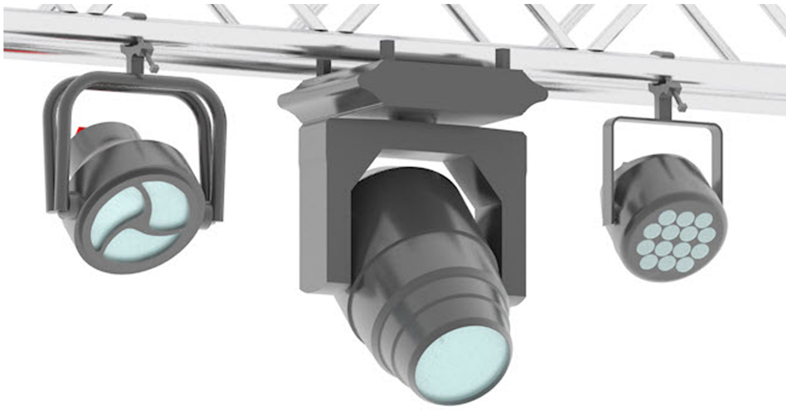
Lighting Device

Please read the **Vectorworks Basic Style Guide** first. It will cover more common aspects that you need to consider in creating your resources, and it is a companion to this guide. The information found there should be understood before using the more specific topic guides like this.



Introduction

This section will provide the guidelines in developing lighting device (aka lighting instruments or fixtures) symbols. It is best to read completely through this guide to get a better understanding of the process before beginning. There is also a development file attached to this guide with examples of properly made symbols and all necessary support material.

Contents

Getting started

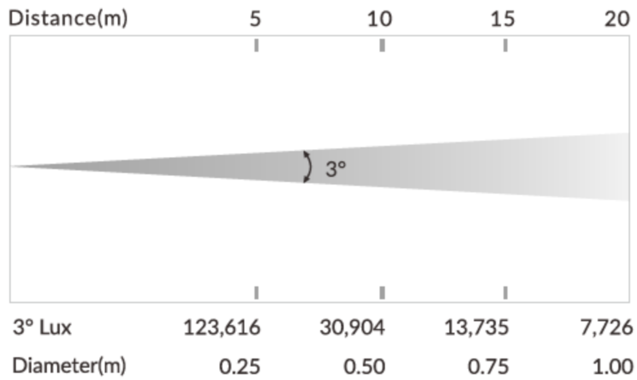
Before proceeding you should have basic Vectorworks knowledge on modeling and creating symbols. This includes understanding the Vectorworks Basic Style Guide.

Also consult the Vectorworks Help section on the Lighting Device, here Vectorworks 2025 Help

Collecting source information

Collect source materials for the device that you wish to create. Gather necessary drawings, CAD files and cut-sheet information.

- **Manufacturer drawings** - these will provide dimensions (side, front, back, and top) 2D. Isometric is useless because you cannot dimension or draw from them. DWG files seem to the best to work with as they can be easily scaled properly. As a secondary option, imported PDFs can be set to “snappable” during import. As a last resort you can try to use product photos if you know dimensions and the photos are in a favorable orientation.
- **Photometrics for lighting devices** – This is not the IES file as Spotlight devices do not use this data. Look for a chart showing measured lux at a given distance with a beam spread.



Example of Photometrics Showing Lux At Given Distances

- **Beam spread information** – This goes along with photometrics but shows how much the light spreads in degrees in both the vertical and horizontal directions. Some lights can have oval lenses so the vertical and horizontal values will be different. Some lights also have different lens options for the same device so all variants will have to be made. The beam spread will also cause the lumens to be different, i.e. the inverse square law of lighting.
- **DMX protocols for fixture** – Most fixtures have multiple modes and the number of channels vary so it is important to know all the options. This chart will be useful if a GDTF file needs to be developed as it will show all mappings.
- **Overall Wattage**- Important for power planning, sometimes the source wattage of the LED for LED fixtures will be listed but look for an overall wattage or max power draw. It is important to know the maximum wattage a device uses while in use.
- **Lamp Type and Wattage** – Most new devices are moving towards LED now but some still use incandescent or mercury arc lamps so it is important to list the lamp code if it is not LED. Also, look for the lamp wattage because some devices have multiple LEDs so the “source wattage” may have to be calculated. Some conventional devices can also use different lamps so it is important to show what wattage lamp is in the fixture for power planning purposes and inventory.
- **Weight** – This is the total weight of the device and any rigging hardware that is built into the symbol such as C-clamps. The value can be in either kilograms or pounds, just be sure to include the proper units in the field of the record, which will be explained later. Use lb for pounds and kg for kilograms.
- **Frame Size** (if applicable) – Some devices will have these and others will not. Only enter values if present, there are individual fields for metric and imperial but be sure to also have units.

Check if the device is on the GDTF-Share. If it is there, see if there is a way to use the geometry as a starting point to save time. Of course, GDTF is open source and there may be errors so data and dimensions will have to be verified.

Home | GDTF Share

Start a file

The library should be made using the *Lighting Device Template.sta* which contains all the correct records, classes, colors, accessories, and textures currently in use. Note, this document is in metric as those are the easiest units for development and most manufacturer CAD files originate in millimeters.

Use the Vectorworks template file below to start your library. It includes example lighting devices that show properly modeled and setup symbols for reference. These symbols are not too overly detailed but still convey enough of a difference from similar fixtures to be distinguishable.

Lighting Device symbols

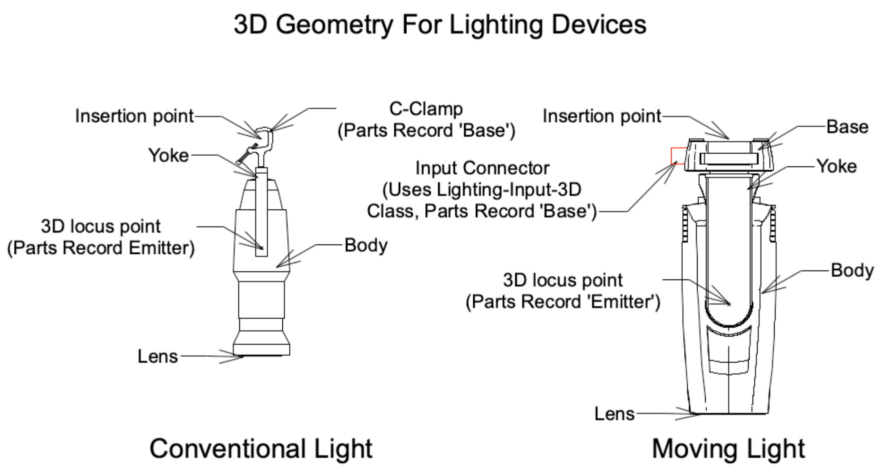
The most important thing when creating lighting devices is to have a nicely modeled hybrid (2D/3D symbol in as small a file size as possible. The device should be unique enough to distinguish itself from other similar looking devices but not so detailed that it could be used to manufacture. For the hybrid symbols there will be a 3D component containing a simplified model, and a 2D component with simple linework and minimum details for top view plan use.

3D component creation

The 3D representation of the device should present an accurately sized model of the real world version. It should be simplified if highly detailed. Avoid drawing minute details like grommets, fins, louvers, cords, knobs, input/output sockets, or shutters as part of the 3D symbol. Every detail you draw will add to the complexity of your scene as you try to render. While a couple extra boxes might not seem like a lot, when you multiply it by tens or hundreds of devices the polygon count increases quickly and performance decreases quickly.

3D parts and positioning overview

When planning the symbol there is a combination of parts and positioning to consider for the 3D model setup inside a symbol depending on the type of lighting device:



3D geometry for lights, shown from a right view (Note: that the body faces down in 3D)

Component parts

The 3D geometry will consist of a combination of several parts: body, lens, 3D locus (for rotation and light emitter origin), a yoke, and base. All devices will have a body and a lens part as a minimum.

- **Lens** – Always present. This is the part of the light that will visibly illuminate when the light is turned on by the user.
- **Body** – Always present. This is the part of the light that will **tilt** up/down when focused, it will also contain the lens.
- **Yoke** – This is the part of the light that will **pan** left/right when focused and will pan the body geometry with it as it moves. This is on more advanced lights.
- **Base** – Typically on moving lights. This is the part of the light that will stay stationary regardless of pan or tilt commands. Primarily used in moving lights and strip lights that have trunnions. More detail will be explained below.
- **3D locus** - For rotation and light emitter origin

- **Clamp** - When required, it becomes the insertion point origin.
- **Input connector** - Plug location and indicates the back of the light.

Each part above will also have a Parts record added to its geometry. The part type determines which Part record to select and attach. More on this will be covered later in the guide.

The *Lighting Device.sta* template file includes the required C-Clamp that is used for non-moving lights that are given a clamp by default. This clamp is already placed at the proper location for an insertion point and just needs to be added to the 3D component.

Positioning the 3D component

The combined 3D components, that make up a lighting device, will also need to be arranged in relation to 0,0,0 in the symbol component and set up for one of three types of typical insertion.

- Hanging location – The device directly mounts where the 0,0,0 point is inside a symbol's 2D and 3d component.
- Hanging location using a clamp – If a clamp is used for mounting, the center of the clamp jaw should be positioned at coordinates (0, 0, 0). within the symbol's 2D and 3D components.
- Base mount fixture – Sits on a surface like a floor. If the device has feet to elevate the base. the 0 z-axis is will be at the bottom of the feet.

All of this will be covered more below.

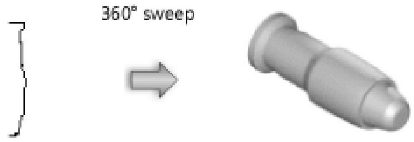
Creating the component parts

The 3D component parts should also be a 3D solid derived from Vectorworks native geometry. Once complete the geometry should be converted to a Generic Solid to reduce file size and remove all modeling history. This should only be done if solid addition/subtraction, mesh, NURBS, or Subdivision modeling was used. But it is not necessary for on sweeps and extrudes.

Modeling done with sweeps or extrudes should remain as their native base geometry because making these types of objects into Generic Solids will increase file size.

The goal is to make the file size as small as possible for the the fixture and no individual fixture should be more than 600 kb in size.

Quick tip - An easy way of generating a reasonable 3D model is to sweep the 2D Portion of the symbol. Make sure you keep the segment angle of your sweep at 15°. See below.



Simple Profile Turned into 3D Model With Sweep Command

Next, each of these lighting parts will be discussed in more detail.

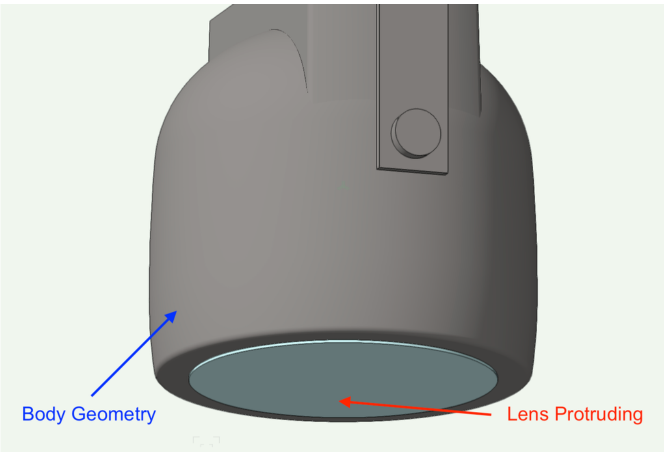
Creating the body and lens

All lights will have a body part. The body is what tilts within the yoke in 3D when the fixture is focused. The easiest way to create the body geometry is to sweep a polyline made using a front view of the body, this is mentioned above. This will work for a good portion of lights as most are symmetrical around the center line. There are some lights that are more flat (or oblate) shaped so the use of subdivision modeling or additive/subtractive solids is best for these. This type of modeling is discouraged, as it can cause file size and poly count to grow but you can use it if it's unavoidable.

When complete the geometry should get the 'Default Instrument Texture RT' applied and have the fill set to Cool Gray 90%. It should then be placed into the yoke in the proper location relative with the front of the body facing the -Z axis when viewed in Top/Plan. Look for a bolt head in the source file from the manufacturer that attaches the body to the yoke. This will determine the pivot point where the body and yoke geometry should connect.

The body also contains the lens object(s). The lens creates a colored lens glow for renderings and is used as the light emitting source when creating a GDTF file. The lens must be made from either an extruded circle or square and should be extruded a distance of 3mm.

If the light is a more advanced type with multiple lenses, draw each lens as a circle, and extrude all those circles in the same operation. The 3D line thickness should always be 0.05mm. Having a single extrude is necessary for the gobo and lens glow mapping to work correctly. The fill color for the lens is included in the template and is labeled as ‘Default Lens Color’, this should be applied to the extruded circles.



Example light device showing the lens

The lens should be placed to match with the source file, and it should protrude from the body geometry halfway. This will allow it to be seen and the light to properly project. The ‘Default Instrument Texture RT’ must also be applied to the lens.

Group the lens and the body geometry with the Group command once the lens is properly placed.

Creating the yoke

Most lights will have yokes, with the exception being some uplights and wall washer strip lights with trunnions. The yoke is the portion of the light that allows the body to pan/rotate when focused. The yoke geometry will rotate around the Z axis of the symbol and move the body with it. As a result, it is important to have the correct placement in relation to the insertion point.

The general shape of yokes resembles a horseshoe with the closed end attaching to the symbol base and the open end encompassing the body of the fixture. The simplest way to draw the yoke is by an extruded polyline from a front view. This will show the best representation of the horseshoe shape. There are some yokes that have circular ends when viewed from a side view. In these instances it is best to draw the polyline from a side view and then extrude to the full width of the yoke. Then, from a front view the horseshoe shape can be created using a solid subtraction. As with the base, any use of the chamfer or fillet tool should be avoided.

Once complete apply the ‘Default Instrument Texture RT’ and choose a fill color of Cool Gray 90%. The 3D line thickness should always be 0.05mm. Be sure to note where the yoke will reside in relation to the base once the symbol is created. The yoke should be a single piece of modeled geometry and not within a group.

Creating the base

Not all devices will have a base and they are mostly reserved for moving lights as this is the location for most of the motors and the circuitry and does not move. This is the simplest geometry to create and can be an extruded rectangle or rounded rectangle in most cases.

There may be built in handles in the base and these are not necessary to draw unless they extend outside the yoke. In that case they need to be present to show they are occupying additional space outside of the base.

It is easiest to create the outer shape as a polyline and create the handle holes by using the Clip Surface command and then extrude the polyline. Sometimes there are rubber feet on the base for times when the light is floor mounted. These feet need to be included as part of the base and should be modeled as a sweep with a 15° segment angle. Try to avoid circles as this creates a lot of faceting and a larger poly count when exported to MVR or GDTF. The Chamfer and Fillet tools should also be avoided as this adds extra rendering time and file size. Once all components of the base are created, use the Add Solids command to combine them.

Then apply the 'Default Instrument Texture RT' texture and choose a fill color of Cool Gray 90%. Both the color and texture are included in the template file. The 3D lines thickness should always be 0.05mm. Once the lighting device is complete any geometry that is defined as the ‘Base’ will remain stationary, when focused.

Creating the input connector

This only applies to **moving lights** and **strip lights** have an input connector. The 3D geometry is a rectangle that is 100mm wide and 50mm long that is then extruded to 50mm to create an extrude. If the light is small and the 3D connector looks too large by comparison it can be made smaller, just make sure to maintain the appropriate proportion.

This geometry should be placed where the power and data connection inputs are on the fixture and can be found in the manufacturer CAD files.

The location of the connector varies depending on the type of light:

- Moving lights - has the connector on the base.
- Strip lights - has the connector on the body. Attach and include it in the Body’s group.

These will get assigned to the 'Lighting-Input-3D' class and the input connector will take the graphic attributes of the class making it display as a solid red box. The 3D line thickness should always be 0.05mm.

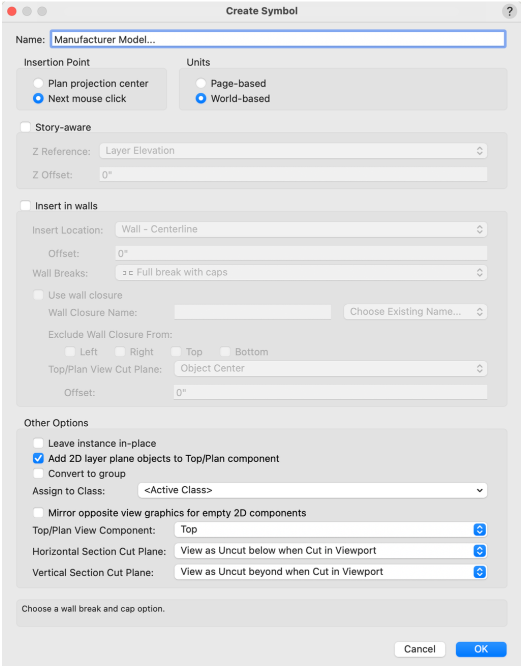
Creating the clamp

Since the clamp is already provided in the template, there is nothing to make! The clamp will be explained after the symbol is created.

Create symbol

Once all of the components have been modeled the symbol should be created using the ‘Modify>Create Symbol…’ command and named appropriately. In the Create Symbol dialog the following settings should be made:

- There is no container class needed for the lighting device symbols. All symbols should be set to use the ‘Active Class’ at insertion.
- Horizontal Section Cut Plane settings are set to ‘Uncut Below”
- Vertical Section Cut Plane settings 'Uncut Beyond'
- Story Aware and Insert In Walls should be unchecked as lighting devices do not work with those functions. (See Image)



Set up 3D geometry for insertion

Now that the symbol has been created, go into the symbol's 3D component to edit. It is very important that the components all be assembled so they can interact properly with each other. The center rotation of the yoke should be placed at (0,0) when in Top/Plan view, there is a snap point to assist this in the software.

From a side view the feet or the bottom of the base should be at the Z-height of zero so that all the geometry lies in the -Z portion of the symbol.

If there is no base and the light just contains a body, lens, and yoke, the yoke centerline should reside at the Z-height of zero. When a user adds a clamp as an accessory the program will lower all geometry to accommodate the added distance the clamp creates so that the insertion of the entire symbol will still be at (0,0,0).

Proper placement of 3D components when a symbol is complete. Note that the feet are placed at the Z=0 Line

Adding the clamp to the 3D symbol definition

If the device requires a clamp it can be added by editing the symbol in the Component Edit.

Conventional lights such as PARS and ellipsoidals require a clamp to hang and are shipped with one by default in the 3D component of the symbol. There are two examples of this in the *Lighting Device Template.sta*. Moving lights will not get a pre-installed clamp as the user must use the Accessory Insertion Tool to add their preferred clamp(s) themselves.

The template file already has the C-Clamp that must be used in the Symbol Parts folder and the clamp is already placed at the proper insertion point with all necessary records attached. The clamp is assigned to the Base in the Parts record. The clamp is already placed in the Lighting-Clamps class and this should not be changed.

To add a clamp, while editing the 3D component of the symbol select the "Symbol Insertion Tool" and choose the C-Clamp from the dropdown. Locate the (0,0,0) point of the 3D component and double click to insert at that point. Then switch to a Right or Left view and move all of the device geometry so that the yoke of the light is at the bottom of the clamp. See image.

Light shown with (left) and without (right) the C-Clamp. Note that both are at (0,0,0) insertion point

Attach the Parts record and textures to the 3D components

Once the 3D geometry is properly placed the next step is to attach the 'Parts' record, which is included in the template, to each component, as follows.

Here is a summary of the parts, the record setting to select in the Object Info palette, as well as whether a texture needs to be attached.

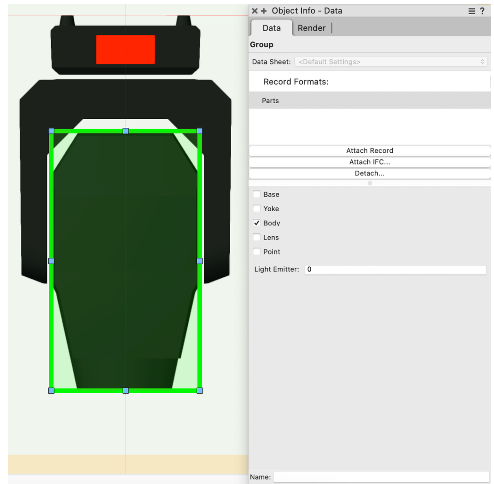
Component part name	**Record**	**Texture**
Body and Lens group	select "Body" for the group.	Default Instrument Texture RT
Lens	select "Lens"	Default Instrument Texture RT
Yoke	select "Yoke"	Default Instrument Texture RT
Base (when applicable)	select "Base"	Default Instrument Texture RT
Input connector	"Base" or "Body" - wherever attached	no texture
Locus	select "Point"	no texture
Clamp (when applicable)	pre-attached in the template file	Default Instrument Texture RT

To attach a record, right click on the geometry while in the Edit Symbol mode and choose 'Attach Record'. Then locate the Parts record in the Records resource folder of the template file, and click the checkmark next to it. Clicking OK will attach the record provided in the template.

Then, enter the body and lens group, then attach the Parts record and choose “Lens.” Exit the group. Select the lens and body group and attach the Parts record and choose “Body.”

Also at this time apply the ‘Default Instrument Texture RT’ to the group.

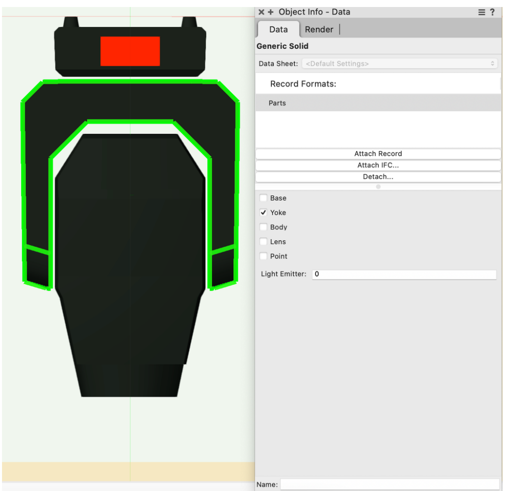
When completed the body should point down towards the -Z axis as that is the home position for setting focus.



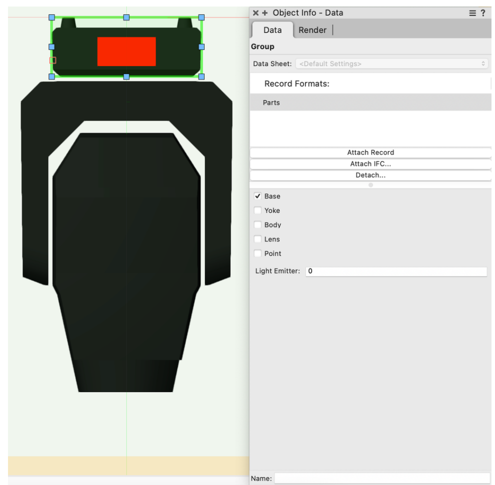
Body and Lens geometry - for a Moving Light (no C-Clamp) with Parts Record

Next, this should be repeated for the yoke, and the base. Below are moving light examples with the base and the yoke.

Then attach the red input connector (if a moving light or strip light) to the base.



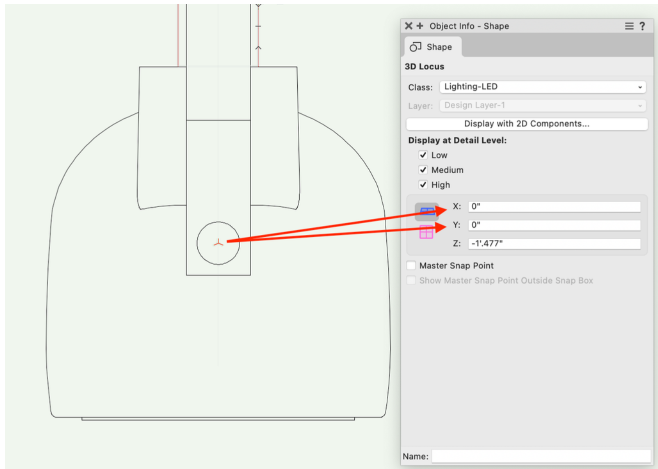
Yoke Geometry - for a Moving Light (no C-Clamp) with the Parts Record



Base Geometry - for a Moving Light (no C-Clamp) with the Parts Record

3D locus placement

A 3D locus should be placed within the Body at the attachment point for the body to the yoke. Its X and Y coordinates should be (0,0) and the Z-height should be the point that the body tilts around. Use the OIP of the 3D Locus to enter the (0,0) values to guarantee they are correct. Sometimes the rounding can make the values appear to be correct but they may be microscopically incorrect. This 3D Locus should have the Parts record attached and be assigned the Point option. If the values are not true (0,0) this can cause the light to break apart when focused.

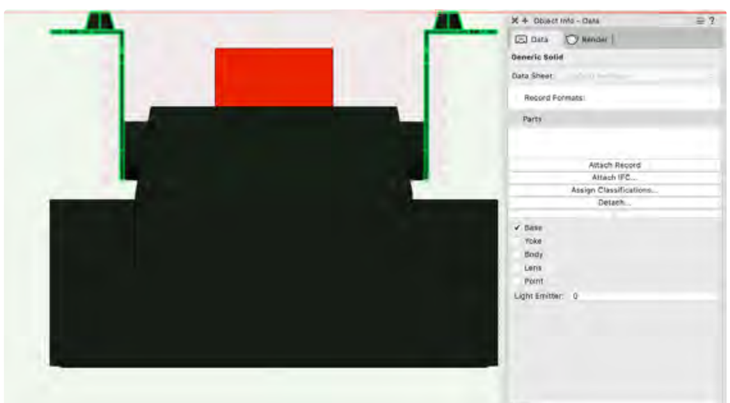


3D locus point in relation to the body and yoke.

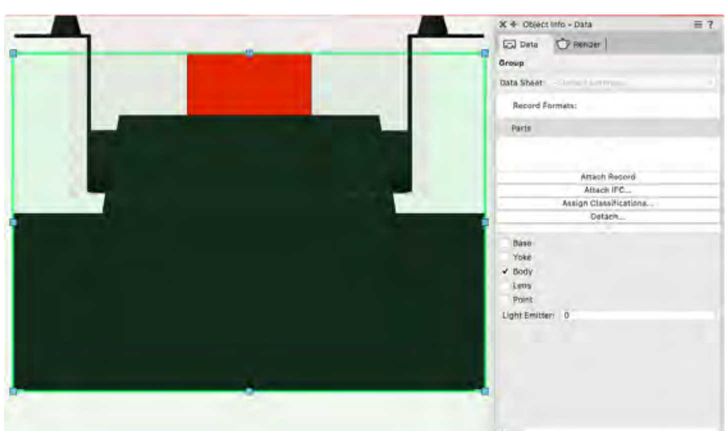
Strip lights - special setup for locus

Strip lights (with floor trunnions) are unique because they will have only a body and a base. They are designed to just tilt from a central pivot point with the floor trunnions acting as a “yoke and base combination” with tilting capability.

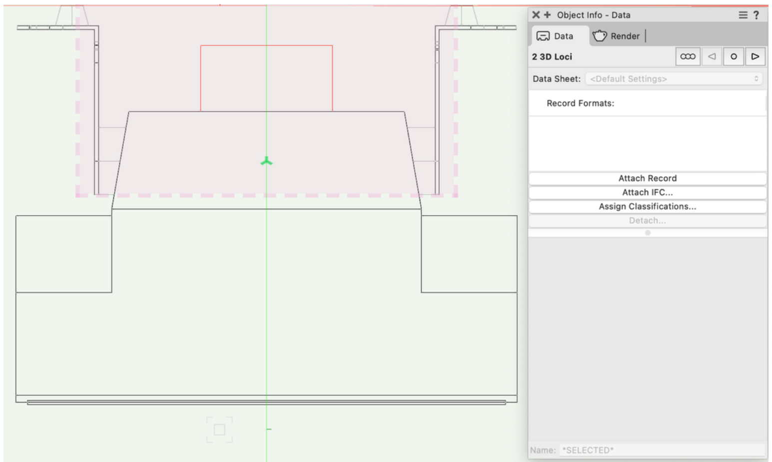
For lights like this a second 3D Locus should be placed in the exact same location as the typical 3D Locus that gets assigned to “Point” in the record. This second 3D Locus will prevent the pan controls from working when the light is used and only allows for tilting. There is an example symbol of this, *Chauvet STRIKE Bolt 1C* in the template file that demonstrates this.



Strip light/uplight with floor trunnions highlighted.



Strip light/uplight with floor trunnions highlighted.



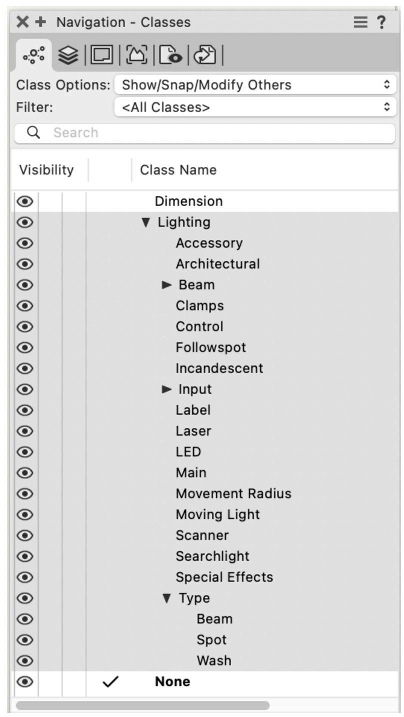
There are two 3D loci in this light to restrict the body to only tilt. Both loci should be in the same spot with only one having the parts record attached.

Classing the 3D geometry in the symbol

The 3D geometry should be classed according to the type of light that it will be. All of the classes are located in the template file under the 'Lighting' parent class (see image of the Navigation palette).

In this image are shown all the classes that can be used depending on the type of lighting device that is being made. Most of the time only one of these classes will be used on a given lighting device symbol.

Tip: If you know the class of the geometry when initially modeling, make that class active so all geometry does not need to be moved to the proper class, later.



Classes found in the *Lighting Device Template.sta* template

The most common types of lighting devices developed are shown below on this table. Consult the table for which class should be used. Each of the listed types should have all of its geometry classed one of the the listed class for a device:

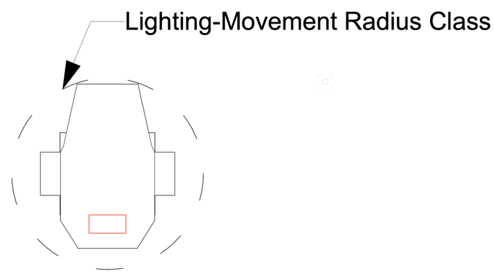
Please note the both the 2D and 3D class assignment is shown below for comparison for special case light types that require a separate class for the 2D. These 2D classes will be covered again when the 2D Component's classes are explained in the next section.

Types of lighting devices	Class(es) applied to 3D geometry
Ellipsoidal (Iekos, S4, Strand 360Q) and PARS that have incandescent lamp	Lighting-Incandescent
Ellipsoidal, PARS, uplights, and non movers that use LED engines	Lighting-LED
Moving lights (typical)	Lighting-Moving Light
	Lighting-Input-3D (input connector only)
Moving light – special type – Beam	Lighting-Moving Light (3D only) *Lighting-Type-Beam (2D part only)*
	Lighting-Input-3D (input connector only)
Moving light – special type – Spot	Lighting-Moving Light (3D only) *Lighting-Type-Beam Spot (2D part only)*
	Lighting-Input-3D (input connector only)
Moving light – special type – Wash	Lighting-Moving Light (3D only) *Lighting-Type-Wash (2D part only)*
	Lighting-Input-3D (input connector only)
Hazers, foggers, pyro, strobe lights, confetti cannons, and other effects	Lighting-Special Effects
Mirror aimed fixtures like Rosco I-Cues or club scanners	Lighting-Scanner
Outdoor sky scanner searchlights	Lighting-Searchlight
Follow Spots (manual or computerized standalone fixtures)	Lighting-Followspot
Clamps (if a clamp is in the product)	Lighting-Clamps

2D component creation

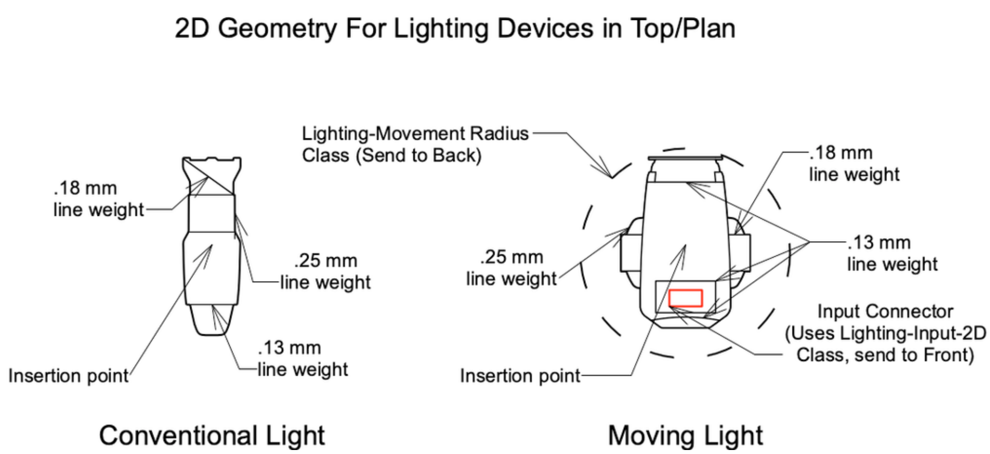
Consider and apply the following when developing the 2D component:

- Consult the Basic Guide’s section on, ‘Creating 2D geometry for 3D/Hybrid symbols’.
- The 2D plan representation of the lighting device is intended to be a means of distinguishing devices on the plot in a Top/Plan view and not a detailed plan schematic of the device. The basic shape of the device and a means of differentiating similar instrumentation is the goal.
- The 2D plan representation should also present an accurately sized representation of the device so that designers know how much space will be needed for each device. If the device is a moving light a circle should be drawn with the center at (0,0) showing the extent of the movement radius. This circle should use the ‘Lighting-Movement Radius' class included in the template file and will take the Class Attributes. Be sure to send the circle to the back using the 'Send to Back’ command so that it will not obscure any geometry if a designer uses Data-Viz or Spotlight Preferences to match the fill color to the gel color field.



Class For The Movement Radius of Moving Light
Fixtures, Make Sure to Send To Back

- The 2D component should be drawn with the front of the device, which emits light, facing the top (North) of the screen. When the body is shown in 2D will it aimed in a horizontal position looking forward from a top view. Unlike the 3D with the body is orientated aiming downward.
- Construct the 2D plan view of the symbol using as few polygons and lines as possible. Wherever possible use a single polyline and trace the body of the device. The more polylines or pieces of geometry present in the symbol the bigger the file size will be. This can cause performance issues. A good trick to help with this is to aim the 3D body and then show the 3D while in the 2D Top component. You will be able to see the 3D geometry to trace.
- You can also generate the 2D by running the ‘Generate 2D from 3D component command’ in the right click menu but be aware that this produces a group made of many segments that will need to be cleaned up and composed. The ‘Generate 2D from 3D component command’ is also explained in the Basic guide.
- The 2D Plan view should have a solid white fill so that it obscures information under the symbol. Be sure all geometry with this fill is closed as not having a closed polyline can cause issues if the designer uses Data-Vis or the Spotlight Preferences to match the fill color to the gel color field.
- The line weight of the perimeter of the 2D plan view should be .25mm. This will provide a slightly heavier line weight that will help the devices stand out when printed. This should be the heaviest line weight of the lighting devices. Major interior details should be drawn with .18mm line weight and any minor interior details should use a .13mm line weight. If the device is very small then use .18mm for the perimeter and .13mm for all interior work. Check how the symbol looks with ‘Zoom Line Thickness’ on to be sure that the 2D does not turn into an indistinguishable blob when used with smaller scales.



Input connector 2D

For the input connector in 2D (similarly to 3D), it should simply be a 2D rectangle drawn over top of the 3D connector. It should get assigned to the ‘Lighting-Input-2D’ class and will take the graphic attributes of the class. Be sure to send the 2D to the front in 2D so that it displays on top of the other geometry.

Specific 2D graphics using USITT

Some 2D will use USITT standard. There is a portion of the USITT Symbol Guideline at the end of this guide.

Graphics

The USITT Lighting Design Graphics should be used depending on the type of device that is being created. For example PARS and ERS devices typically use the USITT graphics, but other types of devices will not use USITT and are also not necessarily covered by those standards.

Text and fonts

- Typically text labeling should not be included with symbol geometry and that should be unique enough to distinguish it from the other symbols. However there can be different internal USITT graphics that are used for various beam spreads. Refer to the standard USITT lighting graphics standard at the end of this guide.
- If text is needed to satisfy the USITT lighting graphics standard (for example a “Z” to indicate a moving lens) you should place the text roughly 20% from the front of the unit. Make sure the text it is not too big that it will obscure and not too small to read. A lot of information gets added to the interior of the light when users make their label legends so you don’t want to get too close to the middle.
- Be sure to convert the text to a polyline as text in symbols sometimes does not scale properly.

⚠ It is always best practice to not introduce fonts into the file and just use Arial fonts in the content, as noted in the Basic Style Guide.

Classing the 2D geometry inside the symbol

Similar to the 3D, the 2D component needs to also be classed. In most cases the class is the same except as noted in the table below.

Types of lighting devices	Class(es) applied to geometry (2D)
(lekos, S4, Strand 360Q) and PARS that have incandescent lamp	Lighting-Incandescent
Ellipsoidal, PARS, uplights, and non movers that use LED engines	Lighting-LED
Moving lights (typical)	Lighting-Moving Light
	Lighting-Input-2D (input connector only)
Moving light – special type – Beam	Lighting-Type-Beam (2D part only)
	Lighting-Input-2D (input connector only)
Moving light – special type – Spot	Lighting-Type-Beam Spot (2D part only)
	Lighting-Input-2D (input connector only)
Moving light – special type – Wash	Lighting-Type-Wash (2D part only)
	Lighting-Input-2D (input connector only)
Hazers, foggers, pyro, strobe lights, confetti cannons, and other effects	Lighting-Special Effects
Mirror aimed fixtures like Rosco I-Cues or club scanners	Lighting-Scanner
Outdoor sky scanner searchlights	Lighting-Searchlight
Follow Spots (manual or computerized standalone fixtures)	Lighting-Followspot

Clamps (if a clamp is in the product)	Lighting-Clamps

Special 2D classing for moving lights

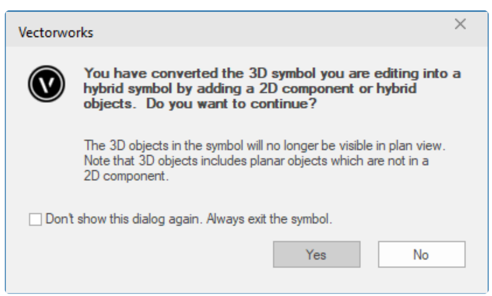
The Lighting-Type-Beam, Wash, and Spot special classes are to be used for the 2D geometry of moving light fixtures that have multiple optical path versions of the same name. You will see this the four moving light types in the table above. Below are examples of the ‘Light Sky Bumblebee F330R’ device with the standard and Beam, Spot, and Wash variants.

By assigning the special class, it will provide the graphic attributes needed differentiate these symbols that share the same overall form but with different internal components, which we do not model.

Remember that the idea of the 2D graphic is to allow the electrician to differentiate between lighting devices on the plot. Make your symbols simple and definitive.

Making it a hybrid symbol

Once all of your 2D editing work is complete in the 2D component, press the 'Exit Symbol' button. Accept and say “Yes” on dialog that appear and this will create aligned 2D and 3D geometry.



The 2D and 3D alignment is also explained in the Basic Style guide in the section:

3D/Hybrids and Symbol Options and this should be read.

Note if you make additional 2D changes be sure that it always align with the 3D. This is important so that the hanging points and geometry of both components line up.

Symbol (container) records

Once the hybrid lighting device symbol has been created, there are two records that need to be attached to the overall symbol and populated with data. They are the ‘Light Info Record’ which is used to make the geometry function as a Spotlight Lighting Device and the 'EntEquipUniversal Record' which is used for inventory and equipment tracking purposes.

To attach a record, right click on the symbol in the Resource Manager and choose ‘Attach Record’. Locate the desired records in the Records folder in the template file and select the check marks next to the records. Click OK to close the dialog and both records will be attached to the symbol. You can attach one or multiple records at once using this command.

The 'Light Info Record’

This should be filled out as completely and accurately as possible with manufacturer's supplied information that is noted on websites and in catalogs. Make sure to put the appropriate units on all the fields and to specify kg or lb in the weight field as some manufacturers only list one. The format and necessary units are shown below.

Field	Value Criteria
Model Name	Manufacturer-Model Name
Inst Type	Manufacturer-Model Name
FixtureID	Manufacturer-Model Name
DeviceBody	Only used if there are interchangeable parts
DeviceLens	Only used if there are interchangeable parts
Device Type	Choose from the following: Light, Moving Light, Accessory, Static Accessory, Device, Practical, SFX, Power, Other
GDTF Type ID	If listed on the GDTF website (http://gdtf-share.com) paste the information into the field
GDTF Fixture Name	If listed on the GDTF website paste the information into the field
GDTF Fixture Mode	By default this should be, 1 - If the GDTF Type ID field of the 'Light Info Record' is left blank, then this field should be left at the default value of 1. If you have located and entered a GDTF Type ID, then replace the 1 in GDTF Fixture Mode field with a value of 0.
Power Connector	Will default to NEMA 5-15 (User will edit)
Signal Connector	Will default to 5-Pin DMX (User will edit)
Wattage	# W (Find this on spec sheet, must be number followed by space and capital W)
Voltage	Leave Blank (User will edit)
Source Wattage	Will default to 0 (User will edit)
Lamp	Lamp code if not LED, otherwise will be LED
LED Color Temperature	Defaults to 5000 K (User can edit)
Frame Size Imp	Frame size in inches followed by “in” as units are needed, if no frame then leave blank

Frame Size Metric	Frame size in millimeters followed by “mm” as units are needed, if no frame then leave blank
Weight	Can be pounds of kilograms, just have appropriate units (kg or lb)
Beam Angle	Vertical minimum of beam spread, just a number (See chart below)
Field Angle	Vertical maximum of beam spread, just a number (See chart below)
Beam Angle 2	Horizontal minimum of beam spread, number (See chart below)
Field Angle 2	Horizontal maximum of beam spread, number (See chart below)
Candlepower	Brightness of light that is emitted (candle power). Usually in the spec sheet as lux so apply the following formula: Candle power: Distance^2 * Lumens
Num Channels	Maximum amount of DMX channels that the fixture can use depending on mode
Remarks	Enter any other relevant information here, if not leave blank. Mostly will be left blank.
Drawn By	Initials of creator, used to confirm the origin of the fixture if there is a bug filed later by a user.

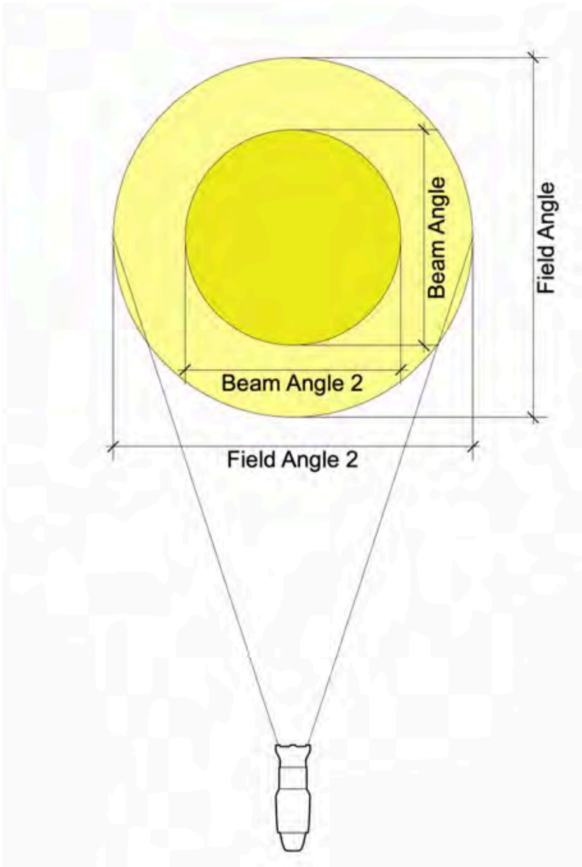


Diagram Showing Beam and Field Angle 1 vs. 2 and Direction

The 'EntEquipUniversal' record

This is used for inventory and equipment tracking purposes. It allows users to keep track of what equipment has been used vs. what is in their inventory. The record is applied to every Spotlight symbol and the fields are populated. As the record is used by all Spotlight symbols the device type varies. For Spotlight Lighting Devices the device type should be ‘Lighting Device’ as shown below.

Field	Value Criteria
Short Name	Model Name
Symbol Name	Manufacturer-Model Name
Location	Leave Blank (User will edit)
Position	Leave Blank (User will edit)
Unit Number	Leave Blank (User will edit)
Notes	Leave Blank (User will edit)
Device Type	Enter ‘Lighting Device’
MRV-ID	Leave Blank (User will edit)

Completing the library/project

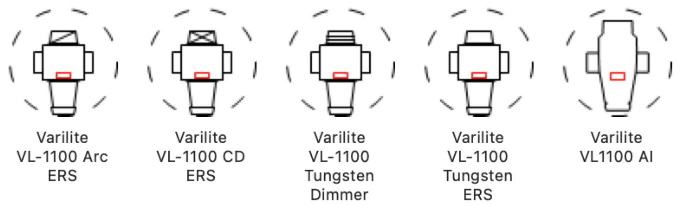
The Finishing the Library section of, the Basic Style Guide, should be referred to in conjunction with the following:

Resource names

The symbol names should use the following structure for their nomenclature. Varilite is used here as an example but each manufacturer will be unique.

Brand	Product line or Product Number
Varilite	VL-1100 Arc ERS

Symbol name example: ***Varilite VL-1100 Arc ERS***



Symbol Names in Resource Manager

Resources arrangement in the Resource Manager (RM)

The main subjects of the library file should be on the top level. This will typically be the products in a library of products from a manufacturer.

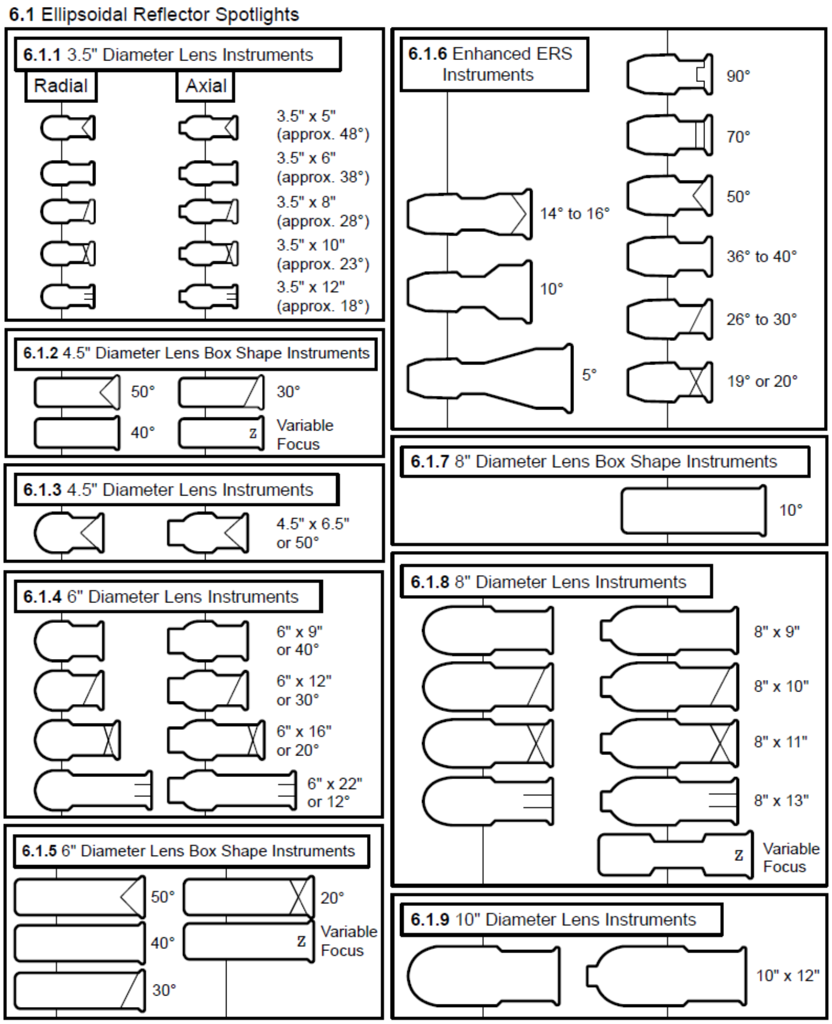
Supporting resources that are nested or optional should be put into appropriate folders (e.g. hatch resources go into a ‘Hatches’ folder).

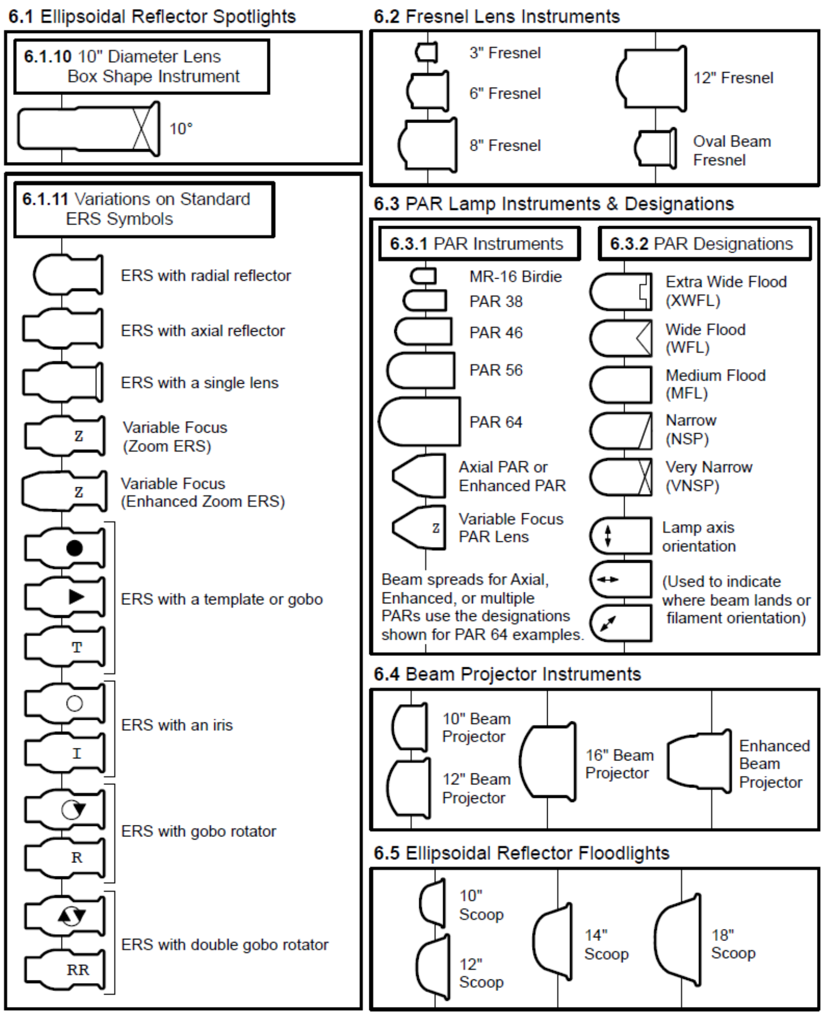
Folders can also be used to hold the main subject content if division is needed for product lines, categories, etcetera in a file.

This completes the Lighting Device Guide!

USITT Lighting Design Graphics

We are providing a portion of their standards for reference. These graphics come courtesy of USITT.





GDTF creation videos

For some additional bonus instruction, below is a series of instructional videos on how to create the GDTF light device (fixture), which is not part of this guide.

How to Create a GDTF Lighting Fixture: Part I

Created by GDTF & MVR • Updated on May 3, 2024

This is the first video of a six-part series. In this video, we'll take a look at the initial steps involved in the process of creating a GDTF lighting fixture. You'll discover the resources you will need, whether to use a template, and how to fill in the device description of The Builder.

YouTube

How to Create a GDTF Lighting Fixture: Part II

Created by GDTF & MVR • Updated on May 3, 2024

We recommend watching "How to Create a GDTF Lighting Fixture: Part I" before watching this video. In part II, you will learn how to set up the geometries of a light and how they link to each other, and the physical attributes of a light.

YouTube

How to Create a GDTF Lighting Fixture: Part III

Created by GDTF & MVR • Updated on May 3, 2024

We recommend watching Part I and Part II of this series before watching this video. In Part III, we will define the spectral color behavior of the light and any of the color filters it has.

YouTube

How to Create a GDTF Lighting Fixture: Part IV

Created by GDTF & MVR • Updated on May 3, 2024

We recommend watching Part I, II and III of this series before viewing this video. By part IV in this series, we know that most lights now include some form of color or effects wheel and in many cases, both. Let's take a look at how these wheels are set up and defined in a GDTF.

YouTube

How to Create a GDTF Lighting Fixture: Part V

Created by GDTF & MVR • Updated on May 22, 2024

We recommend watching part I - IV of this series before watching this video. By now, your fixture has the basic physical, photometric, and spectral aspects of the light defined. This video will dive into how the lights are controlled.

YouTube

How to Create a GDTF Lighting Fixture: Part VI

Created by GDTF & MVR • Updated on Jul 23, 2024

We recommend watching videos I - V in this series before watching this video. By now, your fixture is almost finished! This chapter covers the final steps of creating your GDTF - setting up macros, the fixture summary, and testing your file.

YouTube