Contents

Stage overview

What is the Stage? .................................................................................................................. 7
Objects within a stage .......................................................................................................... 7
Stage overview .................................................................................................................... 10
What is the Stage? ................................................................................................................ 10
Objects within a stage .......................................................................................................... 10
Editing the Stage ................................................................................................................ 13
Stage editor ........................................................................................................................ 13
Manipulators ...................................................................................................................... 26

Screen types overview

What is a screen? ................................................................................................................ 30
Types of screens ................................................................................................................ 30
Screen types overview ...................................................................................................... 32
What is a screen? ................................................................................................................ 32
Types of screens ................................................................................................................ 32
Creating screens .............................................................................................................. 34
Deleting screens ............................................................................................................... 38
Changing the Screen Type ............................................................................................... 40
Removing screens ............................................................................................................ 41

Screen editor

Screen editor ....................................................................................................................... 43
Screen properties .............................................................................................................. 44
Offset ................................................................................................................................. 44
Rotation .............................................................................................................................. 44
Scale .................................................................................................................................. 44
Resolution ........................................................................................................................ 44
Mesh .................................................................................................................................. 44
Warping outputs ........................................................................................................ 160
Blending and masking outputs .................................................................................. 171
Dynamic Blend .......................................................................................................... 176
Enabling Dynamic Blend .......................................................................................... 176
Global Dynamic Blend Settings ................................................................................ 177
Projector Level Dynamic Blend Settings .................................................................. 178
Include Exclude list .................................................................................................... 180
Using the Include & Exclude list ................................................................................ 180

**Projector studies**

**Cameras**

Overview ...................................................................................................................... 187
Workflow ....................................................................................................................... 187
Camera properties ....................................................................................................... 193
Settings ......................................................................................................................... 195
Preview ......................................................................................................................... 196
Output ........................................................................................................................... 196
Hierarchy ....................................................................................................................... 198
Stage Render .................................................................................................................. 199
Camera type .................................................................................................................. 200
Visibility ......................................................................................................................... 201
Physical .......................................................................................................................... 203
Lens ................................................................................................................................. 204
Prediction ....................................................................................................................... 205

**Visualiser renderer overview**

Schematic ...................................................................................................................... 207
Lux .................................................................................................................................. 207
Heatmap ......................................................................................................................... 207

**Heatmap**

Overview ...................................................................................................................... 209
Workflow ....................................................................................................................... 209
Example ......................................................................................................................... 209
Heatmap properties ................................................................. 213

Lux
Overview .................................................................................. 215
Workflow .................................................................................. 218
Example .................................................................................. 218

Schematic
Overview .................................................................................. 223
Workflow .................................................................................. 223
Example .................................................................................. 224

AR workflow
Overview .................................................................................. 228
AR Workflow .............................................................................. 228

VR
Opening the virtual reality navigator .............................................. 231
Using HTC Vive for virtual camera tracking .................................. 232

Venues and props overview
What is a venue? ......................................................................... 234
What is a prop? ........................................................................... 234
Venues and props overview ...................................................... 235
What is a venue? ......................................................................... 235
What is a prop? ........................................................................... 235
Creating and removing props ................................................... 236
Editing props ............................................................................. 241
Prop properties ........................................................................... 242
Creating and switching venues ............................................... 248
Editing venues .......................................................................... 250
Stage overview

What is the Stage?

The Stage is a level in the disguise software that uses the Stage Visualiser to view your current stage from any angle. The terms the Stage and the Stage level have the same meaning but to avoid confusion we will be using the Stage level term most commonly throughout the User Guide.

Left-click stage from the dashboard to open the Stage level.

Objects within a stage

A stage is constructed from a series of objects. These objects are a range of different screen types, venues and props.

Hierarchy of objects contained within a stage: screens, venues and props

Screens

d3 has a number of different types of screens that you can add to a stage. The stage can then be used to play video from the Timeline onto the screens in real-time, avoiding the need for time consuming re-renders.
Venues

A stage can contain multiple *venues*. This is useful if you wish to create a range of stage designs, each venue representing a different stage design option, allowing you to quickly change from one design to another (during a client meeting for example).

**Props**: these venues are constructed from *props*. Props are objects which form the building blocks of your venue.

Multiple Stages

The Stage level can even contain multiple stages, which can be switched between quickly from the same d3 file.

The Stage level can contain multiple stages which can be switched between quickly from the same project file, these stages can share the same screens, venues and props.

Each stage you create can share the same screens, venues and props. In the example above there are two different stages in the same d3 file: **Stage1** and **Stage2**. Notice that both of these stages share certain screen types: **Screen1**, **Screen2** and **Projector1**; they also share specific props: **Prop2**, **Prop4** and **Prop5**. However, their overall prop configurations are different, hence the need to create two different venues; and their overall screen configurations are different, hence the need to create two different stages.
For information on adding, removing, and editing stages, screens, venues and props please read the following sub-chapters.

Please note: For solid performance from disguise hardware it is best to keep the total number of vertex points in your scene below about 150,000-200,000 vertices. This figure varies with the amount of video loading required.
Stage overview

What is the Stage?

The Stage is a level in the disguise software that uses the Stage Visualiser to view your current stage from any angle. The terms the Stage and the Stage level have the same meaning but to avoid confusion we will be using the Stage level term most commonly throughout the User Guide.

Objects within a stage

A stage is constructed from a series of objects. These objects are a range of different screen types, venues and props.

Screens

d3 has a number of different types of screens that you can add to a stage. The stage can then be used to play video from the Timeline onto the screens in real-time, avoiding the need for time consuming re-renders.
Venues

A stage can contain multiple venues. This is useful if you wish to create a range of stage designs, each venue representing a different stage design option, allowing you to quickly change from one design to another (during a client meeting for example).

Props: these venues are constructed from props. Props are objects which form the building blocks of your venue.

Multiple Stages

The Stage level can even contain multiple stages, which can be switched between quickly from the same project file.

The Stage level can contain multiple stages which can be switched between quickly from the same project file, these stages can share the same screens, venues and props.

Each stage you create can share the same screens, venues and props. In the example above there are two different stages in the same d3 file: Stage1 and Stage2. Notice that both of these stages share certain screen types: Screen1, Screen2 and Projector1; they also share specific props: Prop2, Prop4 and Prop5. However, their overall prop configurations are different, hence the need to create two different venues; and their overall screen configurations are different, hence the need to create two different stages.
For information on adding, removing, and editing stages, screens, venues and props please read the following sub-chapters.

**Please note:** For solid performance from disguise hardware it is best to keep the total number of vertex points in your scene below about 150,000-200,000 vertices. This figure varies with the amount of video loading required.
Editing the Stage

Stage editor

To edit the Stage you need to open the **Stage editor**.

To open the stage editor, right-click **stage** from the **dashboard** (bar at the top of the screen).
Stage editor accessed by right-clicking Stage from the dashboard.

Scene

Select the scene tab to display the following properties:

Floor size

This property controls a stage's floor size. There are two numbers laid out horizontally: respectively the x (width) and z (depth) coordinates. The units used for floor size are meters.

Floor position

This property controls a stage's floor position in 3D space. There are three numbers laid out horizontally: respectively the x (left/right), y (up/down) and z (forward/backward) coordinates.

Floor plan

This points to the still image file that defines the floor plan of the currently active stage. Selecting this property will open the Texture object library, which shows all of the still image files saved on your local hard-drive in the DxTexture folder.

To add the floor plan to the stage floor:

1. Left-click floor plan to open the Texture object library.
2. Left-click the image you want to use for the stage floor.

If you want to use a floor plan other than the standard still images provided in the software, you will need to use a custom still image file.

See the Placing media files for a project sub-chapter to understand where to place a custom still image file and how to access it. Also, save the file to a supported file format.
**Venue**

This property enables you to swap a venue with the currently active venue. This switching of venues will be updated in the Stage Visualiser. This is very useful for quickly switching between different prop configurations.

Venue property is used to open the Venue manager

1. Left-click **venue** to open the Venue manager. This will display a list of all of the venues you have created.

2. Left-click the venue you want to switch with the currently active venue.
For more detailed information on using this feature please see the Creating/switching venues, Editing venues, Creating/removing props and Editing props sub-chapters. For general information about venues and props please see the Venues/props overview.

**People**

This property controls the number of human figures displayed in the Stage Visualiser. The maximum number of human figures in a stage is 320.

**Ambient Brightness**

This property controls the ambient brightness level of the stage while in Lux renderer mode.

**Ambient Colour**

This property controls ambient colour of the stage while in Lux renderer mode.

Use the Colour Dropper tool to select an ambient colour for the stage.
Footprints level

This property is a resolution multiplier for projector shadow buffers. 0 disables footprints.

LED Screens

Click the + icon to open the LED screens manager and create new LED screens to add to the stage.
Left click and drag LED screens in the list into the - icon to remove them from the stage.

For more information see screen types overview and creating screens.

DMX Screens

Click the + icon to open the DMX screens manager and create new DMX screens to add to the stage.

Left click and drag DMX screens in the list into the - icon to remove them from the stage.

For more information see screen types overview and creating screens.

DMX Lights

Click the + icon to open the DMX lights manager and create new DMX lights to add to the stage.

Left click and drag DMX lights in the list into the - icon to remove them from the stage.

For more information see screen types overview and creating screens.
Projection Surfaces

Click the + icon to open the Projection Surfaces manager and create new Projection Surfaces to add to the stage.

Left click and drag Projection Surfaces in the list into the - icon to remove them from the stage.

For more information see screen types overview and creating screens.

Projectors

Dynamic Blend

Toggles dynamic blend on and off for projectors, globally.
Blend gamma

Controls the blend gamma for Dynamic Blend.

Blend sharpness

Controls the blend sharpness for Dynamic Blend.

Click the + icon to open the Projector manager and create new Projector to add to the stage.

Left click and drag Projectors in the list into the - icon to remove them from the stage.

For more information see [screen types overview](#) and [creating screens](#).

Projector List...

Clicking this button opens a list of all projectors in the project, with the ability to edit their properties in one editor. This can also be accessed with the shortcut, **CTRL P**.

Reference points...

Clicking this button opens a list of all projector calibration points in the project, with the ability to view the name, parent object as well as toggle the mute, lock & delete functions.

Projector Studies...

Clicking this button opens the Projector Studies editor.

OmniCal Calibration...

Clicking this button opens the OmniCal Calibration editor.
Cameras

The cameras tab allows you to add fixed cameras to the stage that all have their own rendering settings. All cameras can be assigned to feed outputs.

Visualiser camera

The visualiser camera is the camera you are looking through on the local machine. It has its own set of settings, and can be changed by left clicking the property and selecting a different camera. See visualiser renderer for more information.

Camera bookmarks list

Opens the camera bookmarks list.
See storing & recalling camera positions for more information.

Virtual reality navigator

Opens the VR viewfinder.
Actions

Select the Actions tab to display the following properties:

Switch Stage

This property allows you to change the stage displayed in the Stage Visualiser. The currently active stage name will be displayed in the Stage tab of the dashboard as highlighted in the image below.
Switch Stage property is used to open the Stage manager and select a stage

To create a new stage:

1. Left-click **Switch stage** to open the Stage manager. This will display a list of all of the stages you have created.

2. Type the name of a stage into the **new stage** text field and hit **Enter**. This will create a new stage which will be added to the Stage manager.

You can then use the Stage editor to select the stage you want to switch with the currently active stage.

**Export stage**

As of r17.3, the stage can be exported as either FBX, OBJ or 3DS.

1. Enter a name for the export.

2. Append the file extension, such as .fbx .obj or .3ds

3. Click **OK**.

4. The exported file will be written to the output folder of the project folder.

**Import Projectors from table**

This button is used to import projectors via CSV. See **Projector Import** for more information.
**Toggle 3D control**

This button is used to toggle the Manipulators function on and off. It can also be toggled through the axis icon on the track header and with the CTRL M keyboard shortcut.

**Hold/Unhold all outputs**

Hold all outputs is a toggle between hold/unhold. Similar to the existing Hold functionality available in the dashboard, this function allows you to quickly set all screens in the stage to Hold mode which freeze the frame displayed on them, allowing you to continue sequencing without changing their output. This can also be done on a per screen basis as detailed in output properties.
Manipulators

Manipulators allow the user to offset, rotate and scale properties of objects in the stage using mouse input rather than numeric values.

Overview

Manipulators are a feature of the stage added in r17. They allow the user to freely offset, rotate and scale objects such as screens, projectors and props in the stage using a combination of keyboard shortcuts and mouse manipulation. This feature allows for workflows that are more familiar to users from packages such as Blender, 3DS, Cinema4D etc.

Workflow

1. Select items in the stage by left clicking on them. When highlighted, items will receive an orange outline.

2. Use the E, R or T buttons on the keyboard to toggle between offset, rotate and scale modes.

3. Use the on screen gizmo to manipulate the object(s)

Example

1. Select items in the stage by left clicking on them. When highlighted, items will receive an orange outline.

2. Use the E, R or T buttons on the keyboard to toggle between offset, rotate and scale modes.

3. Use the on screen gizmo to manipulate the object(s) by left clicking and dragging on one of the coloured axis handles.
Manipulators shortcuts & actions

Manipulators editor

The Manipulators editor appears as an overlay to the stage when Manipulators are enabled via the icon on the track header, or by clicking **Toggle 3D control** in the actions tab of the stage editor.
Keyboard shortcuts

Once Manipulators are activated, you can then use the following keyboard shortcuts while using the feature:

**Offset (E)**

The offset mode allows you to freely offset x, y or z values using the gizmo in the stage.

**Rotate (R)**

The rotate mode allows you to freely rotate x, y or z values using the gizmo in the stage.

**Scale (T)**

The scale mode allows you to freely scale x, y or z values using the gizmo in the stage.

---

**Please note:** Note that some items, for example Projectors, cannot be scaled.

**Object/Global space toggle (G)**
Global space is the coordinate system for the entire scene. Its origin is at the center of the stage.

Object space is the coordinate system from an object's point of view. The origin of object space is at the object's centroid, and its axes are rotated with the object.

If multiple objects are selected, object space refers to the centroid of all selected objects with the rotation of the most recently selected.
Screen types overview

What is a screen?

A screen in the disguise software consists of exactly as many pixels as there are in the real world screen. Content can be rendered from the Timeline and mapped to the screens in the stage in real time (with the exception of the Projector screen type). The screens can then output their data via the Feed system or DMX depending on the screen type. The disguise software uses DmxScreens to output DMX data, and DmxLights to create DMX-based moving lights and other DMX-based screens.

Types of screens

The disguise software contains five different screen types that can be added to a stage.

Dmx Lights

This is the screen type to use when you want to create DMX-based moving lights and other DMX-based screens where custom DMX drivers are required (that cannot be created using the DmxScreens type, see below). DmxLights also visualises beam effects and movement. Learn more about DmxLights in the DmxScreens/Lights chapter.

Dmx Screens

This screen type is a normal video screen of any shape and form but outputs DMX data instead of video. Learn more about DmxScreens in the DmxScreens/Lights chapter.

Projection Surfaces

This screen type is an object that will be projected onto and should be used in conjunction with a virtual projector. Projection screens have no output feed and require projectors to view them. Learn more about this feature in the Projector Simulation chapter.

LED Screens

This screen type is used for LED screens.

Projectors

This screen type is actually a virtual projector used for projection mapping projects. Learn more about this feature in the Projector Simulation chapter.
In addition, the disguise software contains by default a set of primitive meshes:

- 1 x 1m Rectangle. The name 'Rectangle' may sound confusing as it's actually a 1 x 1 m plane. This plane has been exported from the origin in 3ds Max (0,0,0).

- 1 x 1m Cylinder.

- 1 x 1m Convex.

These three primitive meshes can be used as the screen mesh for the six different screen types (with the exception of Projectors), defining the shape of your screens. They are already UV-mapped and can be re-sized, positioned and configured to match your design requirements. If you are using any other shape than the standard Rectangle, Cylinder or Convex meshes, the screen requires an imported UV-mapped .obj file as the screen mesh. Please see the Mesh section within the Editing screens sub-chapter for information on how to import a screen mesh into the software.
Screen types overview

What is a screen?

A screen in the disguise software consists of exactly as many pixels as there are in the real world screen. Content can be rendered from the Timeline and mapped to the screens in the stage in real time (with the exception of the Projector screen type). The screens can then output their data via the Feed system or DMX depending on the screen type. The disguise software uses DmxScreens to output DMX data, and DmxLights to create DMX-based moving lights and other DMX-based screens.

Types of screens

The disguise software contains five different screen types that can be added to a stage.

Dmx Lights

This is the screen type to use when you want to create DMX-based moving lights and other DMX-based screens where custom DMX drivers are required (that cannot be created using the DmxScreens type, see below). DmxLights also visualises beam effects and movement. Learn more about DmxLights in the DmxScreens/Lights chapter.

Dmx Screens

This screen type is a normal video screen of any shape and form but outputs DMX data instead of video. Learn more about DmxScreens in the DmxScreens/Lights chapter.

Projection Surfaces

This screen type is an object that will be projected onto and should be used in conjunction with a virtual projector. Projection screens have no output feed and require projectors to view them. Learn more about this feature in the Projector Simulation chapter.
LED Screens

This screen type is used for LED screens.

Projectors

This screen type is actually a virtual projector used for projection mapping projects. Learn more about this feature in the Projector Simulation chapter.

In addition, the disguise software contains by default a set of primitive meshes:

- 1 x 1m Rectangle. The name 'Rectangle' may sound confusing as it's actually a 1 x 1 m plane. This plane has been exported from the origin in 3ds Max (0,0,0).

- 1 x 1m Cylinder.

- 1 x 1m Convex.

These three primitive meshes can be used as the screen mesh for the six different screen types (with the exception of Projectors), defining the shape of your screens. They are already UV-mapped and can be re-sized, positioned and configured to match your design requirements. If you are using any other shape than the standard Rectangle, Cylinder or Convex meshes, the screen requires an imported UV-mapped .obj file as the screen mesh. Please see the Mesh section within the Editing screens sub-chapter for information on how to import a screen mesh into the software.
Creating screens

When you create a new project, a projection surface screen and virtual projector are inserted by default. To remove this default screen see removing screens.

Creating a new screen

1. Right-click stage to open the Stage editor. Each type of screen has it's own tab in the stage editor. Listed below each screen type are screens associated to that type. A new project will, by default, contain a Screen called surface 1.
2. Left-click the + button for the type of screen you wish to add. This opens the screen manager for that screen type. In this example we will add a new projection surface.

3. Type the name of the new screen in the **new surface** text box, in this example **Projection Screen**, and hit Enter.
To add a screen left-click the + for the correct type of screen and type a name for the new screen.

The new screen will appear in the list of screens for that screen type. The new screen will be created and added to the currently active stage and the screen editor will open.
New screen listed in the screen list and the screen editor opened.

You can now edit the screen to change its size, position, orientation and shape.
Deleting screens

Permanently deleting a screen

1. Left-click on the + button just below the **screens** list for that type of screen in the Stage editor to open the Screens manager.

2. Left-click and drag the screen you want to delete from the Screens manager to **trash** (represented by a trash-can icon).

   ![surfaces](image)

   Left-click and drag the screen you want to delete from the Screens manager to Trash

   This places the screen in the trash can (deleted items)

To permanently delete the screen out of the trash can:

   Right-click **trash** and select **empty trash** to permanently delete the screen from your stage.

   ![Options](image)
Right-click Trash and select Empty Trash to permanently delete the screen

**Warning:** you can never permanently delete a screen if it's active or referenced from anywhere in the project.
Changing the Screen Type

Changing screen type

1. right-click on the screen editor title bar
2. select the ‘change type’ option
3. select the new type of screen

Changing screen type

Options [Surface]
- Rename: surface 1 OK
- Duplicate: surface 2 OK
- Lock
- Change Type
- Versions

Change Type To
- DmxScreen
- DmxLights
- LedScreen
Removing screens

Removing a screen

Left-click and drag the screen name to the -button below in it's section of the Stage editor.

This will remove the screen from the currently active stage. The screen itself will not be deleted, it is still listed in the ScreensManager. To delete the screen permanently see the deleting screens sub-chapter.

In this example surface 1, has been removed from the currently active stage, but remains in the surfaces manager.
To edit a screen you need to open the **Screen editor**.

To open the Screen editor:

- Right-click a screen directly in the Stage level, or
- Right-click the screen name from the **screens** list in the **Stage editor**.

Screen editor is used to edit a screen.
Screen editor

To edit a screen you need to open the Screen editor.

To open the Screen editor:

- Right-click a screen directly in the Stage level, or
- Right-click the screen name from the screens list in the Stage editor.

Screen editor is used to edit a screen.
Screen properties

Offset

This controls the screen position in the 3D space. There are three numbers laid out horizontally: respectively the x (left/right), y (up/down) and z (into/out of the screen) coordinates.

Rotation

This controls the rotation of the screen mesh, in degrees. The x component controls the rotation around the x axis; the y component around the y axis; and the z component around the z axis. Mesh vertices are rotated around their origin; for this reason, it is recommended that mesh vertices are centered around their origin.

Tracking source

This property defines which automation axis controls the offset & rotation properties of the object. For more information, see Object tracking source.

Scale

The scale property specifies a scaling factor to be applied to the mesh. If the original mesh is a unit mesh (i.e. its x, y and z extents are 1 meter) then the size refers to its actual size. The installed Rectangle, Cylindrical and Concave meshes are examples of unit meshes. The scales z component will have no meaning for flat rectangular screens. For screens with curvature (such as the installed Concave mesh) the z component will scale the z size of the screen.

Resolution

This controls the resolution of the screens physical canvas. If you are using disguise's pre-installed Rectangle mesh as the screen, set the resolution to match the physical dimension (aspect) of the screen.

Mesh

This points to the Mesh file that defines the shape of your screens. Selecting this property will open the Mesh object library, which shows all of the Mesh .obj files saved on your local hard-drive in
the **Mesh** folder. The mesh consists of an array of vertices, each of which specifies a 3D position in space and a 2D texture coordinate that indexes the physical canvas.

To swap a screen mesh with the currently active mesh:

- Left-click **mesh** to open the Mesh object library.
- Left-click the mesh you want to replace the currently active mesh. This will update the screen mesh.

If you are using any other shape than the standard Rectangle, Cylinder or Convex meshes, the screen requires a UV-mapped **.obj** file as the screen mesh.

See the [Placing media files for a project](#) sub-chapter to understand where to place a mesh **.obj** file and how to access it in the disguise software. Also save the file to a [supported file format](#).

To edit a mesh:

- Right-click the mesh inside the Mesh object library to open the Mesh editor. The Mesh editor is used to flip or swap the UV mapping coordinates of a mesh or find out how many vertices a 3D mesh contains.

The chapter [LED Screen examples](#) explains how to create a UV-mapped LED screens.
Appearance properties

Appearance properties control how the screen appears in the visualiser.

**Alpha**

This controls the opacity of the screen. When Alpha is set to full (255), the screen is completely opaque; when Alpha is set to 0, the screen is completely transparent.

*Please note:* the Alpha value only takes effect when the blendMode is set to **alpha**.

**BlendMode**

This controls how the screen contents are composited with the rest of the objects in the stage. Use **over** for nontransparent screens, **add** for screens that are totally invisible when there is no content on them, and **alpha** for screens that are partially transparent.

*Please note:* setting the mode to **over** when either mask contains transparent pixels will still make those pixels see-through.

**Content**

This controls how the screen displays its content in the Stage Visualiser. If you set it to **front**, only the front surfaces of the mesh will show content. If you use **back**, only the back surface will show content. If
you select **both**, both surfaces will display content.

**Brightness (NITS)**
Sets the brightness in NITS. This value affects visualisation only.

**Viewing angle**
Angle in degrees where luminance is at 50%. Visualisation in Lux only. Setting a value of 0 will disable the visualisation.

**Reflectivity**
Proportion of incident light returned by the screen surface.
Pixel mask

This points to the still image file that defines the Pixel mask. This property lets you specify an image mask that is applied to the screen per physical pixel and thus allows you to create specific looks for individual pixels. For instance, if your pixels are circular, your mask should consist of a circle.

Circular Glow Opaque pixel mask applied to a screen

The disguise software comes with a number of pre-installed Pixel masks for simulating LED Stealth screens. Selecting this property will open the Texture object library, which shows all of the Texture still image files saved on your local hard-drive in the DxTexture folder. Pixel masks can also have alpha channels, allowing you to model other seethrough LED technologies.
Creating a Pixel mask

- Create a 36 x 36px .png image. We recommend using Adobe Photoshop for this.
- Make the transparent areas in the screen transparent in the Pixel mask.
- Make the illuminating areas in the screen as a white area in the Pixel mask.
- Make the black areas in the screen to black areas in the Pixel mask.

See the Placing media files for a project sub-chapter to understand where to place a still image file and how to access it in the disguise software. Also save the file to a supported file format.
The Pixel mask will be applied onto every pixel of the screen (the number of pixels is defined by the resolution of the screen).
Population mask

This points to the still image file that defines the Population mask. The population mask allows you to quickly specify which pixels in the screen are present and which are absent. Selecting this property will open the Texture object library, which shows all of the Texture still image files saved on your local hard-drive in the DxTexture folder. However, before placing a Population mask you will need to create it.

How to create a Population mask

Create a .png image. We recommend using Adobe Photoshop for this. The Population mask should have the same resolution as the screen.
Population mask being created in Adobe Photoshop

- Make the areas you want to be see-through in the screen the alpha channel in the Population mask.

- Make the areas you want to be illuminating in the screen white in the Population mask.

- Make the areas you want to be black in the screen black in the Population mask.

See the Placing media files for a project sub-chapter to understand where to place a still image file and how to access it in the disguise software. Also save the file to a supported file format.
Hierarchy

Parenting objects such as screens, props, and projectors.

It is possible to make an object such as a screen, prop or projector the child of another object. This links the objects together, meaning that if you change the offset and rotation properties of the Parent object, the Child object will also be affected.

How to add a child to a screen

1. Right-click the screen you want to turn into a parent screen, in this example mother screen. This will open the Screen editor. Notice that stage 1 is the parent of the Mother screen.

2. Under the hierarchy tab, left-click Add Child. This will open up a list of all screens and props in your Stage.
List of potential child screens and props opened by left-clicking Add Child from the Screen editor

3. Left-click the screens you want to turn into children of the Parent screen. In this example we have added child screen 1 and child screen 2 as children of the Parent screen mother screen 1.

Please note: After adding a child to the parent screen you will see that the child screen position and rotation have changed. This is because its offset position and rotation is now calculated from the pivot point of their parent screen rather than calculated from the centre of the disguise Stage.

How to parent the child back to the Stage

Under the hierarchy tab under the the child screens editor, left-click Parent to Stage. This will bring the child screen back to be parented to the Stage, in this example stage 1.
Parent to Stage property of the Child Screen editor being used to return the Child screens parent back to Stage 1

Note: A Parent screen can have multiple child screens and props.
Output properties

Output properties control additional properties that affect a screen's output.

Render layer

The render layer feature allows you to define where the screen is rendered.

On Stage - the screen is rendered in the camera & projector view ports.

Off Stage - the screen is in the stage, but not rendered at all.

Frontplate (AR) - the screen is in the stage, but only rendered to the camera frontplate.

Master fade

This controls the output level of the screen when sent to the Output Feeds level. It ranges from 0 to 1. Use this to reduce the brightness of a physical screen to compensate for cameras or other less bright fixtures. Reducing Master Fade does not affect the brightness of the screen content in the Stage level.

Colourshift

Each Screen has a Colourshift property which can be used to control the colour balance of the screen when sent to the Output Feeds level. This can be used to compensate for different colour and brightness characteristics of different LED technologies, allowing you to output consistent colour and brightness when using a mix of different technologies. This parameter does not take effect in the visualiser, as it is used exclusively to fix display hardware inconsistencies. For a full explanation on the Colourshift property please see the Colour Shift section of the Common layer properties topic.
Colour profile

Allows you to specify a Colour profile for a screen.

Colour LUT

Allows for a LUT file to be applied to the screen. This is useful if you need to change the colour profile of the screen. We recommend you do this in your LED screen processor or onboard screen hardware, but an option do to it via a LUT exists for circumstances this may not be possible or for other creative uses.

Projector View Blending

Allows for Alpha blending in the projector view. Note that the projector surface blend most must be set to Alpha for this to work.

Hold output

The hold output function allows the user to freeze the output of a screen similar to the project level Hold function. This allows the user to continue sequencing but hold the frame of output on one or a selection of screens.
DMX screens and DMX lights overview

Background

The disguise software DMX functionality was motivated by the need to control RGB LED pixel-based fixtures that were DMX-driven. In this view, a DMX-controlled pixel is no different from a DVI-controlled pixel; it is simply controlled using a different type of signal. This led to the implementation of the DMXScreen, which is identical to the normal screen type except that it outputs the RGB information via DMX instead of via the Feed system. For more information on the different screen types please see the Screen types overview sub-chapter.

Over time, this idea of lights being simply pixels extended to the control of traditional lighting fixtures, with the conversion from RGB to CMY colour spaces and the addition of moving head bases. This led to the creation of the Fixture System, which allows more complex fixtures to be modeled.

Using the disguise software for controlling DMX-fixtures is recommended for tasks involving highly sculptural lighting arrangements where the individual lights/pixels are being looked at, rather than lighting another surface. The disguise software is particularly suited for RGB LED pixel fixtures, since these have no personalities as such and therefore have zero setup time.

However, disguise does not replace traditional lighting desks where typical tasks such as lighting a performer or surface is a requirement.

Current status

At the time of writing, the disguise software has the following limitations when dealing with DMX fixtures:

- The visualiser shows light beams but does not model how light falls on surfaces or smoke/fog.

- Shadows are not visualised.

- CMYK fixtures can be driven, as can RGB, RGB+Amber and RGB+white fixtures. However, fixtures that are controlled via colourwheel are not supported.

- the disguise software does not have a fixture personality library. This means that when using disguise to drive a new fixture, you need to create and validate a fixture personality (the disguise term is Fixture Driver) that specifies which channel numbers have which functions. This is usually
a simple process; however, it can take some time to get the driver working correctly and this time must be calculated into the project planning.

Moving head fixtures are supported via the Target layer and Target2 layer. These allow groups of lights to be aimed at points in stage space. However, the disguise software does not provide as much control over aiming as a standard lighting desk would do.

While the disguise software can control non-colour parameters such as iris, gobo, focus and zoom, a lighting desk is still recommended for controlling such parameters.
DMX screens and DMX lights overview

Background

The disguise software DMX functionality was motivated by the need to control RGB LED pixel-based fixtures that were DMX-driven. In this view, a DMX-controlled pixel is no different from a DVI-controlled pixel; it is simply controlled using a different type of signal. This led to the implementation of the DMXScreen, which is identical to the normal screen type except that it outputs the RGB information via DMX instead of via the Feed system. For more information on the different screen types please see the Screen types overview sub-chapter.

Over time, this idea of lights being simply pixels extended to the control of traditional lighting fixtures, with the conversion from RGB to CMY colour spaces and the addition of moving head bases. This led to the creation of the Fixture System, which allows more complex fixtures to be modeled.

Using the disguise software for controlling DMX-fixtures is recommended for tasks involving highly sculptural lighting arrangements where the individual lights/pixels are being looked at, rather than lighting another surface. The disguise software is particularly suited for RGB LED pixel fixtures, since these have no personalities as such and therefore have zero setup time.

However, disguise does not replace traditional lighting desks where typical tasks such as lighting a performer or surface is a requirement.

Current status

At the time of writing, the disguise software has the following limitations when dealing with DMX fixtures:

- The visualiser shows light beams but does not model how light falls on surfaces or smoke/fog.

- Shadows are not visualised.

- CMYK fixtures can be driven, as can RGB, RGB+Amber and RGB+white fixtures. However, fixtures that are controlled via colourwheel are not supported.

- the disguise software does not have a fixture personality library. This means that when using disguise to drive a new fixture, you need to create and validate a fixture personality (the disguise term is Fixture Driver) that specifies which channel numbers have which functions. This is usually
a simple process; however, it can take some time to get the driver working correctly and this time must be calculated into the project planning.

- Moving head fixtures are supported via the Target layer and Target2 layer. These allow groups of lights to be aimed at points in stage space. However, the disguise software does not provide as much control over aiming as a standard lighting desk would do.

- While the disguise software can control non-colour parameters such as iris, gobo, focus and zoom, a lighting desk is still recommended for controlling such parameters.
Global DMX options

There are four global switches that control how the disguise software outputs DMX information. They are found under the DMX tab of the Program Settings menu.

To access the global DMX properties, rightclick d3 from the dashboard (bar at the top of the screen) and open DMX tab.

Global DMX properties

All lamps on

- Left-click All lamps on to send lamp on commands to all active fixtures in the stage.

All lamps off

- Left-click All lamps off to send lamp off commands to all active fixtures in the stage.
Please note: some fixtures may have a mandatory cooling off period (say 5 minutes) before they can be turned off again.

**Fade inactive DMX fixtures to black**

This option controls what happens when no layers on the Timeline are writing to fixtures. Selecting **yes** (default value) will enable the disguise software to send black; selecting **no** will enable d3 to send nothing.

**Force resend of static DMX values**

This option controls whether or not the disguise software sends DMX values when they change. Selecting **no** (default value) will enable the disguise software to only send DMX values when they change; selecting **yes** will enable DMX values to be sent constantly, regardless of whether the value is changing or not. This setting can be useful when merging disguise's DMX information with other DMX information, for example coming from a lighting desk.
Creating DMX lights

DmxLights are screens to which a rectangular array of fixtures has been attached. The fixtures are spaced out equally across the surface of the screen and sample content from the screen before sending it out via DMX.

You should use a DmxLights screen when you are trying to control an array of complex fixtures. Complex fixtures are those which have any or all of the following:

- Shutter open/close, lamp on/off, reset and other commands
- CMY colour mixers with a traditional lamp
- Extra control channels such as focus, zoom, iris, gobo
- A moving head base with pan and tilt

Creating a DmxLights screen

The DmxLights screen is added in the same way as any other screen type. Please see the sub-chapter Creating/removing screens for step-by-step instructions on how to add a screen to a stage, and select DmxLights from the menu of different screen types.

DmxLights properties

The DmxLights properties are similar to the Screen properties, but with an extra tab called addressing (explained in the sub-chapter DMX addressing) and fixtures. Therefore all of the DmxLights properties not related to addressing or fixtures can be read in the sub-chapter Editing screens.
DmxLights has the same properties as a Screen but with two extra sections called Addressing and Fixtures.

An explanation of each of the properties contained in the **fixtures** section are explained in the following sub-chapter Creating a fixture, and the sub-chapter Setting up a grid of fixtures.
Creating DMX lights

DmxLights are screens to which a rectangular array of fixtures has been attached. The fixtures are spaced out equally across the surface of the screen and sample content from the screen before sending it out via DMX.

You should use a DmxLights screen when you are trying to control an array of complex fixtures. Complex fixtures are those which have any or all of the following:

- Shutter open/close, lamp on/off, reset and other commands
- CMY colour mixers with a traditional lamp
- Extra control channels such as focus, zoom, iris, gobo
- A moving head base with pan and tilt

Creating a DmxLights screen

The DmxLights screen is added in the same way as any other screen type. Please see the sub-chapter Creating/removing screens for step-by-step instructions on how to add a screen to a stage, and select DmxLights from the menu of different screen types.

DmxLights properties

The DmxLights properties are similar to the Screen properties, but with an extra tab called addressing (explained in the sub-chapter DMX addressing) and fixtures. Therefore all of the DmxLights properties not related to addressing or fixtures can be read in the sub-chapter Editing screens.
DmxLights has the same properties as a Screen but with two extra sections called Addressing and Fixtures.

An explanation of each of the properties contained in the **fixtures** section are explained in the following sub-chapter [Creating a fixture](#), and the sub-chapter [Setting up a grid of fixtures](#).
Creating a fixture

When you first create a DmxLights screen, it is just an ordinary screen. You can drive content to it and see
the content in the Stage Visualiser, but you cannot see the screen in the Feed level.

The first step is to set up the fixture the DmxLights screen will use to cover its surface. For information on
how to create a DmxLights screen please read the previous sub-chapter DmxLights.

To create a fixture:

1. Left-click **fixture** from the DmxLights editor. This will open the Fixtures manager.
2. Type in the name of the fixture in the **new fixture** text field, in this example **mac 2000**.
3. Hit **Enter** to open a list of fixture types.
4. Left-click the type **Fixture**.

![Fixtures editor]

Process used to create a fixture, in this example Mac 2000

5. Hit **Enter** to create the new fixture. For information on how to create a grid of fixtures please see
the following sub-chapter **Setting up a grid of fixtures**.
Fixture editor, in this example for the newly created Mac 2000 fixture

6. Hitting **Enter** will also open the Fixture editor.

**Please note:** the fixture object defines a fixture within the Stage Visualiser. It does not set up information about how to actually drive the fixture; that is handled by the **driver** property.

**Fixture properties**

**Size**

This specifies the physical size of the fixture, in meters. The first number is the x component (width), the second is the y component (height) and the third is the z component (depth). These numbers refer to the size of the actual fixture emitter, not to the total size of the fixture base.

**Resolution**

Each fixture is considered to be a mini screen containing one or more pixels. The default resolution is 1 by 1 pixel (i.e. a single pixel) but it is possible to have multiple pixels in the same fixture. For example, a Traxon tile has a grid of 8 by 8 pixels; a Robe Redwash has a grid of 4 by 1 pixels.

**Aimable**

Set this to **yes** if the fixture has a moving head, otherwise select **no**.
Aimable property should be set to yes if the fixture has a moving head.

**Number of Channels**

Specifies the number of DMX channels in the fixtures personality.

**Mask**

This points to the still image file that defines the mask used when rendering the fixture. Selecting this property will open the Texture object library, which shows all of the Texture still image files saved on your local hard-drive in the *DxTexture* folder. By default d3 comes with a number of still image files. For example, fixtures with circular emitter profiles should use the *circlemask* bitmap.

Mask property is used to select a still image file that defines the mask used when rendering the fixture.

The instructions on how to create a custom Pixel mask can be used to create a mask for rendering a fixture. For step-by-step instructions on how to create a Pixel mask, where to place a still image file, and how to access it in d3, scroll down to the section Pixel mask in the *Editing screens* topic.

**Beam Angle**

These control the max (zoomed out) and min (zoomed in) beam angles of the light, in degrees.
**Beam Length**

This sets the beam length, in meters. This property is for visualisation purposes only; it does not send a signal to the light.

**Drivers**

This specifies how colour and orientation information are sent to the actual physical fixture. For detailed information on how to use this property please see the [Fixture drivers topic](#).
Fixture drivers

The final property of a fixture, driver, specifies how colour and orientation information are sent to the actual physical fixture. Please scroll down to the section 'Fixture properties in the Creating a fixture sub-chapter for information on the other fixture properties.

Driver property of a fixture specifies how colour and orientation information are sent to the actual physical fixture.

This is sometimes referred to as the personality of the fixture. It is a list of functions that are supported by the fixture and the channels used to control those functions. This information is usually found in the product manual for the fixture, which is downloadable from the manufacturers website.

You can sequence a complete show in the visualiser without creating fixture drivers. However, a fixture must have a valid driver in order to be assigned a DMX address (channel/universe) using the addressing system and in order to send DMX to the outside world.

Types of fixture drivers

There are four types of fixture drivers available, each suitable for a different class of fixtures.
DmxDriver

This is the easiest type. Use this for DMXcontrolled pixelgrid fixtures that have no moving head or CMY colour mixing hardware.

FixtureDriver

Please ignore this type of fixture driver.

GenericLampDriver

This driver is suitable for traditional moving head lights containing (optionally) a pan and tilt base and (also optionally) a CMY (cyan/magenta/yellow) colour mixing system. It supports programming commands such as lamp on/off, shutter on/off, etc.

MovingHeadLED_driver

This driver combines elements of DmxDriver and GenericLampDriver and is suitable for moving head fixtures that have multiple LED pixels mounted on the moving head.

Creating a fixture driver

To create a fixture driver:

1. Open the Fixture editor by right-clicking fixture, in this example mac 2000 , from the DmxLights editor.

2. Open the Fixturedrivers manager by left-clicking driver from the Fixture editor.

3. Type in the name of the new fixture driver, in this example bb4 driver, in the new fixturedriver text field.

4. Hit Enter to create the new fixture driver.
5. Select a fixture driver type from the list of driver types.

DmxDriver

The DmxDriver is a fixture driver type that routes RGB pixel information to a regular rectangular grid of LED pixels.
DmxDriver properties:

**Mode**

This describes the pixel type of the LED fixture. There are two monochrome modes: *mono [weighted sum]* gives a more natural result for photographic/gradient images, whereas *mono [max signal]* ensures that peak brightness is maintained. Sending a redonly image with all levels at 255 will result in an output level of 255, whereas the weighted result will be less than 255.

The options are:

- **rgb**: traditional three component RGB LED, with eight bits per signal (red, green and blue).
- **rgb + amber**: RGB + an extra amber component, all at eight bits per signal.
- **rgb + white**: RGB + an extra white component, all at eight bits per signal.
- **rgb + amber + white**: RGB + amber and white components, all at eight bits per signal.
- **cmy**: cyan, magenta and yellow components.
- **mono [weighted sum]**: monochrome (8 bits); the value sent is a weighted sum of red, green and blue.
- **mono [max signal]**: monochrome (8 bits); the value sent is the maximum of red, green and blue.

**Horizontal**

This property allows the data sent to the tile to be mirrored in the horizontal axis.

**Vertical**

This property allows the data sent to the tile to be mirrored in the vertical axis.

**Rotation**

This property allows image data being sent to be rotated by 90 degrees clockwise or 90 degrees counterclockwise. This only has an effect if the grid is a square grid.
Amber Level, White Level

These properties allow you to directly control the amber and white signals sent to all pixels. They only have an effect if the mode property is set to RGB+amber or RGB+amber+white. In RGB+white mode, the signal sent to the white LED is computed from the RGB signals.

GenericLampDrivers

To set up a GenericLampDriver object, you need to start with the DMX channel specification of the fixture, usually downloadable as a PDF from the manufacturer. You then read down the channel list, filling in channel numbers into the GenericLampDriver as you go.

Most properties specify either a DMX channel number (which is always taken as an index from the first channel in the fixture, 1 being the first valid channel). If a property is set to 0, no data is sent for that property. For instance, if a light is a CMY fixture (cyan, magenta, yellow), but the cyan channel is set to 0, then information will only be sent to the magenta and yellow channels.

The properties are divided into three major sections: lamp, which controls how information is sent to the lamp; pan/tilt, which controls how rotation commands are sent to moving head fixtures; and commands, which allows for setting default parameters.

Mode

This specifies the colour mixing mode for the lamp. This is initialised to cmy.

Since GenericLampDrivers are only meant for singlelamp/pixel fixtures, the mirror/rotate properties from the DmxDrivers are unnecessary and are therefore removed. You can, however, change this to another value if you wish; this allows the GenericLampDriver to be used for singlepixel movinghead RGB LED fixtures.

Lamp controls

This set of properties controls how information is sent to the lamp.

Lamp properties for a GenericLampDriver

Dimmer channel: standard CMY lights allow you to vary the output brightness using a dimmer. This is a mechanical shutter that blocks more or less light as you change the dimmer setting. This property
sets the channel number assigned to the dimmer; if set to 0 (as it should be for RGB LED fixtures), no information is sent.

**Dimmer flipMode** : some fixtures use the value '0 to signify dimmer closed (i.e. minimum light output); others use this value to signify dimmer open (i.e. maximum light output). This switch allows you to change the fixtures behaviour as appropriate.

**Min brightness** : dimmers, being mechanical, do not always produce a linear response. In particular, some dimmers appear to block all light until the value rises to a non0 value and then open up from there. If the fixtures dimmer exhibits this feature, set min brightness to the value at which output brightness reaches 0 as you dim the light. From then on, the disguise software will scale the output brightness appropriately to give you a clean, linear response.

**Cyan, magenta, yellow channels** : these properties apply to lights that use CMY colour generators. They are the channel numbers within the fixture of the cyan, magenta and yellow channels, respectively. These should be set to 0 for RGB fixtures.

**CMYK flipMode** : if it turns out that your lighting fixture views the world backwards, flip this switch and the CMYK values sent will be inverted, i.e. 0 will be most colour, 255 will be least colour.

**Red, green, blue channels** : these properties apply to lights for which the mode property (above) is set to rgb. They are the channel numbers of the red, green and blue components respectively. These should be set to 0 for CMY fixtures.

**Zoom channel** : if the light has the ability to zoom its beam (make it narrower and wider), enter the channel number of the zoom control here; otherwise, leave it at 0.

**Focus channel** : if the light has the ability to focus its spot so it appears hard edged at different distances, enter the channel number of the focus control here; otherwise, leave it at 0.

**Gobo channel** : if the light has the ability to change its gobo dynamically, enter the number of the gobo control channel here; otherwise, leave it at zero.

**Lamp on, lamp off** : these are DMX commands (see section ' DMX commands below) to switch the fixtures lamp on or off. Fixtures often require you to send a DMX value continuously for a number of seconds to fire the command. The DMX commands gives you the ability to do this.

**Open shutter, close shutter** : many fixtures have a shutter that has two states: open (all light passes through) or closed (no light passes through). Shutter response is usually much faster than dimmer
response, so the disguise software will use the shutter whenever a fast cut to black is required. The open/close shutter DMX commands (see next section) allow you to represent this.

**DMX commands**

Many lamp or fixture commands require you to send a particular value on a particular channel for some number of seconds. For example, the Martin Mac2000 will switch off its lamp if you send the value 250 on channel 1 for 5 seconds.

**Please note:** it is possible to add DMX commands and animate their values using the DmxControl layer.

To create a DMX command:

1. Open the DmxCommands manager by left-clicking the **lamp on**, **lamp off**, **open shutter** or **close shutter** properties.

2. Type in the name of your DMX command, in this example **frost**, in the **new dmxcommand** text field.

3. Hit **Enter**. A DmxCommand editor will open.

**Channel**: the channel number to send the command on. As with all other channel numbers, this is local to the fixture, with the lowest valid channel being 1. If channel is 0, no data will be sent. You can read the channel number directly from the fixtures DMX chart.

**Value**: this is the value to be sent on the channel in order to actuate the command. Permissible values are 0 to 255 inclusive. Some commands work with a range of numbers, eg. 248-255. It is usually best to pick a value near the middle of the range, i.e. 250 in this case.

**Time**: this is the duration that the value should be sent to actuate the command, in seconds. Once the command is actuated, disguise will continue to send the value on the channel for this number of seconds; a command cannot be cancelled once it has been actuated. It is possible for this number to be set to 0. If this is the case, the disguise software will send the value once only.
**Min/max value**: allows the value to be scaled to any range. **min** specifies the lower value in the range, **max** specifies the upper number in the range.

**Pan/tilt controls**

This set of properties specifies the pan/tilt behaviour of the fixture. If the fixture does not have a moving head base, just leave these values at 0.

**Pan/pan fine**: most moving head fixtures allow the control of angles via a 16bit number, split into two parts and sent on two separate channels. These two properties specify those channels. As with all other channels, the lowest permissible value is 1. You can read the values directly off the fixture DMX chart.

**Pan range**: this specifies the range of travel in the pan axis (rotation around the vertical axis), in degrees. The range can be found in the fixtures manual. If, for example, the manual states a range of 270 to +270, the pan range should be set to \(270 \times 2 = 540\) (you can always use disguises built in calculator rather than working it out yourself). You can also set this property to a negative number; this will cause the fixture to flip its rotation direction.

**Tilt/tilt fine**: this specifies the channel numbers for the tilt control and the fine tilt control respectively. As with pan and pan fine, you can read these directly from the manual.

**Tilt min/max**: while *pan range* is usually symmetrical around the central axis, the tilt range may differ from front (positive values of tilt) to back (negative values). Therefore, disguise allows you to specify the minimum and maximum tilt values in degrees. It is permissible for tilt min to be higher than tilt max; if this is the case, the tilt direction response is reversed.

**Commands**

This set of properties allows for setting default parameters.

Commands properties for a GenericLampDriver

**Reset**: this is a DMX command that is issued to reset the fixture.

To reset all stage fixtures:
1. Open the Program Settings menu by right-clicking \textbf{d3} from the dashboard (bar at the top of the screen).

2. Select \textbf{reset all DMX fixtures}.

\textbf{Default} : this is an array of DMX commands that are sent every frame, regardless of what other data is being sent. This is because fixtures often have a number of extra controls that the disguise software does not drive directly, such as pan/tilt speed, gobo speed, strobing, secondary gobo or colour wheel controls, etc. If these are left unset, they can end up with values that interfere with normal operation. For this reason, it is advised that you create a DMX command for each extra fixture parameter, name it appropriately, set the default value to an appropriate value, and add it to the default array. This will ensure smooth operation of the fixture as a pixel.

\textbf{Testing}

Testing property used to test a fixture driver with a real example of a fixture

\textbf{Tester} : the final property of the driver is \textbf{LampTester}. This is a helper designed to help you test the fixture driver with a real example of a fixture, without going to the trouble of addressing the complete grid of fixtures.

- Right-click \textbf{tester} to open the LampTester editor.

To use the \textbf{LampTester} property, plug a single fixture into the disguise software and then modify the properties contained in the LampTester editor. While the LampTester editor is open, all other DMX signals (from the grid of fixtures, or from other DmxScreens) are suppressed.

\textbf{Test channel} : this is the base channel number of the fixture (1 is the lowest valid number). Ensure that this number is the same channel number as you have assigned to the fixture.

\textbf{Test universe} : this is the universe number you have assigned to the fixture (1 is the lowest valid universe).

\textbf{Please note} : universe numbers refer to a physical DMX cable. They are not assigned on the fixture the way channel numbers are.
**Lamp**: this property allows you to test that you have the right lamp on/off commands. At the start, the state is set to **unknown**, as the tester has not sent any on/off commands and therefore does not know whether the lamp is on or off. When you set the property to **on** or **off**, the lamp on/off command is sent to the fixture. If you have the command filled in correctly, the lamp should turn on and off. If not, you need to check the command against the fixture manual.

**Shutter**: this property lets you test the shutter open/close command. Set the option switch to **open** or **closed**; the lamp shutter should immediately respond.

**Test intensity**: this property lets you test the dimmer command. Change this number to a value between 0 (dark) and 255 (bright). The value you specify is sent directly to the dimmer, without mediation by the dimmer **min brightness** value (although the **dimmer flipMode** is taken into account).

**Test colour**: this property lets you send a test colour to the fixture. The colour you select is converted to RGB or CMY (depending on the fixture type), the **dimmer flipMode** mode is taken into account, and the data is sent to the fixture. If you have the channel numbers correct, the fixture should reproduce the colour you specify.

Colour Picker window is used to specify a test colour to the fixture, opened by left-clicking the Test Colour property

**Pan**: this property lets you directly send a pan angle to the fixture (in normal operation, pan and tilt are computer based on target points). 0 degrees should point the fixture towards the front of the fixture base, with negative values turning to the left and positive values to the right. The front of the fixture is defined (by convention) to be the side to the right of the small LED control panel. The procedure is to send an angle (eg. 90 degrees) and check that the fixture does indeed turn to that angle. If it does not, you may have to adjust either the pan channel numbers (if there is no movement) or the pan range.

**Tilt**: this property lets you directly send a tilt angle to the fixture. When pan is set to 0, positive tilt values should tilt the fixture lamp towards the front of the fixture; negative values should tilt it towards the back. Setting tilt to 0 should point the fixture directly upwards. If the fixture fails to move when you change the tilt value, you have incorrect tilt channel number settings; if the fixture angle does not match the angle you enter, you will need to edit the tilt min and max values in the driver.

**MovingHeadLedDrivers**

The MovingHeadLedDriver controls fixtures consisting of a moving head base and a grid of LED pixels. It therefore combines the standard DmxDriver controls with the **pan/tilt** and **commands** properties from the GenericLampDriver. The only extra property of note is **first pixel channel**.
MovingHeadLedDriver properties

**First pixel channel**

This sets the channel (relative to the start of the fixtures channel; 1 is the lowest valid number) of the first pixel in the array.

First Pixel Channel property sets the channel of the first pixel in the array.
Setting up a grid of fixtures

To set up a fixture please see the previous sub-chapter Creating a fixture.

Now that you have set up the fixture, you can begin playing with it in the Stage Visualiser. Initially, the DmxLights screen holds only one fixture; if you send content to the Dmx Lights (for example, by targeting a Colour layer), you will see a single beam of colour coming from the centre of the screen surface (assuming you have set a nonzero beam length).

You can now increase the number of fixtures by editing the properties contained in the fixture section of the DmxLights screen editor.

DmxLights properties

Please note: this sub-chapter explains the DmxLights properties relevant to setting up a grid of fixtures. For an explanation of the other DmxLight properties not mentioned in this sub-chapter please see the earlier topic DmxLights.

Number Across, Number Down

This controls the number of fixtures in the grid attached to the screen surface.
Number Across and Number Down properties control the number of fixtures in the grid attached to a screens surface.

As you increase these values, you should see the number of fixture beams increase (assuming there is content going to the screen). **Number Across** increases the number of fixture beams horizontally, and **Number Down** increases the number of fixture beams vertically.

**Fixture Mask**

This controls the mask used when rendering the fixture.

![Fixture Mask](image)

Fixture Mask property is used to select a still image file that defines the mask used when rendering the fixture.

Please scroll down to the explanation of the **mask** property in the previous sub-chapter Creating a fixture for information on how to use the **fixtureMask** property.

**Positioning**

This controls how the positions of the individual fixtures are generated.

![Positioning](image)

Positioning property controls how the positions of the individual fixtures are generated.

There are two options: **physical** spaces the fixtures out at equal physical separations and samples the underlying content accordingly, **pixelperfect** spaces the fixtures out so that they sample the underlying content at regular intervals and positions them at the appropriate physical locations.

**Use normals**

This controls how the orientation of each fixture instance depends on the screens surface.
Use Normals property controls how the orientation of each fixture instance depends on the screens surface, in this example the fixtures are inverted.

There are three options. If set to no, then the orientation of each fixture does not depend on the screen surface. If set to yes, each fixture is oriented so that it is parallel to the screen surface. If set to inverted, a negative version of the surface normal is used, flipping the fixture upside down relative to the surface.

**Global rotation**

This allows you to rotate the fixtures relative to their standard orientation.

Rightclick global rotation to open a local (untitled) FixtureRotation editor. rotation controls the rotation of the fixtures, in degrees. The x component controls the rotation around the x axis; the y component around the y axis; and the z component around the z axis. offset controls the screen position in the 3D space, in metres. There are three numbers laid out horizontally: respectively the x (left/right), y (up/down) and z (into/out of the screen) coordinates.

Please note: you will not see the effect of this rotation until you visualise the fixture bases. For more information please see the DMX addressing sub-chapter.

Once you see the fixture beams it is possible to just visualise the lights.
Switch off the underlying screen content and just visualise the lights by setting the `alpha` property to 0. This property is contained in the `appearance` section of the DmxLights editor.

Alpha property set to 0 to switch off the underlying screen content
**DMX addressing**

Addressing is the process of assigning a unique DMX channel and universe number to each fixture in the DmxLights screen. There are two ways this can work: either you specify the addressing layout in the disguise software and print out a screenshot for the technicians to work to, or you receive an addressing layout chart and have to match it within the disguise software.

Addressing is handled under the **addressing** tab of the DmxLights editor.

<table>
<thead>
<tr>
<th></th>
<th>mac 2000</th>
</tr>
</thead>
<tbody>
<tr>
<td>Offset</td>
<td>0, 0, 0</td>
</tr>
<tr>
<td>Rotation</td>
<td>0, 0, 0</td>
</tr>
<tr>
<td>Scale</td>
<td>1, 1, 1</td>
</tr>
<tr>
<td>Resolution</td>
<td>256, 256</td>
</tr>
<tr>
<td>Mesh</td>
<td>rectangle</td>
</tr>
</tbody>
</table>

Addressing properties of the DmxLights editor control how DMX channels and universe numbers are assigned to each fixture.

**Addressing properties**

**Assignment**

The **DmxAssigner** specifies a set of addressing strings. These strings are used to assign addresses to the fixtures.
To open the DmxAssigner right-click assignment.

DmxAssigner is used to control how addresses are assigned to fixtures, opened by right-clicking on Assignment from the DmxLights editor.

The DmxAssigner contains an array of DmxStrings. Each DmxString specifies a string of fixtures that have contiguous channel numbers.

**Active**

This indicates whether the DmxLights screen is actually outputting DMX. By default, this is set to off; you should only switch it to on once the addressing is checked and complete.

**Show channels**

This controls whether the fixture bases and channel numbers are displayed in the visualiser.

Show Channels property controls whether channel numbers and fixture bases are displayed for each fixture in the Stage Visualiser.

Switching show channels to yes will display each fixture in the visualiser.

**DmxAssigner**

When wiring DMX fixtures, the usual practice is to arrange fixtures in strings, i.e. a series of fixtures wired in daisychain fashion, with channel numbers that step up in a certain way. A DmxLights screen may contain multiple strings. Each of these strings is specified using a DmxString.

To add a DmxString to the DmxAssigner:

1. Right-click + to open the Dmxstrings manager.
2. Type the string name into the new dmxstring text field, in this example Mac2000 String 1 (be sure to include the name of the DmxLights screen in the name).
Process used to create a new DMX string, in this example Mygroup_str1

3. Hit **Enter**. This creates the new DmxString, adds it to the DmxAssigner, and opens the DmxString editor.

DmxString editor

4. Type in the properties you want and hit **assign!**. The DmxString will then assign the channel numbers, and the visualiser will display the channel numbers.

Assigning the channel numbers to the fixtures by left-clicking Assign!

**DmxString properties**

The DmxString starts at a particular fixture position in the grid and then steps across the grid, assigning channel numbers as it goes. The properties of the DmxString editor control which channels are assigned and the step direction.
**Universe**

This specifies the universe number that should be assigned to the first object in the string. The lowest valid number is 1.

**Channel**

This specifies the channel number within the universe that should be assigned to the first object in the string. Valid numbers are 1 to 512 inclusive.

**Number of Units**

This specifies the number of fixtures you want in your string. In the example above, our DmxLights screen has a 2 by 4 grid of fixtures, so the number of units is set to $2 \times 4 = 8$.

**Start position**

This specifies the index within the DmxLights screen of the first fixture in the string. The topleft fixture is designated (0,0), the fixture to its right is (1,0) and the fixture below it is (0,1). In general, you can find fixture (i, j) by starting at the top left fixture and counting (i) to the right and (j) down. In this example, we set start pos to (0,0) to indicate the top left fixture.

**Direction**

Starting at the first fixture, the DmxString will assign a channel, then step in a certain direction across the surface of the DmxLights screen before assigning the next channel number. There are four possibilities: left , right , up or down . In this example, the channel number are stepping to the right.

**Order**

This specifies the stepping order across the grid as a whole. You can select one of two options. Choosing normal means that when it reaches the edge of the grid, it moves up or down one row/column and starts again at the opposite edge. Selecting alternating means that when it reaches the edge of the grid, it moves up or down one row/column and then starts stepping in the other direction.

**Minor Direction**

This controls the direction of the step that takes place when the assigner gets to the edge of the grid. There are two options : ve (steps left or up) or +ve (steps right or down).

**Universe step**

Since fixtures cannot straddle a universe boundary, the assignment algorithm will step up a universe if it finds that there are not enough channels left in the current universe. However, it is sometimes more useful if each row or column within the grid is constrained to be contained within one universe. If this is the case, set universe step to row/column instead of unit.
Targeting fixtures

The Target2 layer and Target layer are used to change the orientation of moving head fixtures. The advantage of the Target2 layer is that it can edit multiple fixture targets, whereas the Target layer can only edit a single fixture target.

How to target the fixtures

1. Create a Target2 or Target layer depending on the number of fixture targets you wish to edit.

2. Left-click mapping.

3. Assign the layer to either a Direct mapping type with the lights you want to control; or a Feed mapping type that allows you to control the lights of more than one screen simultaneously, or control a subset of lights within a single DmxLights screen. In this example we have assigned a Direct mapping type to the mac2000 fixtures. You will then see a set of target points (small white crosshairs), one per fixture.
Target points used to change the orientation of moving head fixtures, created by mapping the Target 2 layer to the fixtures

**Please note:** to visualise the fixtures following the fixture targets, you must have the Aimable property set to Yes. For more information please see the sub-chapter *Creating a fixture.*
To animate the fixtures by editing multiple fixture targets please see the section Targeting/animating fixtures in the Target2 layer topic. To edit a single fixture target use the Target layer.
Creating DMX screens

Working with DmxScreens is the easiest way to get DMX output with the disguise software. They behave like regular screens, but each pixel's colour and/or brightness information is sent out directly via DMX. The disguise software has no understanding of what kind of fixture is accepting this DMX stream; it sees them only as pixels.

DmxScreens are suitable for LED fixtures that accept RGB or luminance only signals and have no extra control channels.

DmxScreens are not suitable for fixtures that have complex personalities including on/off commands, moving heads, CMY colour mixing lamps, etc. For these kinds of fixtures you should create a DmxLights, which allows more complex fixtures to be modeled. For more information please see the following sub-chapter DmxLights.

Creating a DmxScreen

The DmxScreen is added in the same way as any other screen type. Please see the sub-chapter Creating/removing screens for step-by-step instructions on how to add a screen to a stage, and select DmxScreen from the menu of different screen types.

DmxScreen properties

The DmxScreen properties are similar to the Screen properties, but with an extra tab called addressing which controls the way each pixels channel number is assigned. Therefore all of the DmxScreen properties not related to addressing can be read in the sub-chapter Editing screens.
DmxScreen has the same properties as a Screen but with an extra tab called Addressing (explained below)

**Channel**

The start channel number of the first pixel to be assigned. You can fill in numbers ranging from 1 to 512.
**Universe**

The start universe number of the first pixel to be assigned. Valid universe numbers start from 1.

**Status**

This is either **mute** (not sending DMX) or **active** (sending DMX). Because it may be dangerous to send DMX for an incorrectly addressed screen, the default status is **mute**. When the screen is addressed correctly, set this to **active**.

**Fixture type**

This specifies the fixture type. Options are **rgb** , **monochrome** (which computes a weighted sum from red, green and blue), **monochrome max** (which computes a single luminance from the maximum of red, green and blue) or **rgbw**.

**Neutral white**

This enables the user to color correct the pixels using a paletter or precise RGB value.

**Array**

Starting from the origin (see below), the assignment algorithm steps horizontally or vertically, increasing the channel number as it goes. This property controls whether the step is **horizontal** or **vertical**.

**Origin**

This controls the starting point for channel assignment. If set to **top left**, the top left pixel is the first to have its channel assigned; if set to **top right**, the top right pixel is the first to be assigned; and so on.
Universe step

For large screens, the total number of channels in the screen may exceed 512, the maximum number of channels in a universe. When this happens, the channel assignment algorithm will jump up a universe and resume assignment. If this universe step happens within a row or column, this may prove inconvenient for physical wiring as two cables will have to be run to the row or column. For this reason, disguise allows a row/column option which ensures that the universe number will not step halfway through a row or column.

Universe size

By default, a universe is assumed to be 512 pixels. If you want, you can set the size to something smaller. This allows a portion of the universe to be used for other fixtures or control channels, if required.

Always show channels

When set to yes, disguise superimposes the universe and channel number for each pixel over the pixel in the Stage Visualiser. Left-clicking a channel number allows you to edit it using the mouse wheel. Note that changing the colour, array, origin or universe step properties will reassign the channel numbers.

Please note: showing channels of very large arrays (eg. with 256x256 pixels) will kill the visualiser. For this reason, it helps to make sure you set the screen resolution correctly before you turn on show channels.
Creating DMX screens

Working with DmxScreens is the easiest way to get DMX output with the disguise software. They behave like regular screens, but each pixel's colour and/or brightness information is sent out directly via DMX. The disguise software has no understanding of what kind of fixture is accepting this DMX stream; it sees them only as pixels.

DmxScreens are suitable for LED fixtures that accept RGB or luminance only signals and have no extra control channels.

DmxScreens are not suitable for fixtures that have complex personalities including on/off commands, moving heads, CMY colour mixing lamps, etc. For these kinds of fixtures you should create a DmxLights, which allows more complex fixtures to be modeled. For more information please see the following sub-chapter DmxLights.

Creating a DmxScreen

The DmxScreen is added in the same way as any other screen type. Please see the sub-chapter Creating/removing screens for step-by-step instructions on how to add a screen to a stage, and select DmxScreen from the menu of different screen types.

DmxScreen properties

The DmxScreen properties are similar to the Screen properties, but with an extra tab called addressing which controls the way each pixels channel number is assigned. Therefore all of the DmxScreen properties not related to addressing can be read in the sub-chapter Editing screens.
DmxScreen has the same properties as a Screen but with an extra tab called Addressing (explained below)

**Channel**

The start channel number of the first pixel to be assigned. You can fill in numbers ranging from 1 to 512.

**Universe**

The start universe number of the first pixel to be assigned. Valid universe numbers start from 1.
Status
This is either mute (not sending DMX) or active (sending DMX). Because it may be dangerous to send DMX for an incorrectly addressed screen, the default status is mute. When the screen is addressed correctly, set this to active.

Fixture type
This specifies the fixture type. Options are rgb, monochrome (which computes a weighted sum from red, green and blue), monochrome max (which computes a single luminance from the maximum of red, green and blue) or rgbw.

Neutral white
This enables the user to color correct the pixels using a paletter or precise RGB value.

Array
Starting from the origin (see below), the assignment algorithm steps horizontally or vertically, increasing the channel number as it goes. This property controls whether the step is horizontal or vertical.

Origin
This controls the starting point for channel assignment. If set to top left, the top left pixel is the first to have its channel assigned; if set to top right, the top right pixel is the first to be assigned; and so on.

Universe step
For large screens, the total number of channels in the screen may exceed 512, the maximum number of channels in a universe. When this happens, the channel assignment algorithm will jump up a universe and resume assignment. If this universe step happens within a row or column, this may prove inconvenient for physical wiring as two cables will have to be run to the row or column. For this reason, disguise allows a row/column option which ensures that the universe number will not step halfway through a row or column.

Universe size
By default, a universe is assumed to be 512 pixels. If you want, you can set the size to something smaller. This allows a portion of the universe to be used for other fixtures or control channels, if
Always show channels

When set to **yes**, disguise superimposes the universe and channel number for each pixel over the pixel in the Stage Visualiser. Left-clicking a channel number allows you to edit it using the mouse wheel. Note that changing the colour, array, origin or universe step properties will reassign the channel numbers.

**Please note:** showing channels of very large arrays (eg. with 256x256 pixels) will kill the visualiser. For this reason, it helps to make sure you set the screen resolution correctly before you turn on **show channels**.
Editing local channels

If you want to edit DMX channels locally do the following:

1. Hold down Alt and left-click a channel.

2. If you want to change the universe, change the universe property.

3. Change the start channel in the channel property.

4. In the number of fixtures property choose number of fixtures, i.e. pixels you want to change the channel of.

5. Depending on the orientation of the UV map change the direction to right, left, up, down.

6. Click assign.

Interface showing how edit local DMX channels
DMX table screens

Overview

A DMX table screen is a DMX screen whose DMX patch is defined by a csv file. The behavior of the screen is much like the DMX screen except that the patch is automated by the csv file allowing for more fluid patch mechanisms for non-linear patching.

Tables should be placed in a newly created Table folder, inside the Objects folder of the project file.

CSV format

The first line is the four column headers (pos x, pos y, universe, channel) and is ignored. These properties are explained as follows.

**x position**

x coordinate of the pixel, zero-based, relative to the resolution of the mesh as defined in the disguise software. 0 is left.

**y position**

y coordinate of the pixel, zero-based, relative to the resolution of the mesh as defined in the disguise software. 0 is top.

**universe**

DMX universe number offset, 1-based, relative to the start universe of the screen as defined in the disguise software.
channel
DMX start channel number offset of the pixel, 1-based, relative to the start address of the screen as defined in the disguise software.

Workflow

1. Create your DMX patch using a CSV or spreadsheet editor.

2. Create your DMX screen in the disguise software as normal remembering to set the resolution correctly.

3. Use the Table property of the screen to point to the CSV file that defines the channel assignments.

Example

Creating a table in excel
1. Define the column order in Excel.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th>universe</th>
<th>channel</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>1</td>
<td>4</td>
</tr>
<tr>
<td>1</td>
<td>3</td>
<td>1</td>
<td>7</td>
</tr>
<tr>
<td>1</td>
<td>4</td>
<td>1</td>
<td>10</td>
</tr>
<tr>
<td>1</td>
<td>5</td>
<td>1</td>
<td>13</td>
</tr>
<tr>
<td>1</td>
<td>6</td>
<td>1</td>
<td>16</td>
</tr>
<tr>
<td>1</td>
<td>7</td>
<td>1</td>
<td>19</td>
</tr>
<tr>
<td>1</td>
<td>8</td>
<td>1</td>
<td>22</td>
</tr>
<tr>
<td>1</td>
<td>9</td>
<td>1</td>
<td>25</td>
</tr>
<tr>
<td>1</td>
<td>10</td>
<td>1</td>
<td>28</td>
</tr>
<tr>
<td>1</td>
<td>11</td>
<td>1</td>
<td>31</td>
</tr>
<tr>
<td>1</td>
<td>12</td>
<td>1</td>
<td>34</td>
</tr>
<tr>
<td>1</td>
<td>13</td>
<td>1</td>
<td>37</td>
</tr>
</tbody>
</table>

The X is the x axis in the UV Space and Y is the y axis in the UV Space when looking at UVs of a 3d Model.

2. So if the resolution of the Screen is 10x10 for example then the co-ordinates of the UV space would look like the image below.
3. If the Fixture is a 3 channel fixture, then the address would be as follows compared to the table screen.
4. Now export the x, y, universe, channel information as a .csv file.

**Excel table creation tip**

The following method expects the LEDs to be addressed column by column, left-to-right. If you require a different addressing order, the x and y equations can be swapped.

**x column**

Use the equation `=TRUNC((ROW()-1)/y)`, repeat this down the sheet to x*y number of rows, this will make each number from 0 to x repeat y-number of times.
**y column**

Use the equation \(=\text{TRUNC}(\text{ROW()-1})-(y*\text{TRUNC}((\text{ROW()-1})/y))\), repeat this down the sheet to \(x*y\) number of rows. This will make the numbers from 0 to \(y\) repeat with each number from 0 to \(x\), covering every permutation of these two ranges.

**Universe column**

Find the maximum number of LEDs you can fit in one universe (\(n\)). For RGB or any other three-channel LED's this number is 170. Use the equation \(=\text{TRUNC}(((\text{ROW()}+\text{(n-1)})/n)\), repeat for \(x*y\) rows.

**Channel column**

Put a 1 in the first row, then \(=\text{MOD}(D1+i, 510)\) (where \(i\) is the number of channels required by a single LED) in the next row down, copy down letting the D co-ordinate increment. 510 is obtained by adding 1 to the highest addressable number (509 in the case of RGB LED's), repeat for \(x*y\) rows.

Once this is complete, export as csv, open in a text editor, create a new line at the top of the list and type "x,y,universe,channel". Save the csv and use it in the disguise software.

**Placing the CSV file**

1. Copy the exported CSV file from excel.
2. Create a new folder inside the objects folder of your project file, name this folder **Table**.
3. Paste the CSV file into this newly created table folder.

Creating the DMXtablescreen

1. Open the stage editor by right clicking Stage from the dashboard or by right clicking the floor in the visualiser.

2. Expand the DMXscreens tab.
3. Click the + icon to open the DMXscreens manager.

4. Define a name for your screen and left click OK.

5. Position your screen in the stage as normal.

6. Define the **Table property** as the CSV file you placed in the Table folder earlier.
Projector Simulation overview

In the disguise software, content is created and as a texture applied to the projection surface - the car in the example here. Content can be made without any knowledge of the actual number of projectors that will be used - whether there is one projector or twenty, the content will stay the same.

In the disguise software, you begin by creating Projector objects and add them to the scene; you set up their position, rotation and lens qualities to precisely match those of the real projectors. The software then renders the projection surface (and its content) from the point of view of each virtual projector, and outputs the resulting image to the real projector. If the projection surface and virtual projectors precisely match their counterparts in the real world, the result is a perfect image.

The disguise software used to visualise, simulate and calculate how physical projectors should be positioned for the **Bentley Continental GT UK Tour**

If the physical projector has to move a couple of meters to the side, there is no need to re-render any content, just move the corresponding virtual projector to match. It will then render a new image from the
new point of view. Similarly, if new projectors are added, there’s no need to re-render content; just add a
new projector to the scene and set it up.

There are two ways of matching a virtual projector to its physical counterpart: Manual calibration, or
the calibration process referred to as QuickCal. For situations where your 3D model does not precisely
match the real projection surface, disguise provides a selection of warping tools.

Outputs that the three virtual projectors from the above image see in the Output Feeds level

Projector Simulation as a decision-making tool

Projector simulation does not only have to be used for output purposes. During the early production
stages, Projector Simulation can be used to explore projector positions, aiming points, appropriate
lens sizes etc; it is a tool for specifying the correct hardware for a show.

3D models preparation

As always when working in the disguise software, you need to model an exact virtual replica of the real-
world screens. If only standard rectangular screens are being used, these can be created directly in the
disguise software. However, for more complex three-dimensional objects, it is required that the screen is
built and exported from a 3D software such as 3ds Max, Maya, Cinema4D or similar. It is crucial to not
only model and export the 3D models, but to also properly texture-map (UV-map) them and to carefully
optimize the polygon count of the model.

The more accurate the 3D model the better. The level of accuracy usually depends on the distance
between the audience and the object. For building projections, it is recommended to laser scan the
building. A laser scan can generate a highly accurate 3D mesh (+-5mm accuracy) that you can import
into a 3D software application from which the optimization, UV-mapping and content template creation
process can take place. The mesh is then exported as a texture mapped .obj file into disguise.

For smaller objects, where the audience stands much closer to the object and the pixel density is higher
(for example for car projections), it is even more important that the 3D mesh is as accurate as possible.
However, as this usually is not the case, disguise has a number of 2D-based warping tools allowing you
to compensate for the inaccuracies between the virtual and the real-world model. Please see the sub-
chapter Warping outputs for more information.
Content creation

Based on 3D models, animators can create content either in 2D or in 3D, or both. For simple setups, a 2D content template generated from the UV-coordinates of the 3D model can be used as a background image in for example After Effects.

Video files are rendered and imported. In the software, the content is then applied back onto the same 3D models, allowing the content to be pre-visualised in 3D and in real-time, from any point of view. As the output to the physical projectors is handled by the virtual projectors, the animators only need to think about how to create content that looks good on the screen in the disguise software.

How do I learn about preparing a projection-based 3D model for disguise?

The 3D modelling and UV-mapping processes are currently not covered in detail in this User Guide. To learn more about these processes, register for the Pre-Production Workflows course offered online in our free e-learning system at https://training.disguise.one.
Projector Simulation overview

In the disguise software, content is created and as a texture applied to the projection surface - the car in the example here. Content can be made without any knowledge of the actual number of projectors that will be used - whether there is one projector or twenty, the content will stay the same.

In the disguise software, you begin by creating Projector objects and add them to the scene; you set up their position, rotation and lens qualities to precisely match those of the real projectors. The software then renders the projection surface (and its content) from the point of view of each virtual projector, and outputs the resulting image to the real projector. If the projection surface and virtual projectors precisely match their counterparts in the real world, the result is a perfect image.

The disguise software used to visualise, simulate and calculate how physical projectors should be positioned for the Bentley Continental GT UK Tour

If the physical projector has to move a couple of meters to the side, there is no need to re-render any content, just move the corresponding virtual projector to match. It will then render a new image from the
new point of view. Similarly, if new projectors are added, there’s no need to re-render content; just add a new projector to the scene and set it up.

There are two ways of matching a virtual projector to its physical counterpart: Manual calibration, or the calibration process referred to as QuickCal. For situations where your 3D model does not precisely match the real projection surface, disguise provides a selection of warping tools.

Outputs that the three virtual projectors from the above image see in the Output Feeds level

Projector Simulation as a decision-making tool

Projector simulation does not only have to be used for output purposes. During the early production stages, Projector Simulation can be used to explore projector positions, aiming points, appropriate lens sizes etc; it is a tool for specifying the correct hardware for a show.

3D models preparation

As always when working in the disguise software, you need to model an exact virtual replica of the real-world screens. If only standard rectangular screens are being used, these can be created directly in the disguise software. However, for more complex three-dimensional objects, it is required that the screen is built and exported from a 3D software such as 3ds Max, Maya, Cinema4D or similar. It is crucial to not only model and export the 3D models, but to also properly texture-map (UV-map) them and to carefully optimize the polygon count of the model.

The more accurate the 3D model the better. The level of accuracy usually depends on the distance between the audience and the object. For building projections, it is recommended to laser scan the building. A laser scan can generate a highly accurate 3D mesh (+/-5mm accuracy) that you can import into a 3D software application from which the optimization, UV-mapping and content template creation process can take place. The mesh is then exported as a texture mapped .obj file into disguise.

For smaller objects, where the audience stands much closer to the object and the pixel density is higher (for example for car projections), it is even more important that the 3D mesh is as accurate as possible. However, as this usually is not the case, disguise has a number of 2D-based warping tools allowing you to compensate for the inaccuracies between the virtual and the real-world model. Please see the sub-chapter Warping outputs for more information.
Content creation

Based on 3D models, animators can create content either in 2D or in 3D, or both. For simple setups, a 2D content template generated from the UV-coordinates of the 3D model can be used as a background image in for example After Effects.

Video files are rendered and imported. In the software, the content is then applied back onto the same 3D models, allowing the content to be pre-visualised in 3D and in real-time, from any point of view. As the output to the physical projectors is handled by the virtual projectors, the animators only need to think about how to create content that looks good on the screen in the disguise software.

How do I learn about preparing a projection-based 3D model for disguise?

The 3D modelling and UV-mapping processes are currently not covered in detail in this User Guide. To learn more about these processes, register for the Pre-Production Workflows course offered online in our free e-learning system at https://training.disguise.one.
Creating and removing projectors

Projectors in the disguise software are currently categorised as a type of screen, as they share so many of the same properties - they have a position, a rotation, a resolution, and output content via the feed system.

Creating a projector for a stage

Please see the sub-chapter Creating screens for step-by-step instructions on how to add a screen to the disguise Stage.

Select Projector from the menu of screen types.

The below image shows an example of two projectors added to the d3 Stage.

Stage Visualiser being used to visualise two projectors and a sculpture in a section of the d3 studio

Removing a projector from a stage

Open the Stage Editor by right clicking Stage on the d3 dashboard
Or alternatively
- Right click the floor
- Left click and drag the Projector into the - icon
Importing projectors

From r16, the disguise software started supporting import of projector information from external CSV files. Currently, both Mapping Matter and SimProj are supported.

Overview

The feature is intended to take a CSV file with a .txt file extension that contains the projector data. Currently, Mapping Matter and SimProj are known to work well. The CSV can be generated automatically from their programs, or manually, as long as the same format is used.

An example of CSV format without a projector UUID.

Projector_Name,Projector_Qte(Stack),Projector_Native-Rez-X,Projector_Native-Rez-Y,Projector_Lumens(lux),Projector_Brightness(%),Projector_Total_Lumens(lux),Projector_Trow-Ratio,Lens_Shift-H(%),Lens_Shift-V(%),Lens_X,Lens_Y,Lens_Z,Pitch(deg),Yaw(deg),Roll(deg),Target_X,Target_Y,Target_Z,Target_Distance,Target_Width,Target_Height,Target_Illuminance,Target_DPI,Unit_Dim,Unit_Illuminance

An example of CSV format with projector UUID.

Projector_Name,Projector_Qte(Stack),Projector_Native-Rez-X,Projector_Native-Rez-Y,Projector_Lumens(lux),Projector_Brightness(%),Projector_Total_Lumens(lux),Projector_Trow-Ratio,Lens_Shift-H(%),Lens_Shift-V(%),Lens_X,Lens_Y,Lens_Z,Pitch(deg),Yaw(deg),Roll(deg),Target_X,Target_Y,Target_Z,Target_Distance,Target_Width,Target_Height,Target_Illuminance,Target_DPI,Unit_Dim,Unit_Illuminance,Projector_UUID

Workflow

The intended workflow for this feature is as follows.

- User creates their projector configuration in Mapping Matter or SimProj.
- Export projector configuration to a correctly formatted CSV file.
- Create a table subfolder in the objects folder of your disguise project and put the CSV file there.
- Use the Import Projector functionality in the disguise software.
Example

- Navigate to the Objects folder of your project.
- Create a Table sub folder if one does not already exist.
- Put the exported CSV file in the table folder.
- In the disguise software open the stage editor and navigate to actions.
- Left click **Import projectors from table**.
- Select the CSV file from the list of table files.

If an unsupported file is selected (e.g. containing something else than projectors), the import will fail, and an error message will inform the user.

If it succeeds, all projectors in the CSV file will be imported and added to the Stage.

Existing projectors with the same name will be modified. If they are not in the Stage at the moment, they will be added.

If a projector with a certain name does not yet exist, it will be created and added to the Stage.

If the file contains multiple projector entries with the same name, disguise will create duplicates with different names.

the disguise software will notify the user with a popup message about how many projectors were added, modified, or failed.

**Please note:** The import of projectors is a one-off process, and needs to be repeated manually every time the file changes. The table filename is not saved and tracked in the project, like it is with other Table workflows in the disguise software.
Editing projectors

This sub-chapter explains the definition of the projector properties. These properties can be edited using the projector editor. To understand how to use the properties please see the sub-chapters Manual Calibration and QuickCal.

Editing a projector

To edit a projector, either right-click directly on a projector in the Stage or right-click the projector from the screens list in the Stage editor.
General projector properties

**Resolution**

The x and y resolution of the projector, set in pixels. This should ideally be the native resolution of the physical projector model you’re using.

**Currently outputting**

Type of output mode that currently is active.

- **Content**: Outputs the content played on the timeline.
- **Wireframe**: Outputs the wireframe of the model.
- **Identify**: Full colour image with the projectors name on it.
- Grid: Outputs a coloured grid applied to the projection object
- None: Outputs nothing (except reference points during QuickCal lineup)
Offset

The x, y and z position of the projector based from the origin point (0,0,0), set in meters.

Rotation

The rotation of the projector around the x, y and z axes respectively, in degrees.

Look At

The x, y and z position (in meters) of the Look At point. The projector always rotates to point at the Look At point.

Look Distance

The absolute distance between the projector position and its Look At position, in meters.

Throw ratio

The throw ratio is defined as the throw distance "D" divided by the width of the projected image, "W". Since D and W are both in meters, the throw ratio quantity itself has no unit. Simply, the larger the throw ratio, the narrower the focus. Projector lenses are usually rated at particular throw ratios; you would simply type in the lens throw ratio here.

Field of View

Field of view in degrees for the specified throw ratio
Lens shift

Represents horizontal and vertical lens shift. When horizontal shift is set to 0, there is no shift; when it's set to 1, the left edge of the projected image lines up with the central projection axis. The same applies to negative numbers (-1 lines up the right edge) and to the vertical shift (bottom and top edges respectively).

Radial distortion

Radial lens distortion coefficients k1, k2, k3

Tangential distortion

Tangential lens distortion coefficients p1, p2
Surfaces

The list of screen surfaces the projector will project onto. The projectors view of the virtual stage will only contain these surfaces.

To add a projection surface to the list:

- Left-click the + button to open the Surfaces manager.
- Select the surfaces you wish the projector to see.
Mask objects

The list of objects this projector should render as black.

To add a mask object to the list:
- Left-click the + button to open the Objects manager.
- Select the objects you wish the projector to mask out (render black)

**Mask scale**

Mask scale allows you to scale the texture that is generated as the mask.

**Blur shadow**

Blur shadow allows you to add a blur effect to the edges of the generated mask.

**Mask resolution**

Mask resolution, set to Full by default, allows you to only generate the mask at half and quarter resolution for times when full resolution is not needed, or to save performance.
Appearance

The colour of the projector beam in the visualiser.

Colour

The colour of the projector beam in the visualiser.
Render Mode

Switches the projector's output in the virtual stage for easier identification of testing.

Types of render mode:

- Hidden
- Beam
- Wire
- Beam & Wire

QuickCal reticle

Create and chose a custom bitmap as the QuickCal lineup cursor.

Brightness (Lumens)

Projector brightness, in ANSI Lumens. This setting affects the visualiser only.
Calibration

Configuration

The configuration of a projector stores its **calibration** - a precise measurement of its actual position and lens settings. Calibrated quantities are not necessarily the same as the properties you enter in the preceding boxes, but will usually match fairly closely.
Left-click on the configuration property to open the library of settings for all projectors. You can also create multiple configuration files for the same projector.

Right-click configuration to start a QuickCal lineup.
Output
**Master Fade**

Controls the output brightness.

*Please note:* the Master fade only controls the brightness of the output feed. The brightness in the stage visualiser is not affected.

**Feather**

Shrinks or grows screens in the projector's view, to fix over-projection.

**Final Adjust**

Adjust the offset added to a projector's position, in meters

**Final Adjust Rotation**

Offset added to a projector's rotation, in degrees

**Anti-Aliasing**

The anti-aliasing options for the projector.

*Warning:* enabling anti-aliasing on projectors can cause performance degradation.

- None
- 2X FXAA
2X MSAA

4X MSAA

8X MSAA

**Face Cull Mode**

Use this to skip rendering faces with normals pointing toward (Front) or away (Back) from the projector.

**Colour shift**

The *Colour shift* property allows you to perform a number of colour-correction operations on the output of a projector.

To apply a *Colour shift* to a projector:

- Left-click *Colour shift* to open the Colour shift manager
- Either create a new Colour shift by typing a name into the new *colour shift* text box and hit **Enter**, or select a colour shift configuration from the list.

To edit a *Colour shift*:

- Right-click a *Colour shift* in the Colour shift manager to bring up the Colour shift editor.

**Colour Profile**

Colour/gamma space of the output display

**Colour LUT**

Cube LUT applied to correct display device output; does not appear in visualizer.
Hold Output

Stops the output of this display until unchecked

Change Output Mode

Left-click to change the type of output mode that currently is active.

- **Content**: Outputs the content played on the timeline.
- **Wireframe**: Outputs the wireframe of the model.
- **Identify**: Full colour image with the projectors name on it.
- **Grid**: Outputs a coloured grid applied to the projection object
- **None**: Outputs nothing (except reference points during QuickCal lineup)
- **BlackTrax**: For Blacktrax line up
Dynamic Blend
**Dynamic Blend**

Allows you to enable/disable Dynamic blend on individual projectors.

**Membership function**

Dynamic Blend Membership Algorithm;
Options are Linear, Radial, and Linear to Radial

**Gamma bias**

Adjusts the estimate of this projector's gamma relative to the Stage's global blend gamma.

**Barn door (top)**

Top edge of the Dynamic Blend

**Barn door (left)**

Left edge of the Dynamic Blend

**Barn door (bottom)**

Bottom edge of the Dynamic Blend

**Barn door (right)**

Right edge of the Dynamic Blend
Exclude Projectors

Projectors excluded from the Dynamic Blend

Use snapshot

Lock the Dynamic Blend in a snapshot

For more information on using Dynamic Blend, please visit this link.
Hierarchy

For more information on Hierarchy, visit this link
Outputting to projectors

The disguise software uses Feed scenes to output projector outputs. The output displays contained in these Feed scenes can be configured to a specific resolution and frame rate. For more information please see the sub-chapter Configuring Feed scenes.

Creating an output from a projector

Go to the Feed level by left-clicking Feed from the dashboard. The top half of the screen displays the video screens and the outputs from the projectors currently active stage in the Stage level.

Right-click the first projector output, then left-click add feed rectangle. Please make sure that you clicked a projector output and not the video screen that the projector is sampling from. This will generate an output feed from the projector and place it on head 2 (by default, output head 1 is represented by the GUI). This should now send a feed to the first physical projector. Repeat this step for each projector that needs to be outputted. For a show that requires more than 3 projectors (1 GUI + 3 outputs) you will have to create a Director/Actor network explained in the Director/Actor setup using d3Net.
The Add Feed Rectangle option is used to add a Feed rectangle and generate an output feed from a projector.
Projector List Overview

The Projector List is a faster way of managing calibration when you have a large number of projectors to work through.

To open the Projector List, hit **Ctrl + P**

The Projector List. Open the Projector List by hitting Ctrl-P or via the Stage Editor.

Features of the Projector List

**Select**

- Left-click to select which projectors are to be controlled together.

**Projector**

- Left-click on a projector name to open its Projector Editor

**Status**

- Left-click to turn projectors on and off (‘mute’ means ‘faded to black’)

---

149
Lock
- Left-click to lock the projector settings

Output Mode
- Left-click to switch the output mode of the projectors

Calibration
- Left-click on a projector config to open the projection calibration editor (QuickCal)

Source Rectangle
- Left-click to jump to the feed source rectangle for that projector in the feed view

Destination Rectangle
- Left-click to jump to the destination (output) rectangle in the feed view

Final Adjust
- Fine-adjust selected projector’s positions in x, y & x axes for last minute adjustments (hold down the Ctrl key for fine control or the shift key for fast control)

Multiple Projector Lists
Everytime you hit Ctrl + P another instance of the projection list is opened. Using the select buttons you can filter them differently to aid workflow when using large numbers of projectors.
Manual Calibration - Projectors

Warning: reset all digital and optical warps inside the projector, including lens-shift, otherwise the lineup features will be interrupted.

- Open the projector editor by either right-clicking the projector directly in the Stage, or by right-clicking the projector from the screens list in the Stage editor.

- Familiarize yourself with the properties of a projector. After this follow the instructions explained below.

Set the correct resolution

Set the resolution of the projector to match the corresponding output head’s resolution.
**Warning:** If the resolution of the projector doesn’t match the real world output, you will end up with an incorrect calibration.

**Add surfaces**

Under the **Surfaces** tab add the projection screens that the particular projector is covering.
Warning: If you don't add a screen to the projector the output will stay black.

Add the projector outputs to the Output Feeds

Ensure all of the Feed rectangles from the projector outputs have been added to the output heads.

Please see adding feed rectangles for information how to do this.
Place the projector correctly

Change the pos (projector position) and the throw ratio (lens value) properties so that the projector covers the required part of the screen surface. A laser measure may be required to calculate the correct position by measuring the distance from the physical projector to the video surface.

Adjust the Look at position

The Look at position of a projector defines the centre point of its corresponding output feed. When lining up a virtual projector to the physical projector it is therefore crucial to match the Look at position to its corresponding point in the real world.

Disguise has a built-in wireframe feature allowing the system to generate a line-drawing based on the 3D mesh of the projection surface. In the same output frame disguise will also output a red cross positioned in the centre of the output feed. When the wireframe test pattern is applied, disguise will output this red cross regardless of the orientation of the projector.

Consequently, if the look at position of the projector is aligned to the red cross on the physical projection surface, the virtual projector and the physical projector are orientated around the same point which is a great starting point for an accurate manual lineup.

Align the look at position of the projector

- Change the output mode to wireframe by clicking the output tab at the bottom of the projector editor.

- Return to the Stage level. Set the step value of the look at position to 0.01 to enable smoother scrolling of the look at values. To change the step value, right-click the property name and change the value in the Step value property.

- Begin aligning the look at position of the projector to the red cross by comparing the look at positions crosshair in the Stage level and the red cross being outputted from the physical projector.
Look at position of the virtual projector now matches its corresponding point on the physical video screen, in this example a wooden sculpture.

Adjust the throw Ratio

- To zoom in on the physical video screens content, change the throwRatio value. This value corresponds to the lens size of the physical projector.
- If the exact lens size is known (for example a fixed lens size is being used), type it in.
- If a zoom lens is being used, set the start value to the lowest (or highest) value in the zoom range. Slowly change the value by scrolling the mouse wheel in the property field.

Fine tune property values

- After setting the initial lens value, try not to edit the position properties. Instead, start adjusting the rotation parameters if needed, in particular the x and y rotations. Aim to establish parallel lines mapping onto the video screen globally rather than focusing only on one part of the screen. Adjust the step value if needed.
- Go back and fine tune the values of the lookAt position to center the output to the physical video screen. Remember to establish parallel lines.
- Adjust the throw ratio to zoom in/out of the content on the video screen. Assuming that the 3D mesh is accurate to the physical video screen, the mapping should gradually fall into place.
Projector has been manually calibrated; the 3D mesh test pattern now lines up with the wooden sculpture.

If the projection surface does not match the 3D model after carrying out manual calibration the lineup may need to be fine-tuned. Please see the sub-chapter Warping outputs for more information.
Warping outputs

In cases where the projection surface does not match the virtual 3D model (i.e. where the lineup from the manual or automatic calibration needs a slight fine-tune), the disguise software provides a number of powerful warping layers that can be applied directly to the output. By using these tools there is no requirement for projector keystoning or any other warping inside the physical projector.

⚠️ Warning: reset all digital and optical warps inside the projector, including lens-shift, otherwise the lineup features within the disguise software will be interrupted.

Adding/using warping layers

To add a warping layer to an output left-click **feed** in the dashboard (bar at the top of the screen), right-click the appropriate output and select **Edit Warp** from the options list.
In the editing window, left-click the + button under the **Deformation layers** title. This will display a list of deformation layers.
Deformation layers for warping an output feed, opened by left-clicking + under the Deformation Layers title

**Please note:** Warp points can be added to the output so you can see exactly which part of the output is being manipulated
Warp points being added to an output
Drag any of the 4 points inwards to globally adjust the warping of the output. Adjust the horizontal and vertical bias to shift-scale the content.
9point warp allows you to drag 9 points inwards and adjust the pinch and straightness between them. This feed warp is useful when projecting onto a cylindrical screen when not using Projector Simulation. It easily conforms a rectangular output to fit to a cylindrical screen.
Custom

Creates an arbitrary number of grid points on the output which can be group-selected. You can use this when local warping on an object is required.
Grid
Reference other projector
Circular
Lens deform

Warp points can be added to the output so you can see exactly which part of the output is being manipulated.
Blending and masking outputs

The disguise software allows the user to generate custom softedge textures and masks inside it. You can then apply the softedge texture directly on the output and update the shape, gradient and gamma settings in real-time.

Create a softedge mask on the output

Select Mask option is used to create a custom SoftEdgeTexture mask

<table>
<thead>
<tr>
<th>untitled FeedRect</th>
</tr>
</thead>
<tbody>
<tr>
<td>Display</td>
</tr>
<tr>
<td>Source position</td>
</tr>
<tr>
<td>Source size</td>
</tr>
<tr>
<td>Lock sizes</td>
</tr>
<tr>
<td>Output position</td>
</tr>
<tr>
<td>Output size</td>
</tr>
<tr>
<td>Output rotation</td>
</tr>
<tr>
<td>Output render mode</td>
</tr>
<tr>
<td>Mask</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>Export As Table:</td>
</tr>
<tr>
<td>Toggle selection</td>
</tr>
</tbody>
</table>
This will open the Texture object library which shows all of the still image files saved on your local hard-drive in the *DxTexture* folder. For more information on the *DxTexture* folder and other folders please see the sub-chapter Placing media files for a project.

In the top-left corner of the library, where it says *new texture*, type in the name of the softedge mask, for example *Output1_softedge*.

*Hit Enter* and select the texture type *SoftEdgeTexture*.

Open the Feed level by left-clicking *Feed* from the dashboard (bar at the top of the screen), right-click the appropriate output and select *select mask* from the options list.

Texture object library being used to create a SoftEdgeTexture mask.

Create a new Texture, and choose SoftEdge Texture when prompted.
Softedge editor

Resolution

The resolution values are shown in the top right corner of the softedge editor. The resolution is automatically set to the resolution of the output head.

Adjust global gradients through gamma

The **gamma** value adjusts the global gradient between the inner and outer points in the softedge mask.
Select and move points

- Select individual points, pairs of inner and outer points, or drag-select multiple points to adjust the blend level.

Add and remove points

- Add more points by holding down Ctrl and left-clicking between existing points.
- Remove points by right-clicking a point.

Adjust local gradients

- Adjust the gradient between an inner and outer point by dragging them further apart, or closer together.

Please note: press Esc to exit texture editing. Right-click the output feed and select ‘Edit mask (Name)’ to edit the texture again.

Create a mask rather than a softedge

- To create a mask rather than a softedge, set either gamma to 0 (this will set a sharp edge between the inner and outer points regardless of spacing between them) or move the inner and outer points closer together.

Composite multiple textures

Disguise allows the user to generate two different textures and composite them into one single texture. To do this:
Create two different textures (they do not have to be of the type \texttt{SoftEdgeTexture} but can also be of the types \texttt{GradientTexture} and \texttt{SolidTexture}.

Repeat the same process as above when creating a texture but this time choose the texture type \texttt{CompositeTexture} and name it, for example, \texttt{SoftEdge\_output1} and \texttt{Gradient\_output1}.

After creating the the CompositeTexture textures, their corresponding outputs will turn grey. This is because the CompositeTexture textures need to be told what textures they contain.

Right-click the output head and select edit mask (\texttt{SoftEdge\_output1}) and edit mask (\texttt{Gradient\_output1}) in the options window.

Set the resolution of the CompositeTexture texture.

Add the first texture by left-clicking \texttt{add new layer}, and then selecting the image from the Texture library.

Click \texttt{add new layer} again and select the second image.

To multiply the two images together, set the blend mode of image two to \texttt{multiply}.

Press \texttt{Esc} to exit texture editing.
Dynamic Blend

Dynamic Blend is a feature of the disguise software that enables automatic calculation of soft edge blends across projectors automatically. The disguise software can do this as long as it has accurate information of where the projector is in 3D space.

Enabling Dynamic Blend

- Right click the stage editor from the dashboard or alternatively
  - Right click the stage floor
- Expand the Projectors tab
Left click the Dynamic Blend property and set it to **On**

**Global Dynamic Blend Settings**

**Blend Gamma**

Blend Gamma controls the gamma curve and should be matched to the same as the physical projectors.
Blend Sharpness

Controls the size of the blend zones created; the higher the number, the smaller the blend zones.

Projector Level Dynamic Blend Settings

There are further controls for Dynamic Blend within the individual projector settings.
**Dynamic Blend**

Enabled/Disabled. This allows you to enable or disable dynamic blend on individual projectors.

**Membership Function**

Linear/Radial/Linear to Radial. This controls how the blend is calculated for the individual projector.

**Gamma bias**

This adjusts the gamma bias of the individual projector relative to the gamma set in the stage editor.

Barn Door top/left/bottom/right

**Include/Exclude list**

See the Include/Exclude topic

**Use Snapshot**

Yes/No. This allows you to snapshot the soft edge mask generated by dynamic blend, edit it in external program, then have it re-applied to the projector. The texture files will appear in the DxTexture folder.
Include Exclude list

The include & exclude functionality found under the dynamic blend tab of a projectors properties, is specifically for use when dynamic blending large numbers of projectors. Dynamic blend has a limit of 12 projectors, so when going beyond 12 the include/exclude functionality allows you to specifically defined which projectors blend with which other projectors.

When using this functionality, you should stick to using either include or exclude per group of projectors.

Using the Include & Exclude list

- Right click on a projector to edit its properties
- Expand the Dynamic Blend tab
- Left click Exclude to choose between either Exclude or Include
- Select the other projectors that this projector should include or exclude blending with
Projector studies

Projector studies allows you create technical documents from the disguise software that contains photometric heatmaps of the scene and projector properties.

The photometric quantities can be specified as either the luminance of the scene or the illuminance.

The PDF document contains the heatmap from the visualiser cameras view and each projector. Each page of a projector study will show the following:

- Software version
- Project name
- The photometric quantity used.
- The user specified revision and comment.
- The time stamp the study was created.

In addition, each page that contains a projector view will show the following properties:

- Name
- Resolution
- Brightness (lumens)
- Throw Ratio
- Lens Shift Horizontal
- Lens Shift Vertical
- Position

The final page will show a table of all projector properties.
Projector Properties Table

This table is a .csv file that contains the properties of the projectors in the stage.

The properties are as follows:

- Name
- Resolution X
- Resolution Y
- Brightness (lumens)
- Throw Ratio
- Lens Shift Horizontal
- Lens Shift Vertical
- Position X - Meters
- Position Y - Meters
- Position Z - Meters

Considerations

The photometric analysis does not take into account the following

- Atmospheric absorption of light as it passes through space.
- Brightness decay due to projector lamp life.
- Effects of lens focus and distortion.
- Projector brightness uniformity.
- Physical properties of the surface material.
It assumes a lamberian reflectance of all surfaces, that they are perfectly diffuse and reflect light equally in all directions so that the brightness is the same no matter the viewing angle.

1. Set your project up as normal, placing projectors, screens and other elements.
2. View the scene lighting in the heatmap rendering mode to see how close the setup matches expectations.
3. Open the Stage editor.
4. Define the desired settings in the projector studies editor.
5. Export the projector study.

Enabling heatmap

1. Open the stage editor by right clicking Stage from the dashboard or by right clicking the floor in the visualiser.
2. Expand the Cameras tab, and right click Visualiser camera to edit the visualiser camera settings. Alternatively, you can edit another camera and output that to the feeds.
3. In the camera editor, expand the Renderer tab and set the render mode to Heatmap.
You are now viewing the heatmap visualisation of the photometric quantity specified by the Light field under the render tab within the camera editor.

The key shown in the heatmap editor shows the colour scheme and scale of the heatmap.
Editing heatmap properties

1. Open the heatmap editor.

2. Set the target value to be the preferred surface illuminance or luminance that you want to achieve.

Creating a new projector study

1. Open the stage editor by right clicking Stage from the dashboard or by right clicking the floor in the visualiser.

2. Expand the Projectors tab of the stage editor.

3. Left click Projector studies... to open the Projector studies editor.

4. Left click Export projector study and the projector study will be written to the projectorstudies folder within your project folder.

![Projector Study](image)

*The projector study editor*

Light
**Luminance** - Luminance - The intensity of light emitting from an object or surface per meter in a given direction, measured in nits (cd/m²).

**Illuminance** - The density of incident light in lumens hitting a surface per meter, measured in lux (lm/m²).

**Project**

The name of the project. This can be changed without affecting the main project name.

**Revision**

User definable revision field.

**Comment**

User definable comment field.

**Export Projector Study**

Clicking Export Projector Study will write the projector study to the projectorstudies folder in the disguise software project folder.
Cameras

Cameras are another display type in the disguise software. Similarly to projectors they render the scene from a specific point of view and allow you to output that view using feed rectangles in the Feed editor.

Overview

The Visualiser’s view of the stage is also a Camera called the Visualiser Camera. This can be accessed by clicking the Visualiser Camera button below the Cameras list or right clicking on the background.

⚠️ **Warning:** Be aware that having multiple cameras requires part of the scene to be rendered multiple times so can incur a major performance cost.

Workflow

Creating a camera

1. Open the stage editor by right clicking **stage** from the dashboard or by right clicking the floor in the visualiser.

2. Expand the **Cameras** tab.
3. Click + to add a camera.

4. Left click the **new camera** field in the Camera manager.

5. Enter a name for the camera.

6. Left click **OK**.
7. Choose either camera or sphericalCamera

8. You will now see a camera object in the visualiser, at the origin point (0,0,0).
Altering camera properties

1. Open the stage editor by right clicking **stage** from the dashboard or by right clicking the floor in the visualiser.

2. Expand the **Cameras** tab.

3. Right click a camera from the list.

4. Adjust the desired properties of the camera using the editor.
Assigning the camera to a feed

1. Position the camera in the visualiser.

2. Left click **Feed** from the dashboard.

3. Right click on the camera feed at the top of the feed scene and left click **Add feed rectangle** or alternatively hold **ALT**, left click and drag an arrow from the camera feed to an available output head.
Camera properties
The camera properties include the following categories:

- **Settings**
- **Preview**
- **Output**
- **Hierarchy**
- **Stage Render**
- **Physical**
Settings

Offset

The offset in meters of the camera’s position from its parent. When parented to the stage which is the default this is the camera’s world position.

Rotation

The rotation transformation in degrees from its parent. When parented to the stage, which is the default, this is the camera’s world rotation.

Tracking source

See Object tracking source for more information.

Resolution Source (visualiser camera only)

Take from GUI - Default, Use the resolution of the GUI display.

Fixed - You can specify a custom resolution. Will not change the resolution of the display, only the camera’s view.

Resolution

The resolution that the camera will use to render its view.

Field of view (vertical)

The vertical field of view in degrees of the camera.
Field of view (horizontal)

The horizontal field of view in degrees of the camera.

Preview

Camera plate

Sets the camera to Transmission, Backplate or Frontplate.

Show Alpha

Toggles whether the camera will show Alpha or not.

Output

Set extension fade

Opacity of the set extension.

Set extension feather

Feather of the set extension. This is defined as the width of the feather.
Video input fade
Defines whether you have video input on or off. Helpful if you want to see how well the set extension is lined up.

AR fade
AR content on or off. Frontplate content only. Helpful to see how well the AR content is lined up.

Radial mask scale
Applies a radial mask to the film content.

Frontplate reprojection

Backplate aligned
Front plate is delayed and reprojected with the backplate so frontplate and backplate content are displayed in sync. This is the default.

No reprojection
Frontplate is delayed along with the backplate content but is not reprojected. This is used for frontplate content that aligns more with the filmed content.

Minimal latency
Backplate image and frontplate images are taken from two different points in time. This results in the latest frontpate frame being displayed.

Frontplate composite order
Defines how content assigned to frontplate is ordered in the composite stack.
Colour LUT

Defines which LUT is applied to the camera. The disguise software has a number of LUT files built in.

Colour profile

Defines which colour profile the camera is. The disguise software has a number of colour profiles built in.

Hold output

Hold output freezes the current frame of the camera output. This is a toggle setting.

Hierarchy

Parent

Shows the parent object (this is usually the stage, so the object will be offset from the origin point of the stage).

Children

Shows children of the object.

Click Add child to create a parent/child relationship between this object and another.

Click Parent to stage to re-parent the object back to the stages origin point.
Anti-aliasing

Anti-aliasing can smooth out jagged edges caused by aliasing. The higher the number the smoother aliasing, but at more of a performance cost.

Background colour

Changes the background colour of the scene. In schematic this will be the background colour.

In Lux this will be the base colour of the environment and is affected by lighting. For example if the ambient light is 0 the background will be black regardless of the colour.

In heatmaps the background is always black.

Renderer

**Schematic** - Uses the Schematic renderer.

**Lux** - Uses the Lux renderer.

**Heatmap** - Uses the Heatmap renderer.

Light (only available for Heatmap)

**Luminance** - Luminance - The intensity of light emitting from an object or surface per meter in a given direction, measured in nits (cd/m²)

**Illuminance** - The density of incident light in lumens hitting a surface per meter, measured in lux (lm/m²).
Lux settings

White point - Camera exposure in lux.
Ambient occlusion power - Power of the ambient occlusion effect in the lux renderer. 0 disables it.
Glow brightness - Impact of glow effect in lux renderer.

Near & Far clipping plane

Any geometry closer than the near clipping plane or further than the far clipping plane will clipped (not be rendered). The values are in meters.
Reducing the range of these values can improve shadow quality and reduce shadow acne.

Face Cull Mode

Used to temporarily disable rendering of faces with normals pointing to the front or to the back

Camera type

Perspective - Perspective view.
Orthographic - Orthographic view.

Orthographic scale

Orthographic Camera has no concept of depth and distance from the scene. A zoom-like effect can be achieved by adjusting the Orthographic Scale, but this setting is independent from Zoom in the equivalent Perspective Camera.
Visibility

Measurements

Hidden (default) - hides measurements from view.
Visible - shows measurements in view.

Projectors

Hidden - hides projector from the view
Beam - Shows projector beam only
Wire - Shows projector wireframe only
Beam and wire (default) - Shows projector beam and wireframe.

Labels

Projectors - Labels for projectors only.
Projection Surfaces - Labels for projection surfaces only.
LED Screens - Labels for LED screens only.
DMX Screens - Labels for DMX Screens only.
DMX Lights - Labels for DMX Fixtures only.
Props - Labels for props only.
Cameras - Turn on or off the visibility of Cameras.
Camera Labels - Labels for cameras.

People

Visible (default) - People are visible in the view.
Hidden - People are hidden from the view.

Venue

Visible (default) - Venue is visible in the view.
Hidden - Venue is hidden from the view.

Tracked points

Visible - Tracked points from Blacktrax are visible in the view.
Hidden (default) - Tracked points from Blacktrax are not visible in the view.

Tracked points labels

Visible - Tracked point labels are visible in the view.
Hidden (default) - Tracked point labels are not visible in the view.

Movement speed

Visible - Movement speed is visible in the view.
Hidden (default) - Movement speed is not visible in the view.

Floor

Visible (default) - The floor is visible in the view.
Hidden - The floor is not visible in the view.
Reference Points

Visible (default) - Blacktrax reference points are visible in the view.
Hidden - Blacktrax reference points are not visible in the view.

Reference Points Labels

Visible (default) - Labels for Blacktrax reference points are visible in the view.
Hidden - Labels for Blacktrax reference points are not visible in the view.

Physical

Video in

Defines the live video input this camera is linked to.

Live action position marker

Used as a reference to re-align filmed content when using virtual zoom.

Spatial calibration

Used to create spatial calibration configurations

Video receive delay (sec)

Roundtrip latency time calculated from calibration process.
Spatial Tracker delay (sec)

Amount of time in seconds between the camera frame arriving and the tracking data associated with it arriving. This is calibrated automatically.

Lens Tracker delay (sec)

Allows for adjustment to the time needed to sync between lens and tracking data.

Lens

Overscan

Defines the overscan ratio. Default is 1. Setting to 1.1 would mean an overscan of 10%. The higher the overscan, the higher the performance impact.

Overscan resolution

Overscan resolution is the readout value in resolution of the overscan values.

Aspect ratio

This is a readout value of the final aspect ratio of the camera.

Focal length

This is a readout value of the calculated focal length of the camera.
Sensor width
Defines the sensor width.

Centre shift
Offset of the focal point of the lens from the centre of the light sensor, measured in mm.

K1, K2, K3

k1 - First order radial distortion coefficient, as measured.
k2 - Second order radial distortion coefficient, as measured.
k3 - Third order radial distortion coefficient, as measured.

Prediction

Camera imaging delay (sec)
The amount of time between the disguise software outputting the frame and capturing the image. Prediction is turned off when this value is zero. If you want to turn prediction on, increase this value until the image looks good. Do not increase beyond the Video Receive Delay value.

Camera imaging delay prediction samples
How much of the history of the tracking data to predict into the future. Expressed in frames.
Camera imaging delay prediction terms

How many terms we used to model the movement. Two terms is linear. Three terms is a cubic prediction.
Visualiser renderer overview

The disguise software has three visualiser render modes. Schematic, lux and heatmap.

In r15, the visualiser rendering mode was changed to encompass two additional new modes. Heatmap and lux. Schematic was the default rendering mode for the visualiser before r15.

Schematic

Effectively communicate the technical aspects of your creative concept to the whole production team. View your stage, storyboards, and test content from any angle in the real-time 3D stage simulator.

For more information see Schematic.

Lux

Use as a communication tool to bring together both the technical planning and creative vision of your project. Lux now includes ambient occlusion, directional light, camera exposure and enhanced projector simulation. Communicate ideas clearly and quickly to get everyone, from clients to operators, inspired and on board.

For more information see Lux.

Heatmap

Heatmaps for projector studies using evidence based calculations - use the orthographic camera mode to ensure accuracy in positioning of objects. Export your projector study to provide paperwork evidencing the proposed setup against a target light level.

For more information see Heatmap.
Image of the three different render modes, side by side.
Heatmap

This renders the view using colour coding for different lux levels, rather than applying the glow filter.

Overview

The heatmap render mode is primarily used for visualising light levels within the scene.

All projectors and LEDs output white content in this mode. The graph on the right goes up to twice the camera’s exposure max-lux level, so use that to scale it if necessary. This is the mode that will be used for Projector studies.

Please note: Modern rendering modes such as Heatmap depend on appropriately powerful GPU choices. This feature requires significantly more GPU power than other rendering features in the disguise software. For project specific guidance, please contact the support team.

Workflow

- Setup your project as normal by adding projectors, screens, venues & props.
- Change your visualiser camera to Heatmap mode, or make a new Camera to assign to a feed output.
- Define your target lux in the heatmap editor.

Example
Enabling heatmap

1. Open the stage editor by right clicking **Stage** from the dashboard or by right clicking the **floor** in the visualiser.

2. Expand the Cameras tab, and right click **Visualiser camera** to edit the visualiser camera settings. Alternatively, you can edit another camera and output that to the feeds.

3. In the camera editor, expand the **Renderer** tab and set the render mode to **Heatmap**.
You are now viewing the heatmap visualisation of the photometric quantity specified by the Light field under the render tab within the camera editor.

The key shown in the heatmap editor shows the colour scheme and scale of the heatmap.
Editing heatmap properties

1. Open the heatmap editor.
2. Set the target value to be the preferred surface illuminance or luminance that you want to achieve.

Viewing heatmaps through cameras

1. Position the camera in the visualiser.
2. Left click Feed from the dashboard.
3. Right click on the camera feed at the top of the feed scene and left click Add feed rectangle or alternatively hold ALT, left click and drag an arrow from the camera feed to an available output head.
Heatmap properties

**Target lux/nits**

The value that determines the centre of the colour scheme, which is green.

**Heatmap min**

Manually sets the minimum value of the scheme.
Heatmap max

Manually sets the maximum value of the scheme.

*Please note:* When you change target, this will change the minimum & maximum values to be zero for minimum and double for maximum of the target value.
Lux

Lux is an enhanced rendering mode designed to simulate a more realistic impression of how a show would look.

⚠ Warning: Lux is an intensive performance mode and should not be used for playback!

Overview

Lux is most useful for showing clients a more realistic impression of how a project would look. This includes ambient lighting, shadowing of objects being hit by projectors based on physical lighting and scene settings. This allows you to make more technical & creative decisions in advance before going onsite.

Please note: Modern rendering modes such as Lux depend on appropriately powerful GPU choices. This feature requires significantly more GPU power than other rendering features in the disguise software. For project specific guidance, please contact the support team.

Features of this rendering mode

- Physical lighting values
- Ambient light
- Ambient occlusion
---

- Projector shadows
- Bloom effect
- Diffuse maps

**Limitations of the rendering mode**

- Lux does not support transparent objects, that includes transparent pixel and population masks.
- Lux does not support blend modes other than 'Over', i.e. any non-opaque modes.

**Scene Lighting**

Lux aims to simulate light in an accurate way, as such the Scene is absent of light by default and only Projectors and Scene Lights can be used to introduce lights.

All objects will be affected by the Ambient Brightness in lux of the scene where ever they are. This must be accounted for when setting Exposure and is required to allow props to be seen.

The Ambient Colour effects the colour of this light. So a high Ambient Brightness and a red Ambient Colour will give everything a red tint.

Directional Lights can be used to add a sun-like light source to the scene, it has no source point but casts lights at a constant angle into the scene in parallel rays. This improves the appearance of a 3D object by providing shading to surfaces depending on their exposure to the Directional Light, much like an object in the real world. Directional Lights are visualised in the Scene at the specified Offset but this value has no effect on the lighting and visual results as the Directional Light has no source point.

The Reflectivity of an object can be set to adjust the amount of light reflected by the surface. This number is based on a perfect lambertian surface and represents the fraction of light that will be reflected. It does not indicate screen gain or simulate colour response.

⚠️ **Warning:** Scene lighting does not cast shadows.
**Screens**

All objects will be affected by any projectors that hit them. The brightness is determined by the projectors lumen value. Projectors will cast shadows.

LED Screens, DMX Screens have physical luminance values based on NITS but do not cast light on other objects in the scene.

⚠️ **Warning:** DMX Lights are not currently supported as lights.

**Ambient Occlusion**

Ambient Occlusion is an effect that emulates the occlusion seen in dark corners of objects in the real-world but is not related to any real lighting. It is only affected by the ambient light in the scene.

Ambient Occlusion Power can be changed via the Camera Editor and turned off to improve performance.

**Enhanced Projector Footprints**

Projector brightness is specified in Lux which is used for simulating the brightness of an image in Lux and also the resulting Lux readings in Heat Map mode.

In Lux mode all objects will cast shadows when in the projector frustum for the use of pre-visualisation and checking of sight-lines. These shadows are not shown in the feed outputs.

**Diffuse Maps**

A Diffuse Map can be added to Projection Surfaces or Props to simulate how the projected light will interact with the surface. This is useful when projecting on surfaces that are not ideal for projection such as buildings with a stone surface or painted set pieces where the light that is reflected may be different than imagined due to the interference of the surface.
Workflow

- Setup your project as normal by adding projectors, screens, venues & props.
- Define Ambient Brightness & Colour.
- Change your visualiser camera to Lux mode, or make a new Camera to assign to a feed output.

Example

Enabling lux

1. Open the stage editor by right clicking **Stage** from the dashboard or by right clicking the **floor** in the visualiser.

2. Expand the Cameras tab, and right click **Visualiser camera** to edit the visualiser camera settings. Alternatively, you can edit another camera and output that to the feeds.

3. In the camera editor, expand the **Renderer** tab and set the render mode to **Lux**.
You are now viewing a Lux rendering of the scene based on the values defined in the visualiser camera.

For more information, see the Cameras topic.
Setting scene lighting

1. Open the stage editor by right clicking **Stage** from the dashboard or by right clicking the **floor** in the visualiser.

2. Expand the Scene tab.

3. Adjust the ambient light and colour values to suit.

```
 stage 1
Scene
Floor size 25, 25
Floor position 0, 0, 0
Floor plan Venue None
People 1
Ambient Brightness 1000
Ambient Colour
Lights + –
LED Screens
DMX Screens
DMX Lights
Projection Surfaces
Projectors
Cameras
Actions
```
Viewing Lux through cameras

1. Position the camera in the visualiser.

2. Left click **Feed** from the dashboard.

3. Right click on the camera feed at the top of the feed scene and left click **Add feed rectangle** or alternatively hold **ALT**, left click and drag an arrow from the camera feed to an available output head.
Schematic

Schematic render is an operational rendering mode that provides unlit, but accurate rendering of content in the stage visualiser.

Overview

Previous to r15, the schematic rendering mode was the only mode available in the stage visualiser settings. It is intended to be used during playback, sequencing and when maximum performance is needed.

Effectively communicate the technical aspects of your creative concept to the whole production team. View your stage, storyboards, and test content from any angle in the real-time 3D stage simulator.

Workflow

- Setup your project as normal by adding projectors, screens, venues & props.

- Change your visualiser camera to schematic mode, or make a new Camera to assign to a feed output.

Please note: Schematic is the default enabled mode of the three renderers.
Example

Enabling schematic mode

1. Open the stage editor by right clicking **Stage** from the **dashboard** or by right clicking the **floor** in the visualiser.

2. Expand the Cameras tab, and right click **Visualiser camera** to edit the visualiser camera settings. Alternatively, you can edit another camera and output that to the feeds.

3. In the camera editor, expand the **Renderer** tab and set the render mode to **Schematic**.
<table>
<thead>
<tr>
<th>Settings</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Visibility</td>
<td></td>
</tr>
<tr>
<td>Renderer</td>
<td></td>
</tr>
<tr>
<td>Renderer</td>
<td></td>
</tr>
<tr>
<td>Light</td>
<td></td>
</tr>
<tr>
<td>White Point</td>
<td>200</td>
</tr>
<tr>
<td>Ambient Occlusion Power</td>
<td>1</td>
</tr>
<tr>
<td>Glow Brightness</td>
<td>1</td>
</tr>
<tr>
<td>Camera Type</td>
<td></td>
</tr>
<tr>
<td>Actions</td>
<td></td>
</tr>
<tr>
<td>Hierarchy</td>
<td></td>
</tr>
</tbody>
</table>
Viewing schematic rendering through cameras

1. Position the camera in the visualiser.

2. Left click **Feed** from the dashboard.

3. Right click on the camera feed at the top of the feed scene and left click **Add feed rectangle** or alternatively hold **ALT**, left click and drag an arrow from the camera feed to an available output head.
AR workflow

Overview

AR content from a generative content engine, such as Notch, can be placed in the 3D scene overlaying existing video content or a live video input using this workflow.

AR Workflow

1. Create an MR set
2. Create a virtual camera(s)
3. Create an LED screen on the stage, check that the Render Layer (located in the Output tab of the layer) is set to 'On Stage'
4. In the MR Set properties editor, add the LED surface to the MR set, create an indirection controller, and select the virtual camera(s) as the resource.
5. Create a Notch Layer and select the desired Notch block that you want to use as an AR object
6. Create a Spatial mapping for the Notch layer and add the virtual camera(s) as a screen
7. Set the Camera plate of the Spatial mapping to Backplate
   At this point, you will see the AR object in the Camera transmission preview and the MR Set
preview windows
Virtual Reality visualiser is an OpenVR / Steam compatible visualiser mode allowing the user to enter and navigate through the scene.

VR was designed to primarily work with disguise designer machines, utilising GPU’s such as GTX 1060 & higher (or other manufacturer equivalent).

As such, gx range machines are well suited for VR work. However, Pro & Plus ranges are not recommended for VR usage as they are not as well suited to the feature as newer machines. However VR does also work on these machines, but performance may vary.

We have tested HTC Vive headsets on the disguise software and they are known to work. However, there are some known issues.

**HTC Vive**

- The machine needs to be restarted after first installation, otherwise the headset will not be detected
- All EDID’s need to be de-emulated otherwise the headset will not work
- The box needs to be plugged into a USB 3 port.

**Results - HTC Vive**

GX 2 - Both the HDMI and DisplayPort VFC cards are supported.
GX 1 - Both the HDMI and DisplayPort VFC cards are supported.

**Please note:** Modern rendering modes such as VR depend on appropriately powerful GPU choices. This feature requires significantly more GPU power than other rendering features in the disguise software. For project specific guidance, please contact the [support team](mailto:support@disguise.com).

So far there is no plan to add more UI functionality; it’s just for viewing and walking around.

You need to have the Steam system running; you should be able to get to the point where the headset and one controller are reporting green and working, and the headset should show the Steam basic
scene. Once that’s working, if steam is installed, and the VR equipment is connected - when starting a disguise project, you will automatically be placed into VR mode.

Once that’s working, if steam is installed, and the VR equipment is connected - when starting a disguise project, you will automatically be placed into VR mode. The visualiser shows you the position and orientation of the headset in stage space (small white cuboid about the same size as a headset, with an orange line down to the floor). It also shows you where the controller is pointing (white line when it hits the floor level, blue line otherwise).

Navigation - the VR user navigates by pointing the controller at the floor location they want to go to; it goes white; they click the button and they jump there. The property **movement speed** in the VR Navigator window can be set to zero (jump immediately) or some non-zero number, which moves the VR world in a straight line on a linear path. This gives you a little ‘jerk’ at start and end but doesn’t seem to result in nausea. It’s less confusing for new users, but may be annoying for experienced users.

The Virtual reality navigator also lets you set up the position and orientation of the scene relative to the physical world. The VR experience keeps this orientation constant to avoid users getting lost. They still get lost of course, and when they do you can hit the **reset** button to bring them back to the center.

**Opening the virtual reality navigator**

1. Open the **stage editor** by right clicking **Stage** from the **dashboard** or by right clicking the **floor** in the visualiser.

2. Select the **Cameras** tab

3. Left click **Virtual reality navigator** and then you can see what the VR user sees.

**Warning:** Navigation is restricted to floor level. We need to be able to move to the nearest horizontal surface on whatever venue / stage you’re on.

**Warning:** There’s no ‘reverse’ button; the workaround is just to point behind you and hit the button.

The image shows the basic elements of VR functionality.
Using HTC Vive for virtual camera tracking

As of r17.2, virtual cameras can now be tracked by the headset or controller of the HTC Vive

To assign HTC Vive as tracking source for a virtual camera:

1. Connect your HTC Vive to your machine and ensure the device is working in SteamVR.
2. Launch your disguise project version r17.2 or later.
3. Create a new virtual camera in your project file by right clicking on the Stage menu.
4. Hit the + icon under the Cameras tab and input the name and specifications of your camera.
5. Open the camera's menu by right clicking on its name from the Stage menu, or on the icon of the object in the visualiser.
6. Under the first tab called Settings, right click on the Tracking source field.
7. The Vive's headset and any active controllers will be listed here; left clicking on one will assign the position of the camera to it.

8. The Offset and Rotation fields of the virtual camera's menu will turn green when receiving a working signal.
Venues and props overview

What is a venue?

A stage can contain multiple venues. This is useful if you wish to create a range of stage designs, each venue representing a different stage design option, allowing you to quickly change from one design to another (during a client meeting for example). When you create a new project there is already one venue prepared called venue 1. It only contains one prop called Arena.

What is a prop?

The venues are constructed from props. Props are objects which form the building blocks of your venue. A new project will by default contain a series of props contained in the Mesh object library. These props consist of a range of different buildings, a 3D man and a 3D woman.

Props are .obj meshes. If you want to use a prop other than the standard props available in the disguise software, you can import this prop. Please see the Mesh section within the Editing props sub-chapter for information on how to import a prop mesh into the software.
Venues and props overview

What is a venue?
A stage can contain multiple venues. This is useful if you wish to create a range of stage designs, each venue representing a different stage design option, allowing you to quickly change from one design to another (during a client meeting for example). When you create a new project there is already one venue prepared called venue 1. It only contains one prop called Arena.

What is a prop?
The venues are constructed from props. Props are objects which form the building blocks of your venue. A new project will by default contain a series of props contained in the Mesh object library. These props consist of a range of different buildings, a 3D man and a 3D woman.

Props are .obj meshes. If you want to use a prop other than the standard props available in the disguise software, you can import this prop. Please see the Mesh section within the Editing props sub-chapter for information on how to import a prop mesh into the software.
Creating and removing props

Creating a prop for a venue

Right-click **stage** from the dashboard (bar at the top of the screen). This will open the Stage editor.

Right-click **venue**. This will open up the Props menu which displays a list of all of the active props in the currently active venue.

A new project will by default contain one venue called **venue 1**, which itself will contain one prop called **Arena**.

Left-click +. This will open a Props manager that lists all of your props. It is important to note that the props in this list can be added to either, some, or all of your venues; you do not have to limit
a prop to only one venue.

![Props manager opened by left-clicking + from the Props menu]

- Left-click a prop from the Prop manager to add it to the currently active venue, in this example **venue 1**. You can now edit the prop.

- If you wish to create a new prop, type the name of a new prop in the **new prop** text field and hit **Enter**. This will create the prop and add it to the currently active venue.

- Repeat these steps for each prop you wish to add to the currently active venue.
Example of a complete venue imported into the Stage for the U2 Vertigo Tour.

Removing a prop from a venue

- Left-click and drag a prop, from the prop menu to the `-` button. This will remove the prop from the prop menu, although the prop itself will not be deleted: it is still listed in the Prop manager. Notice how this change has been updated in the Stage Visualiser.
Permanently deleting a prop

**Warning:** you can never permanently delete a prop if its active or referenced to from anywhere in the project.

Left-click and drag the prop you want to delete from the Prop manager to **trash** (represented by a trash-can icon).

*Left-click and drag the prop you want to delete, from the Props manager to Trash.*
Right-click **trash** and select **empty trash** to permanently delete the prop from your stage.
Editing props

Editing a prop

- Right-click a prop directly in the Stage level, or

- Right click the prop's name from the list of props in the Props Manager menu. Read the Creating/Removing props sub-chapter above to learn how to open the Props Manager/Props menu.
The Prop editor is used to edit a prop, in this example a prop called Arena.

Prop properties

Offset
Offset values are expressed in meters and are based on the original prop position along the x, y, z in the 3D application it was exported from.

**Rotation**

The rotation value is always based around the origin point in the 3D application it was exported from. For example, if you export a cube which is offset 10m to the right in Blender it will show up 10m to the right in the stage. When you change the rotation values it will rotate around the origin in disguise, rather than the local origin of the object.

To avoid this, you should export the object at the origin 0,0,0 of your 3D application. Then offset it 10m in the stage. When you rotate the object in disguise it will now rotate around its local axis.

**Scale**

The scale property is a multiplier. It multiplies the prop size with its original size as exported from the 3D application.

**Mesh**

This points to the Mesh file that defines the shape of your props. Selecting this property will open the Mesh object library, which shows all of the Mesh .obj files saved on your local hard-drive in the Mesh folder.
Mesh property being used to update the mesh of a prop from the Mesh object library

To change a prop mesh:

- Left-click `mesh` to open the Mesh object library.
- Left-click the mesh you want to replace the currently active mesh. This will update the prop mesh.

If you want to use any other shape than the standard prop meshes provided in the disguise software, you will need to use a UV-mapped `.obj` file of the prop.

See the Placing media files for a project sub-chapter to understand where to place a mesh `.obj` file and how to access it in the software. Also save the file to a supported file format.

**Diffuse Map**

This points to the Texture file that defines a prop's texture. You can use this property to create and apply a texture to a prop, allowing you to visualise your stage as realistically as possible. Selecting this property will open the Texture object library, which shows all of the still image files saved on your local hard-drive in the `DxTexture` folder.
Diffuse Map property being used to apply a still image to a prop from the Texture object library

To apply a texture to a prop:

- Left-click **texture** to open the Texture object library.

- Left-click the texture you want to apply to the prop.

If you want to use a texture other than the standard still images provided in the disguise software, you will need to use a custom still image file.

See the **Placing media files for a project** sub-chapter to understand where to place a custom still image file and how to access it in the software. Also save the file to a **supported** file format.
Please note: to apply a texture to a prop you need to UV-map the mesh of the prop.

Flat colour
This property defines the color of a prop.

Wire colour
This defines the color of the wire-frame of the Prop. In order to see the wire-frame you need to set the props Display Mode to either flat+wire or wireframe.

Blend mode
This controls how the props contents are composited with the rest of the objects in the stage.

**Over:** should be used for non-transparent props.

**Add:** should be used for screens that are totally invisible when there is no content on them.

**Alpha:** should be used for screens that are partially transparent.

Display mode
This controls the different shading types of the prop.

**Hidden:** does not render the prop.

**Flat:** only renders a flat shaded prop.

**Wireframe:** only renders a wire-frame of the prop.
**Flat+wire**: renders a flat shaded prop with highlighted edges.

**Wireframe cleanup**

This defines the level of detail of the wire-frame rendering. The default value is 20. The lowest value is 0 which will render the most wire-frames, and the maximum value is 255 which will render the least wire-frames.
Creating and switching venues

Creating a venue

Right-click stage from the dashboard (bar at the top of the screen). This will open the Stage editor.

Left-click venue. This will open up the Venues manager which displays a list of all of the venues you have created. The venues listed in the Venues manager are the same regardless of which stage is active, allowing you to share venues in multiple stages. A new project will by default contain one venue called venue 1.

Type the name of the new venue in the new venue text box, in this example venue 2, and hit Enter. This will create the venue and add it to the list of venues in the Venue manager. You can now edit the venue by adding or removing props.

Process used to add a venue to the currently active stage, in this example Venue 1
Switching from one venue to another

- Left-click the venue from the Venue manager you want to switch to. The venue will then be updated in the currently active stage.

Deleting a venue

- Left-click and drag the venue you want to delete from the Venue manager to trash (represented by a trash-can icon).

- Right-click Trash and select Empty Trash to permanently delete the venue

**Warning:** you can never permanently delete a venue if its active or referenced to from anywhere in the project.
Editing venues

Unlike screens, venues do not have a series of properties which can be edited. Instead, venues contain props, which are objects that form the building blocks of your venue. Props have a series of editable properties including the option to add your own textures, all of which are explained in the Editing props topic.