

RenderMan for Houdini 23.0

Welcome to RenderMan for Houdini 23!

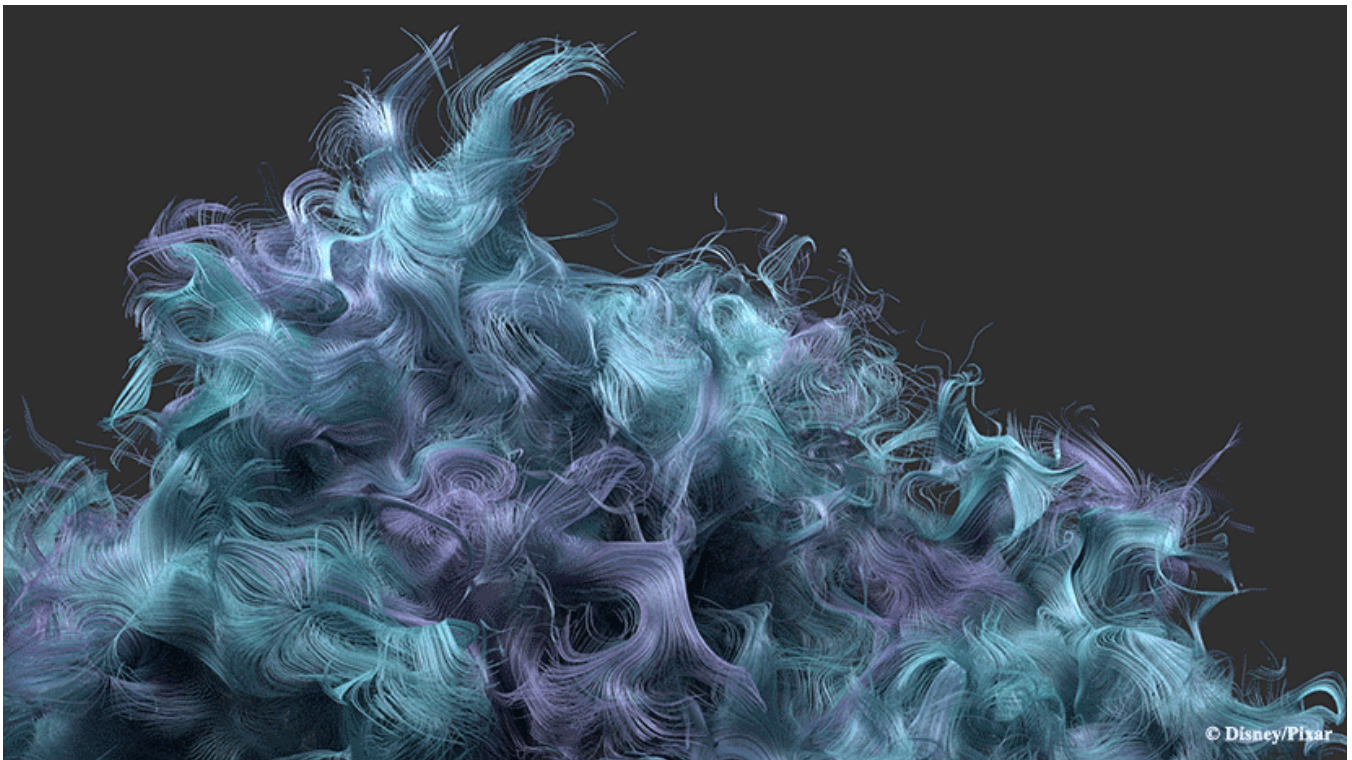
This new release of RenderMan for Houdini 23 (RfH), includes a number of new features and bug fixes that address feedback.



RfH 23 is only available for Houdini versions 17.0 and higher. We no longer support Houdini 16.5.



- Rendering with RenderMan will not operate on Houdini Apprentice editions, see Side FX's website for options that allow Third Party Rendering
- Houdini Indie is only compatible with 17.5, you must be using at least version 17.5.210 or higher



"The Unsullied" by Dylan Sisson, rendered in RenderMan for Houdini

What's New

Solaris support in Houdini 18.0

For those using the 18.0 of Houdini, the key feature is the new support for the RenderMan delegate in Solaris. In addition to RenderMan Hydra Delegate support with v23, RenderMan for Houdini also offers additional interactive display options, batch rendering options, RenderMan specific geometry settings, and additional advanced settings for light sources when using Solaris.

INTERACTIVE OPTION EDITS

You can render through even more stages of your pipeline. RenderMan options on the ROP that previously required a render restart to take affect are now interactive. These options include resolution, pixel variance, screen window, min and max samples, and new interactive refinement. In addition you can add, remove, and edit AOVs live.

PDG SUPPORT

Schedule rendering tasks with the new RenderMan TOP nodes - Texture Maker (txmake), Generate RIB (generaterib), Render RIB (renderrib), ROP RenderMan Render (roprib), RenderMan Denoise (risdenoise) - or render with a ROP Fetch node.

IMPROVED HDA CONVERSION

Set environment variable RFH_ARGS2HDA=1 to create houdini digital assets on start up based on .args or .oso files found in RMAN_SHADERPATH and /or RMAN_RIXPLUGINPATH. Don't want to convert shaders every houdini startup? You can also use the args2hda.py script elsewhere in your pipeline. We have also improved the args2hda.py script to handle more complex shader metadata.

AND MANY OTHER CHANGES & BUG FIXES...

- Created dynamic array node that allows integer inputs. This node replaces the buildarray node that caused warnings when connected to utilityPattern of PxrSurface
- The Houdini node path is now used as the RenderMan object identifier for rib debugging
- No longer propagate non-fatal rendering errors to the ROP node. RenderMan errors are now considered warnings on the ROP node
- Initial rendering now obey crop parameters
- Corrected NURBS surface orientation
- Added MatteID users parameters to RenderMan spare parameter tab for easier access
- Null node can now represent a coordsys with it's node path
- Created sun direction handle for PxrEnvDayLight
- Mesh light geometry no longer needs to be included in candidate lights to illuminate scene
- A bug that prevented an auto headlight from being generated when using render region has been fixed.
- Added support vector3, vector4, matrix4x4 parameter script_ritype tags
- Deprecated PxrManifold3DN
- Forced matte objects are now rendered regardless of display toggle or forced object membership
- Fixed a bug on linux where Houdini could crash if rendering to 'it' after having rendered in the Render View
- Added "View Selected RIB" menu tool (only works in versions 17.5.229 and later)
- Simplified displacement workflow by always setting displacement bound on geometry
- Fixed extraneous warnings created when rendering motion blur on alembic geometry imported
- A bug that prevented the use of DWAa and DWAb EXR compressions has been addressed

Known Issues

- Display edits cannot be changed from the image tool to mplay
- Cross-frame denoising pdg support.
- Phantom objects are removed from all rays, not just camera rays
- Applying undo of deleted nodes or collapsing to subnet may not work as expected
- Rewiring subnet indirect inputs

Known Limitations

- Rendering with Mantra lights or VOPs not possible
- Bypassing shader nodes
- Auto camera creation for IPR
- RenderMan clipping planes