

AOVs and Displays in Katana

- [EXR Metadata](#)
- [Setting Custom Display Data](#)

Follow these sets to set up AOVs in RenderMan for Katana:

- Create a `PrmanOutputChannelDefine` node for each AOV channel. This sets up the channel name for `RiDisplayChannel`.

? Unknown Attachment

- [LPE](#) will require a source to define the light path to collect. For example, Indirect Diffuse would be

```
color lpe:C<RD>.+[<L.>O]
```

- Next, Create a `RenderOutputDefine` (Display Driver) node for each AOV file. This sets up the output file name and channel name be used for `RiDisplay`.

? Unknown Attachment

- Wire the `PrmanOutputChannelDefine` node into `RenderOutputDefine` to see the added channel in the `RenderOutputDefine`'s channel drop down (`MatteID0` in the example above).

? Unknown Attachment



To correctly render per-lobe [LPE](#) in Katana, you need to use an [OpScript](#) to declare how these are routed to the outputs. Below is the full OpScript to use for the lobe names defined by `PxrSurface` and `PxrMarschnerHair`. If your studio has its own bxdfs with different names, you will need to add those names to these Options.

```
Interface.SetAttr('prmanGlobalStatements.options.lpe.diffuse2', StringAttribute("Diffuse,HairDiffuse,
diffuse,translucent,hair4,irradiance"))
Interface.SetAttr('prmanGlobalStatements.options.lpe.diffuse3', StringAttribute("Subsurface,
subsurface"))
Interface.SetAttr('prmanGlobalStatements.options.lpe.specular2', StringAttribute("Specular,
HairSpecularR,specular,hair1"))
Interface.SetAttr('prmanGlobalStatements.options.lpe.specular3', StringAttribute("RoughSpecular,
HairSpecularTRT,hair3"))
Interface.SetAttr('prmanGlobalStatements.options.lpe.specular4', StringAttribute("Clearcoat"))
Interface.SetAttr('prmanGlobalStatements.options.lpe.specular5', StringAttribute("Iridescence"))
Interface.SetAttr('prmanGlobalStatements.options.lpe.specular6', StringAttribute("Fuzz,
HairSpecularGLINTS"))
Interface.SetAttr('prmanGlobalStatements.options.lpe.specular7', StringAttribute("SingleScatter,
HairSpecularTT,hair2"))
Interface.SetAttr('prmanGlobalStatements.options.lpe.specular8', StringAttribute("Glass,specular"))
Interface.SetAttr('prmanGlobalStatements.options.lpe.user2', StringAttribute("Albedo,DiffuseAlbedo,
SubsurfaceAlbedo,HairAlbedo"))
Interface.SetAttr('prmanGlobalStatements.options.lpe.user3', StringAttribute("Position"))
Interface.SetAttr('prmanGlobalStatements.options.lpe.user4', StringAttribute("UserColor"))
Interface.SetAttr('prmanGlobalStatements.options.lpe.user6', StringAttribute("Normal,DiffuseNormal,
HairTangent,SubsurfaceNormal,SpecularNormal,RoughSpecularNormal,SingleScatterNormal,FuzzNormal,
IridescenceNormal,GlassNormal"))
```

- To make this AOV render interactively you can add it to the `interactiveOutputs` selection in the `RenderSettings` node.

? Unknown Attachment

- **Advanced:** You can also add a statistics selection to the node. Typically used for diagnosis and tuning:

? Unknown Attachment

- **"variance"** – estimates the statistical variance of values contributing to the pixel in associated source channel.
- **"mse"** – like variance, but diminishes towards zero as the number of samples increases. Though somewhat noisy itself, this can provide an estimate for the amount of mean-squared-error versus a hypothetical ground-truth image.
- **"even"** – an image produced using only one half of the camera samples
- **"odd"** – an image produced using just the other half of the camera samples

EXR Metadata

You can add metadata to the EXR file to facilitate pipeline functions. The Foundry documents this [here](#).

Essentially, on the Render Settings node:

```
SetAttr( "renderSettings.ouputs.primary.renderSettings.exrheaders.test_string",[ "Your string" ] )
```

Setting Custom Display Data

You can pass a custom Katana attribute into the display driver using an OpScript at the /root. Below is a simple example setting "myParam" onto a display called "myDisplay"

```
Interface.SetAttr( "renderSettings.outputs.myDisplay.renderSettings.displayOptions.myParam", StringAttribute  
("paramValue" ) )
```