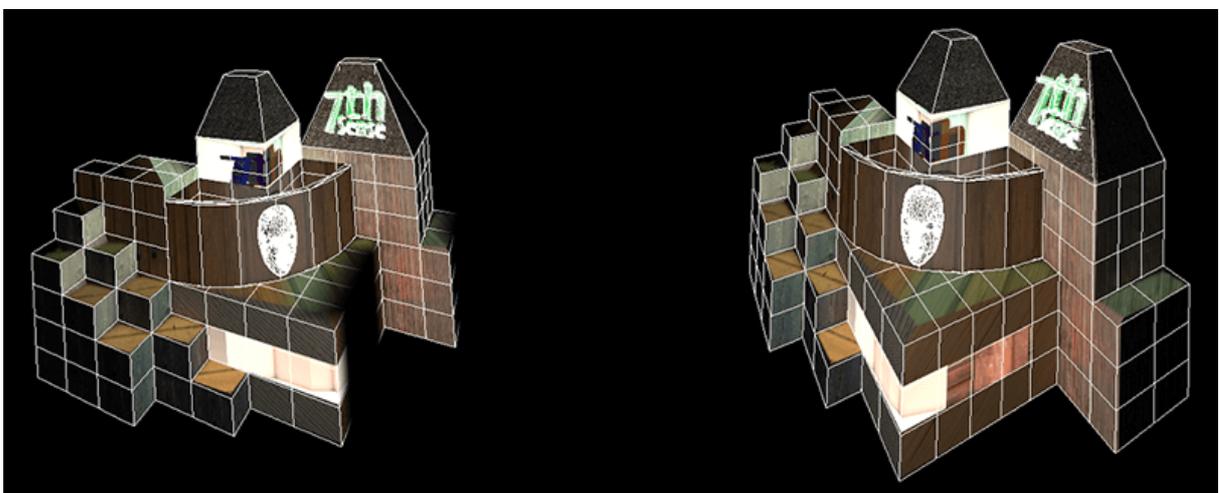
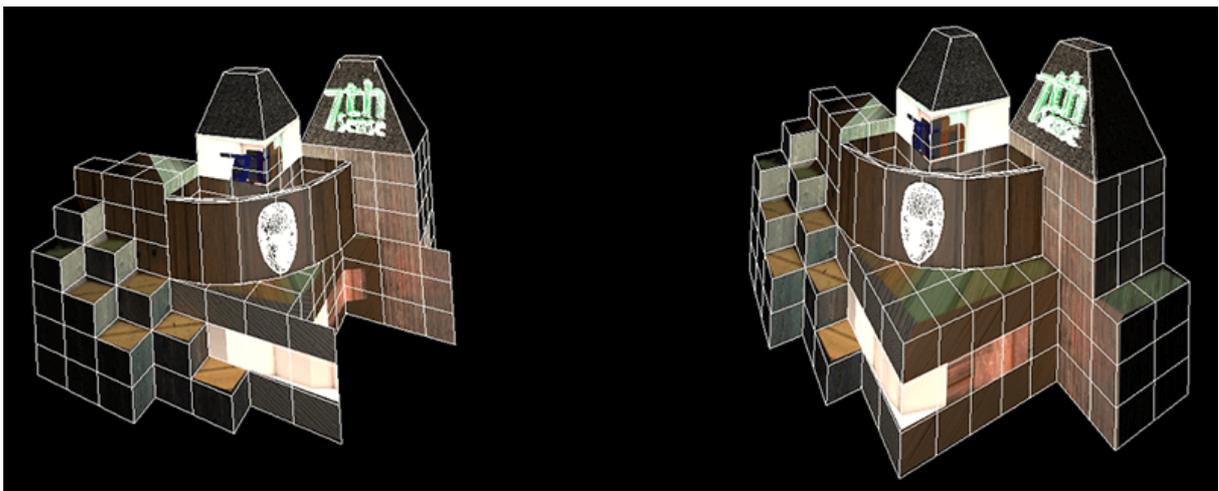
These adjustments can be seen in the editing screen above.

What moves?

If the mesh and media are texture-mapped, the media will follow the moved element, so the media can be adjusted to a real-world object that does not quite match.

If the media is not texture-mapped (i.e. projection mode), then it will continue to be projected in situ. This is not the same as warping the mesh.

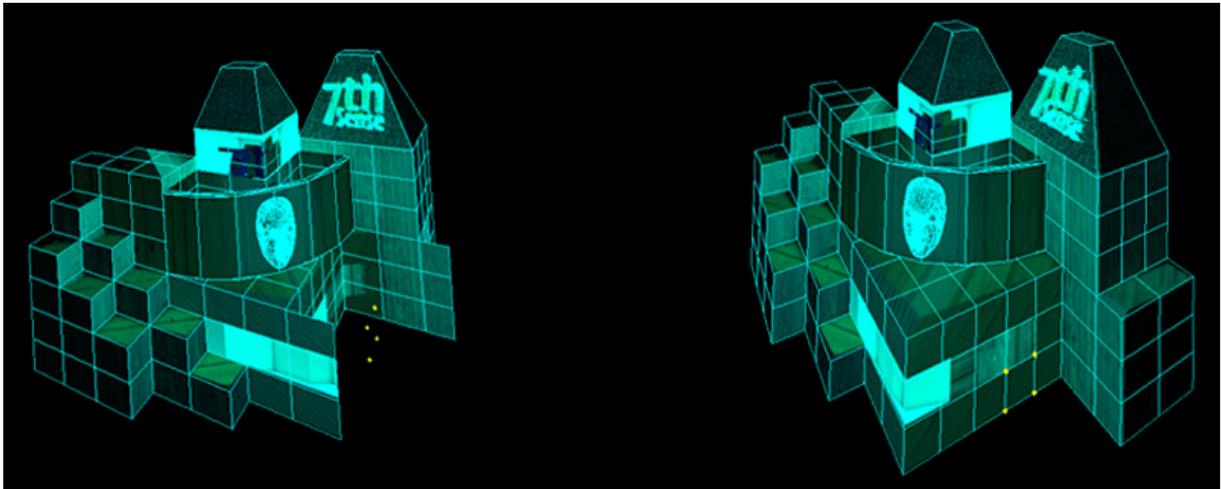
Hiding points: There are two ways to affect the projected display of selected parts of the media: Show/Hide and Black/Paint. Hiding a group removes all edges joined to the selected vertices, and enclosed faces. Single vertex selection is not useful for hiding, but a single vertex in a small group may be. In the example below, per channel selection has been made on the oblique sides in channel 1 (left), to be filled exclusively by channel 2 (right). In the first Playback Window illustration below, **Hide** removes faces from display, in the second, **Black** fades it out, similar to blending.



Note how vertex selection has not included the vertical edges required by channel 1.

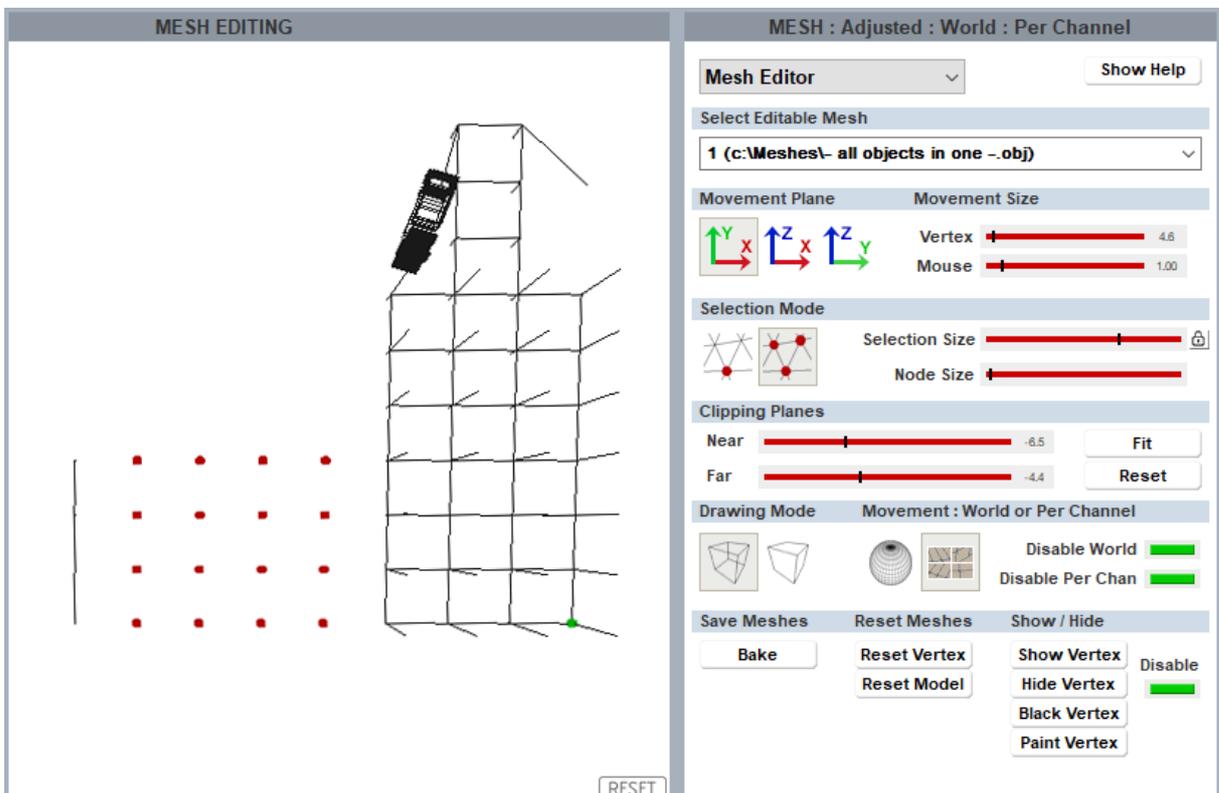
Compare this with [adding a blend](#) ³², and the need to preserve display on edges; a blend may fade out more than you want. Blends are more suitable for textured mode.

When mesh editing, Playback will show in blue-green colour, indicating that it is in low-performance mode. Restore this by going back to channel mode.



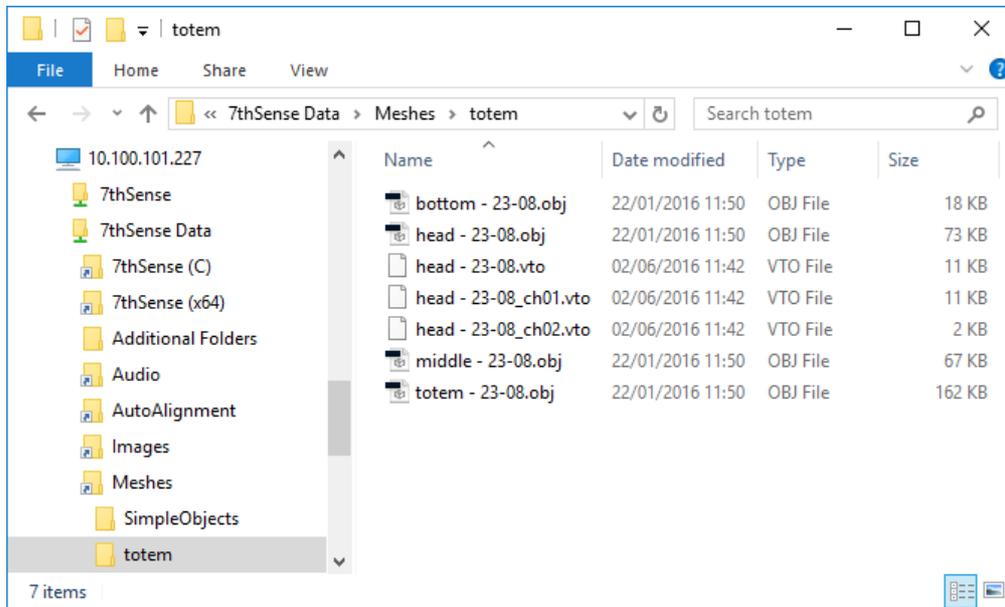
Clipping Planes

These contain your view for selection, between Near and Far planes. Use the **Fit** button to set the range to the size of your mesh, otherwise the mesh can be too small a part of the maximum available range, and disappear from view. In this example, clipping planes have been used to limit selection to points on the facing wall in the above illustration:



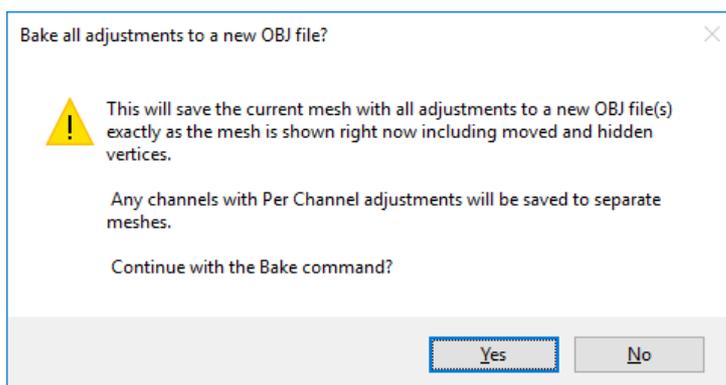
Save Meshes

You can save changes to the mesh simply by using *File > Save As* (and then *Save* to save further changes). This saves the original .obj file, together with .vto files containing adjustment instructions. Every time the mesh is used for a show, these instructions will be applied. 'World' and 'Per Channel' change files are created separately.



To remove the changes, returning to the original mesh alone, simply reset in the GUI or delete the .vto files here. If you have 'Sync Mesh Files from Server' ticked in the Mesh selection screen, the files will also be copied over automatically to the GUI.

The mesh editor, however, also allows you to 'bake' the changes into either a new single revised 'world' .obj file, or a new .obj per channel. Changes cannot be reset in a 'baked' file.



Reset Meshes

Note that resetting the whole model sets all vertices back to their original positions in their .obj file. If you have loaded an .obj file that has associated .vto files, resetting the model removes the .vto amendments from view, and if then saved, the .vto files are removed from the C:\Meshes directory. If in doubt, and you want to compare before and after adjustments, use the Disable button instead.

Movement Nodes

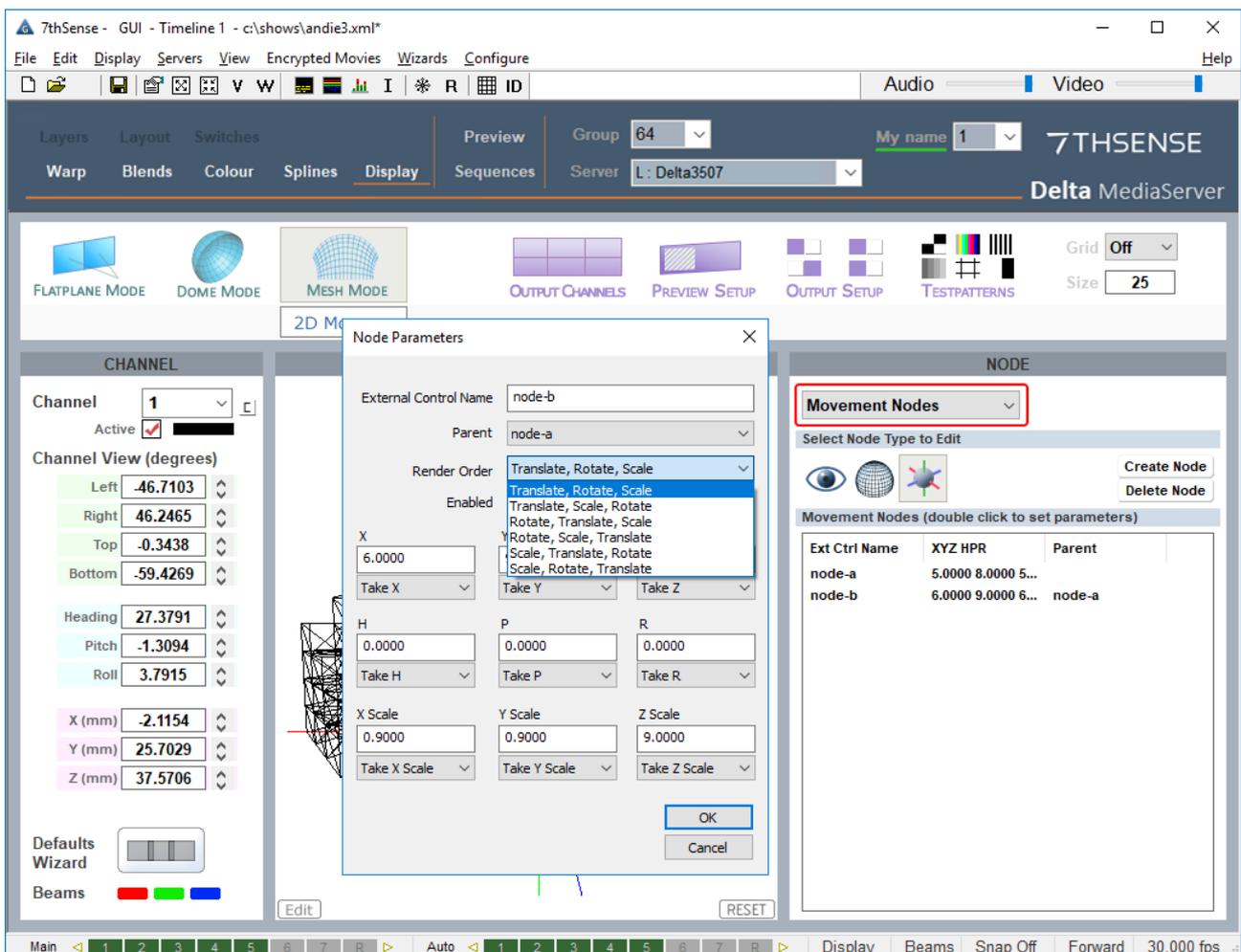
Movement nodes add the capability to synchronise movement between meshes, media and eyepoints using their names in **Sequences** (see Delta User Guide). By manipulating a parent node, linked nodes will follow without separate programming. Examples include rotating images on car wheels, growing and moving two eyes together, or bringing several meshes into view simultaneously.



The three controls are: create or delete nodes (third button), assign nodes to eyepoints (first button), and assign nodes to meshes (middle button).

Creating a Movement Node

Node translations for each parameter can be set by the setnodeparams external control command, or use the Node Parameter editor in *Display Config > Mesh Mode > Movement Nodes* and double click on a node. There are six sequences of Translate (change XYZ coordinates), Rotate (HPR angles) and Scale (XYZ relative dimensions), though not all three aspects may be needed.

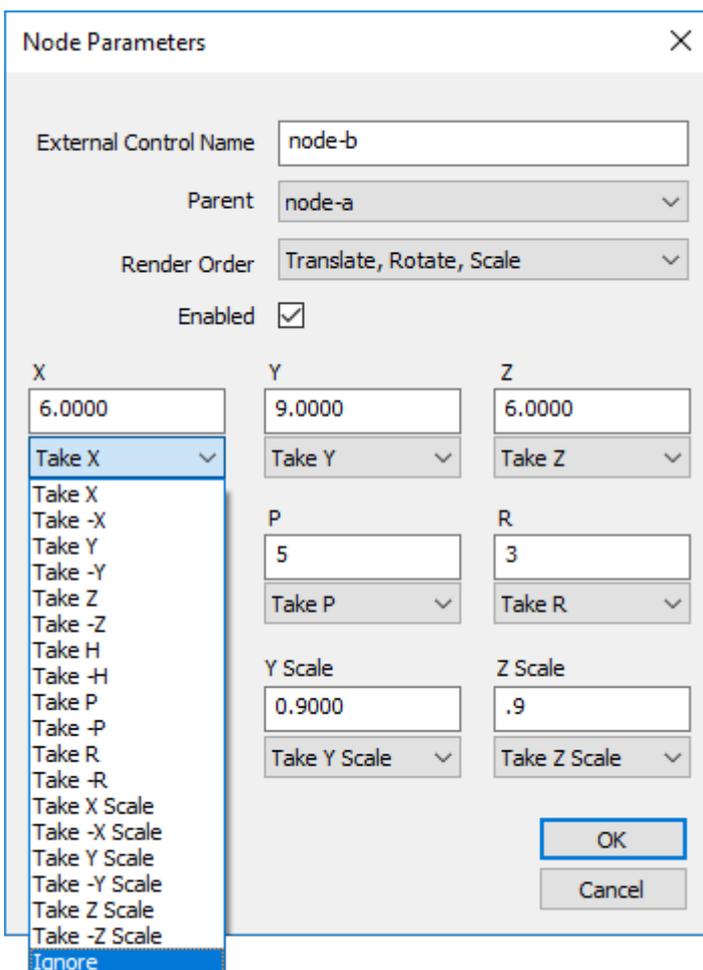


- To synchronise multiple movement nodes, one must be a parent and the others assigned to it.
- Movement nodes are assigned to eyepoints and/or meshes.

- Movement nodes can also be addressed by External Controls (see the Delta User Guide for how to use these). For available commands in DeltaGUI, Click *Help > External Controls* and find Movement Node Commands:

Command	Arguments	Description	Examples
CREATEMOVEMENTNODE	extctriname=name [x=xval] [y=yval] [z=zval] [h=hval] [p=pval] [r=rval] [xscale=xscaleval] [yscale=yscaleval] [zscale=zscaleval] [order=orderstring] [enabled=y/n] [parent=extctrinameofparent] [takex=**] [takey=**] [takez=**] [takeh=**] [takep=**] [take=**] [takeyscale=**] [takezscale=**]	Create a movement node which can be addressed from external control and meshes or eyepoints attached to it (see ** below)	
SETPARENTNODE	target=[mesh/eyepoint/node] [index=1-16 / ALL] [node=nodeextctriname] [parent=extctrinameofparent] [detach=true/false]	Attach a mesh, eyepoint or other node to this node	setparentnode target=mesh index=1 parent=myparentnode setparentnode target=eyepoint index=3 parent=myotherparentnode setparentnode target=node node=nodeextctriname parent=myparentnode setparentnode target=mesh index=ALL detach=true
DELETENODE	None	Delete an existing movement node	
SETNODEPARAMS	extctriname=name [x=xval] [y=yval] [z=zval] [h=hval] [p=pval] [r=rval] [xscale=xscaleval] [yscale=yscaleval] [zscale=zscaleval] [order=orderstring] [enabled=y/n] [parent=extctrinameofparent] [detach=true/false] [takex=**] [takey=**] [takez=**] [takeh=**] [takep=**] [take=**] [takeyscale=**] [takezscale=**] [takeyscale=**] [takezscale=**]	Set an existing node's parameters (see ** below)	

- Each parameter of the node (XYZ, HPR, XYZScale) can be set to act as any of the others – or be ignored, so for example the X value used for a node can be taken from Z or even -Z, or the XScale value can be set to be ignored:



- The values shown in the dialog are always the untranslated values, the translation happens internally in the render stage.

- This translation can be used to take care of axis differences between systems or to ignore certain values from a datastream that are not interesting.
- Node values can be set by external control (using setnodeparams), ArtNet (via setnodeparams) or NatNet tracking input.

Mesh Optical Tracking

For users of OptiTrack Motive systems for optical tracking, Delta can use Motive's NatNet streaming data over IP, linked to DeltaGUI's [Movement Nodes](#) ⁽⁵²⁾.

In OptiTrack Motive

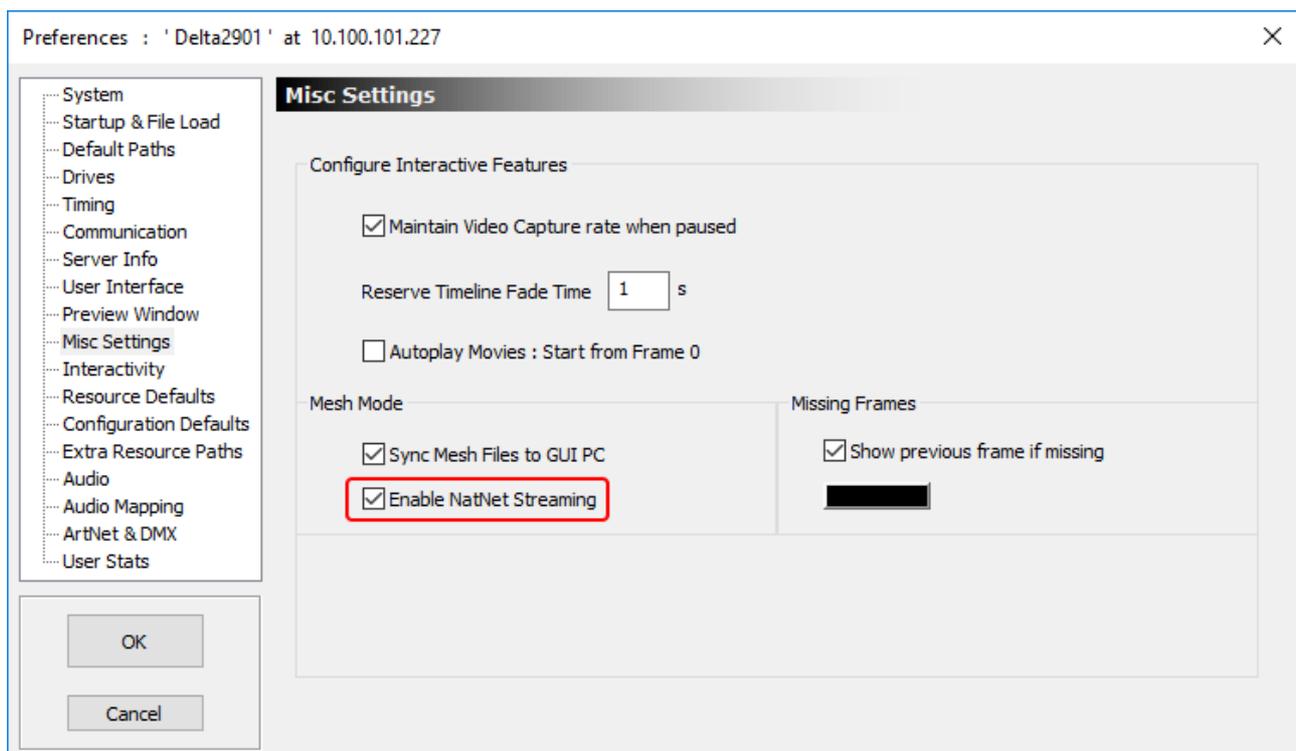
Enable 'Broadcast Frame Data' in the OptiTrak Streaming Engine and connect the Motive source Data Port with the Delta server as client over NatNet. For details on the streaming engine, see https://v22.wiki.optitrack.com/index.php?title=Data_Streaming.

Output node **1** will be identified in Delta as movement node name: **natnet_1**

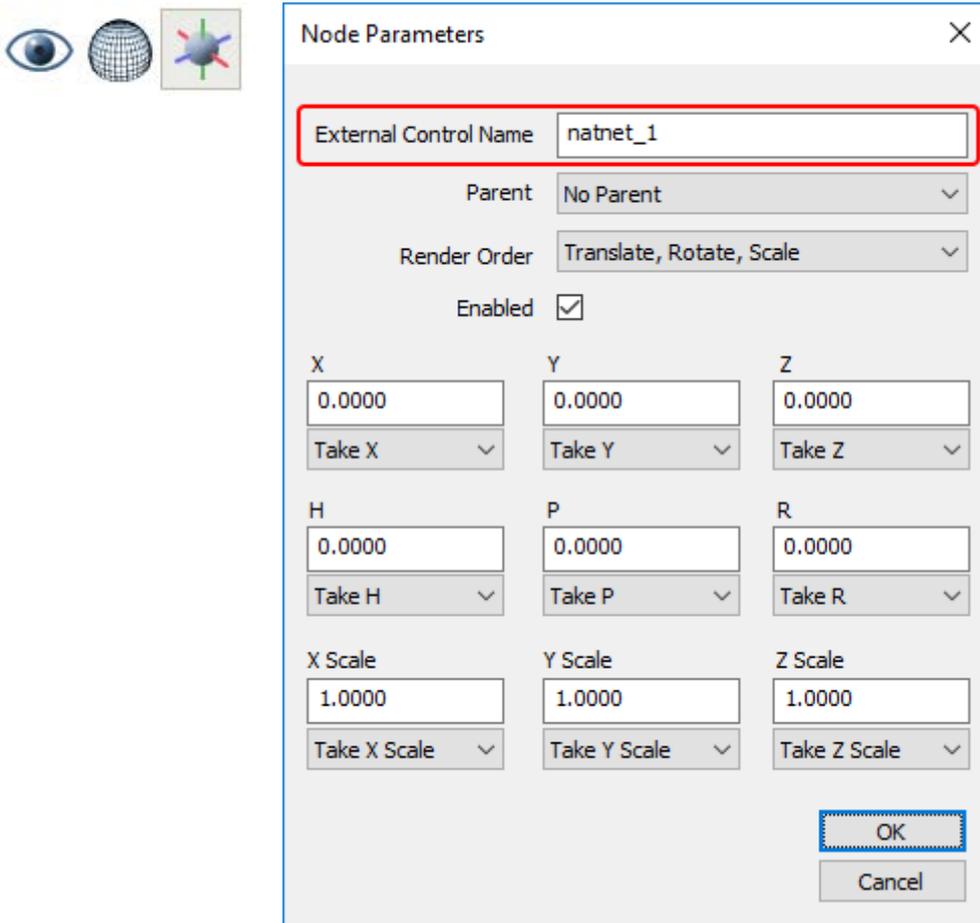
Output node **2** will be identified in Delta as movement node name: **natnet_2** and so on.

In DeltaGUI

First, NatNet must be enabled in DeltaGUI > *Preferences* > *Misc Settings*:

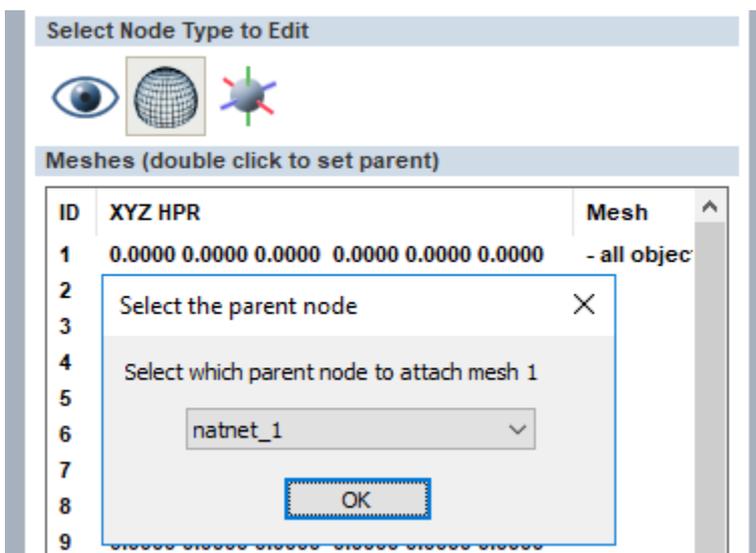


Connections can now be made to meshes and/or eyepoints, by creating and naming a **movement node** for each:



The node(s) can then be synchronised with a mesh or an eyepoint, by making the NatNet node a parent.

Example: take the mesh you want to connect the NatNet node to as parent, and double-click to edit it:



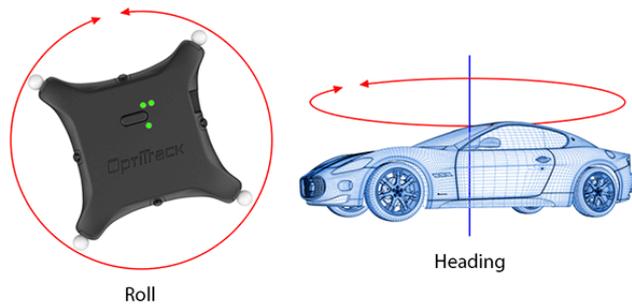
Assign the NatNet node name as parent.

The XYZ-HPR and scale values of the NatNet node can now be taken by the mesh (or eyepoint).

Transformation Example

Node values be used directly, or in more complex scenarios, can be transformed, for example to reverse orientation, invert, or convert roll to pitch. For example, here we want to take the orientation of an active puck and transform its **roll** movement to change the **heading** of a car:

Node Parameters		
External Control Name	natnet_1	
Parent	No Parent	
Render Order	Translate, Rotate, Scale	
Enabled	<input checked="" type="checkbox"/>	
X	Y	Z
0.0000	0.0000	0.0000
Ignore	Ignore	Ignore
H	P	R
0.0000	0.0000	0.0000
Take R	Ignore	Ignore
X Scale	Y Scale	Z Scale
1.0000	1.0000	1.0000
Ignore	Ignore	Ignore
OK		
Cancel		



Document Information

Date	Document edition	Software version	Revision Details	Author/Editor
August 2017	3	Delta 2.5	New edition	Andie Davidson
May 2018	4	Delta 2.6	Updated for Delta 2.6 3D calibration methods, movement nodes, tracking, and vertex group selection enhancements.	Andie Davidson
April 2019	5	Delta 2.6.48	Updates to channel and mesh selection for background warps and blends.	Andie Davidson
June 2020	6	Delta 2.6, 2.7	Inclusion of process for Active 3D mesh setup	Andie Davidson
July 2020	7	Delta 2.6, 2.7	Leader-Follower terminology updated	Andie Davidson

.

- .obj files 45
- .vto files 45

E

- editing a mesh .obj 45

M

- mesh mode 4
 - 2D/3D view (calibration) 20
 - 3D calibration 20
 - active 3D workflow 40
 - apply lens distortion 20
 - blending 32
 - canvas 13
 - channel beam controls 13
 - clipping planes 45
 - DeltaGUI 13
 - document information 58
 - editing a mesh .obj 45
 - editing points 20
 - explained 5
 - eyepoint 13
 - eyepoint parameters 9, 12
 - how it works 9
 - lens modes 20
 - linked vertex selection 45
 - manipulating the mesh view 20
 - media 5
 - mesh editor 45
 - mesh not showing on playback 33
 - movement nodes 52
 - node movement size 45
 - node selection size 45
 - node size 45
 - projection (flat) mode 5
 - projector positions 13
 - setup sequence 11
 - textured (UV) mode 5
 - textured mode 36
 - troubleshooting 33
 - using multiple meshes 37
 - warp, distort or edit? 33
 - warping a mesh 30
- mesh models 5
- motion tracking 55

N

- NatNet streaming 55

O

- optical tracking 55
- OptiTrack Motive 55

R

- restoring a mesh .obj 45

T

- troubleshooting
 - mesh mode 33
 - mesh not showing in playback 33

U

- UV mapped media 5

E: info@7thsense.one
W: 7thsense.one

7thSense Design Ltd
2 The Courtyard, Shoreham Road
Upper Beeding
Steyning
West Sussex
BN44 3TN
UK

T: +44 (0) 1903 812299

7thSense Design LLC, Michigan
332 E Lincoln Ave
Suite 100
Royal Oak, MI 48067
USA

T: +1 248 599 2717

7thSense Design LLC, Orlando
4207 Vineland Rd
Suite M1
Orlando, FL 32811
USA

T: +1 407 505 5200