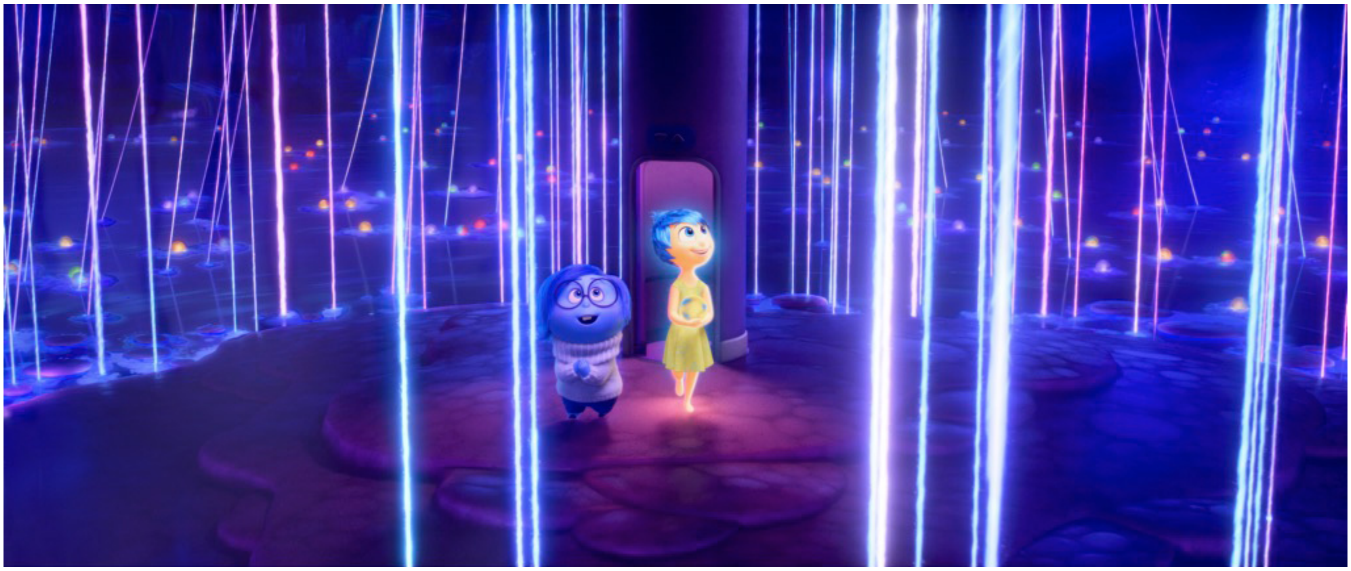
RenderMan 26.0



Inside Out 2 © Disney/Pixar

April 8, 2024

Welcome to RenderMan 26.0

This release features major updates to interactivity and scalability, thanks to significant advancements to Pixar's state-of-the-art renderer, RenderMan XPU. RenderMan 26 RIS also features an interactive version of the advanced Denoiser from Disney Research which uses machine learning to significantly accelerate the rendering process.

Improvements to Stylized Looks, significant instancing speedups, an improved statistics system, as well as support for the latest 3D applications, round off this new release.

Here are some of the highlights:

XPU

Pixar's GPU + CPU hybrid renderer receives major updates. Highlights include improvements in sampling, an expanded lighting and camera toolset, as well as light selection, significantly speeding up rendering in a wider range of production use cases.

- **Analytic Lights** — RenderMan XPU now supports most lighting features, including IES profiles and light temperature, giving artists a key and shot lighting toolset of production complexity. Mesh lights will be coming in a future release.
- **Light Linking** — A lighter's bread and butter! Light and shadow linking is now fully supported in RenderMan XPU, allowing artists to isolate lighting to particular geometry sets for maximum creative control.
- **Light Filters** — XPU now supports all light filters, including gobos and cookies, maximizing artist control for shot lighting. You can light link them too!
- **Camera Controls** — XPU has greatly expanded support for Pixar Camera, including tilt shift, lens aberrations, vignetting, split diopter, shutter controls, and many other features which have made Pixar's camera system a comprehensive cinematography tool.
- **Interactivity** — Progressive Pixels, a feature which displays fractional iterations for improved interactivity in XPU, can now be dialed in, allowing artists to find the sweet spot between speed and fidelity for their given project, be it a single asset, tens of volumes, or a citiscape.
- **Adaptive Sampling** — XPU can now render images to an acceptable variance metric instead of fully converging to a given sample count, greatly speeding up render times.
- **Light Selection** — Scenes with hundreds of lights can now be efficiently rendered in XPU, significantly expanding interactivity for layout, as well as key and shot lighting.

For further details about RenderMan XPU™, see the following documentation sections:

- XPU Guide - Overview of XPU in RenderMan.
- Features & Limitations - Understand the differences between XPU and RIS and known limitations.

RIS

- **Interactive Denoising** — Now available interactively in Katana and Blender when using RIS, the RenderMan Denoiser is a completely new state-of-the-art denoising technology developed by Disney Research, which takes a new approach to denoising, using machine learning and training data from Disney, ILM, and Pixar. The interactive Denoiser can significantly reduce artist iteration times for both feature animation and VFX, by producing results predictive of the offline denoiser during an interactive rendering session.
- **Stylized Looks** — Continued improvements to the Stylized Looks toolset give artists new ways to express line work, hatching, and toon effects, as well as a new Canvas layer, which simplifies an important step in the process.

Core Enhancements

RenderMan XPU and RIS receive a series of core updates which dramatically speed up time to first pixel and advance support for USD pipelines.

- **Faster Instancing** — RIS and XPU have greatly improved performance when creating and editing scenes with many instances.
- **Faster Textures** — Texture read parallelism has been dramatically improved, resulting in significantly faster texture lookups in all scenarios, especially when using EXR textures. This greatly improves time to first pixel.
- **Improved Statistics** — Advancements to telemetry and reporting, as well as an improved interactive stats portal.

Stylized Looks

RenderMan's Stylized Looks toolset in version 26 offers more creative control for artists to create non-photorealistic images. Highlights include better smoothing of stylized results, new controls for color remapping, as well as expanded compositing modes and detection methods, offering artists new ways to stylize their Toon, Hatching, and Lines. The user experience has also received an update to be more intuitive, and artist friendly.

Highlights:

- **Lines** — RenderMan Stylized Looks now features easier line detection, remapping controls, and added filtering controls, for smoother looking lines.
- **Toon** — The Stylized Control toolset now includes an artistic style toon mode, which is not physically based, expanding the range of stylization achievable.
- **UX Improvements** — The user experience for Stylized Looks continues to improve for better artist workflows, including a reorganization of attributes, a new Canvas layer, better AOV organization, as well as better filtering and smoothing effects.

Artist Tools

Many updates have been implemented to plugin usability and stability. Ongoing collaboration with the USD and Hydra teams continue to accelerate RenderMan's support for the industry standard format. The enhancements provide a more seamless workflow for artists using RenderMan in their production pipeline.

- **Updated Bridge Tools** — The latest 3D apps are now supported by RenderMan: Houdini, Solaris, Katana, Maya, and Blender.
- **New Stats Portal Tool** — There is a new stand-alone Stats Portal application for interfacing with RenderMan's new stats system.

Additional Features

RenderMan also includes:

- **VFX Reference Platform 2023** — All plug-ins are now updated to conform to the standard.
- **CUDA for XPU** – Updated to Version 12
- **Enterprise Linux 7** – For those of you using Linux, RenderMan provides builds for Enterprise Linux 7. Enterprise Linux 9 support will arrive in a future release.

Application Compatibility and Requirements

RenderMan RIS requires CPUs capable of running the AVX instruction set or better. A minimum of 8GB of RAM is recommended for RIS.

RenderMan XPU requires CPUs capable of running the AVX instruction set. XPU is only supported on Linux and Windows systems. GPU acceleration is supported on NVIDIA graphics cards from the Quadro, Tesla or Data Center GPU ranges, with the Pascal architecture or later. A minimum of 11 GB of VRAM is recommended and 24 GB of VRAM is suggested for best performance. A minimum of 16GB of system RAM is recommended for hybrid (CPU + GPU) processing. For more information on driver requirements, please consult the XPU Technical Specifications.

For utilizing the NVIDIA Optix AI Denoiser in "it", Optix 7 supported hardware, NVIDIA "Pascal" architectures and newer is required.

RenderMan is compatible with the following 64-bit operating systems:

- Linux CentOS/RHEL 7.2 - 7.9 (EL9 support will be coming in a point release)
- Windows 10 and 11
- macOS 10.15 through 13.X. Apple Silicon is only supported with Rosetta 2. Intel version of DCCs required.

RenderMan is also compatible with the following DCCs:

- Houdini 19.0, 19.5, and 20 (production versions only). Solaris support for 19.5 and 20 (production versions only)
- Katana 5.0, 6.0, 6.5 (7.0 support will come in a later release)
- Maya 2022, 2023, and 2024 (2024 is Windows and macOS only, support for Linux will come in a later release)
- Blender 3.0+ and 4.1+

RenderMan XPU is compatible with the following 64-bit operating systems:

- Linux CentOS/RHEL 7.2 - 7.9 (EL9 support will be coming in a point release)
- Windows 10 and 11

RenderMan XPU is compatible with the following for its GPU accelerated mode:

- NVIDIA “Pascal” architectures and above.
- Driver requirement for Linux is 525.60.13 or above. For Windows, 527.41 or above.

Deprecations

- Python 2 support is being deprecated. This is the last release that supports Python 2.
- Support for Pixar's deep texture format is being phased out in favor of DeepEXR. We anticipate pulling libdtex and the DTex format in the next release.

More Details

XPU

General

- Adaptive sampling has been implemented in XPU. The workflow and controls are the same as in RIS: PixelVariance, "darkfalloff", "exposurebracket", "adaptall", and per-channel "relativepixelvariance". There are two "adaptivemetric"s: "contrast" and "variance", with the latter being the default. RMAN-20888
- An issue where using Render region in Solaris in XPU mode could lead to a crash has been addressed. RMAN-20929
- XPU will fall back to CPU only mode if the variant is "xpu" and no capable GPU hardware is detected. RMAN-21526
- Multiple crash issues surrounding switching between RIS and XPU in Solaris have been addressed.
- The amount of memory required to compile PTX shaders has been reduced, lowering GPU VRAM requirements significantly in some cases.
- Eliminated "IMAGE_REL_AMD64_ADDR32NB relocation requires an ordered section layout." errors that could lead to crashes. RMAN-21782

Ray tracing

- XPU now has full support for trace groups and membership. This includes:
  - supporting Attribute "trace" "reflectsubset", "reflectexcludesubset", "transmitsubset", "transmitexcludesubset", "shadowsubset", and "shadowexcludesubset";
  - supporting the "shadowSubset" and "shadowExcludeSubset" parameters in light sources, which implement light-shadow linking;
  - supporting Attribute "lighting" "subset" and "excludesubset", which implement light linking;
  - supporting Attribute "lightfilter" "subset", which implement light filter linking;
  - support for the "subsurfaceSubset" and "singlescatterSubset" parameters in PxrSurface.

Image Output and Displays

- Fixed crashes in XPU that could occur on scenes with high numbers of complicated LPE expressions. RMAN-19362
- XPU now correctly respects /prman/lpe/{diffuse,specular,user}* settings defined in rendermn.ini. RMAN-20605
- XPU now supports the "sum" pixel filter type. RMAN-21472

Geometry

- The time to process large on screen subdivision meshes has been greatly improved, resulting in a significant improvement in time to first pixel for e.g. scenes with large ground planes or large characters. RMAN-19710
- Optimizations have been made to preprocessing of dense and complicated geometry which also result in significant improvement in time to first pixel for complex scenes. RMAN-21184
- A crash issue with subdivision mesh primvars of type float[X] where X is not 1 or 3 has been fixed. RMAN-21851
- XPU now supports Attribute "dice" "strategy". RMAN-20687
- XPU now supports Attribute "dice" "string referenceinstance". RMAN-20748
- XPU now supports the sphericalprojection dicing strategy. RMAN-20747
- XPU no longer forces a lower bound of 0.1 on the micropolygon length. RMAN-21423
- An issue with support for the near clipping plane in XPU has been fixed. RMAN-20928
- XPU now supports RiClippingPlanes (and the associated Riley entry points). RMAN-20033
- Fixed a crash when rendering motion blurred curves in XPU using RfSolaris or Hydra.
- Several issues surrounding bias calculations for round curves in XPU which could lead to false self-intersections have been addressed. RMAN-21837
- A crash on polymeshes with polygon:smoothdisplacement has been fixed. RMAN-20938
- Fix an issue in XPU that could lead to sporadic crashing on dense polygon meshes. RMAN-20993

Volumes

- XPU edits on aggregate volumes materials now correctly update the render in all cases. RMAN-20516
- Fixed a bug with volumes with colored extinction in XPU GPU mode. RMAN-21112
- Problems when rendering volumes in conjunction with visibility flags (camera, indirect, and transmission) have been addressed. RMAN-21105
- Fixed an issue with volumes which could lead to bucket artifacts on the CPU when rendering with camera visible lights. RMAN-20670
- Fixed an issue with dsominmax 0 volumes which most noticeably caused problems when rendered with camera visible lighting.
- XPU is now more robust against misdeclarations of the data type of VDB grids used for aggregate volumes. RMAN-20987
- A problem that could cause aggregate volumes to disappear on the GPU due to infinite bounding boxes has been addressed. RMAN-20970

Lighting

- Portal lights are now supported in XPU. RMAN-18217
- Light filters are now supported in XPU. RMAN-18747
- Fixed a bug where rect light textures were read incorrectly. RMAN-20906
- Fixed an issue in XPU where light texture maps required absolute paths. RMAN-21184
- Fixed XPU lighting bug where light map saturation/gamma controls did not get updated in an interactive render session. RMAN-21635
- Shadow falloff has now been implemented. RMAN-19949
- Fix bug where XPU & RIS images did not match involving IES profiles. RMAN-21744
- Indirect LPEs in XPU now match RIS better for the case of surfaces with multiple lobes and high specular roughness. RMAN-19002

Integrators

- XPU can now switch integrators during an interactive render. The Riley entry points CreateIntegrator, ModifyIntegrator and DeleteIntegrator are now supported. RMAN-18198
- In XPU, the numLightSamples parameter to PxrPathTracer is now supported. RMAN-21401
- Negative Bxdf emission colors are now allowed in XPU (they are no longer clamped to zero).

Shading

- XPU's default verbosity for OSL is now NORMAL instead of VERBOSE. XPU now supports Option "osl" "int verbose" to change this verbosity.
- Fixed issues which could cause crashing when interactively editing OSL pattern nodes in a DCC. RMAN-21227
- An inconsistency between the getattribute() OSL function between XPU and RIS when dealing with array lengths has been fixed. RMAN-21935
- XPU now has partial support for the utilityPattern parameter in PxrSurface and other shaders. Currently, only a single utilityPattern is currently supported. RMAN-21283
- The view vector Vn is no longer incorrectly normalized in XPU. RMAN-21399
- Implemented "followTopology" and "continuationRayMode" for XPU PxrSurface Burley diffusion ("subsurfaceType" 4). RMAN-21774
- Implemented OSL getattribute("context", "reyesGrid", ...) lookup in XPU.
- DeleteDisplacement and DeleteMaterial (from the Riley API) are now implemented in XPU.
- The PxrSurface "diffuseExponent" parameter is now implemented in XPU. RMAN-21701
- The PxrSurface "glassBumpNormal" parameter is now implemented in XPU. RMAN-21790
- The PxrSurface "subsurfaceDoubleSided" parameter is now implemented in XPU. RMAN-21773
- The PxrSurface "subsurfaceTransmitGain" parameter is now implemented in XPU.
- XPU user attribute lookups in OSL now work correctly with and without the "user:" prefix. RMAN-21003
- In XPU, the OSL calculatenormal() shadeop now computes the correctly oriented result when the geometry has an inverse scale. Similar issues with the backfacing() shadeop have also been fixed. RMAN-20795
- Enabling subsurface in PxrSurface no longer disrupts user normal lobe output. RMAN-19037
- An issue where modifying a geometry instance via the Riley interface ModifyGeometryInstance did not always update the material binding has been fixed. RMAN-21856

Texturing

- Fixed XPU texture filtering for displacement to better match RIS. RMAN-20904
- The XPU memory requirements of PTex has been greatly reduced, by a factor of almost 4X on complex shots. RMAN-18833
- Some issues in XPU's substitution of <u> and <v> tokens in texture file map names have been addressed. RMAN-19353
- A bug in XPU causing the texture filter widths on non-quad subdivision faces to be off by a factor of two has been fixed. RMAN-21751
- Fixed a bug affecting the invalidation of non .tex textures in XPU where the updated texture would not be rendered. Instead the old texture before the invalidation would be used. RMAN-17686

Statistics

- XPU stats are now included in the Image Tool (‘it’) HUD.
- Fixed the incorrect live reporting of XPU maxSamples, timeToFirst* and other high-level metrics.
- Graphs of XPU timer breakdown and ray count breakdown are now available in the new stand-alone stportal app.

RIS and XPU

General

- Fixed a bug which would cause the renderer to crash if very long (longer than 8MB) string params were passed to it. PRMAN-2789

Deep Output

- dsview, dtexmerge (now renamed deepmerge), sho, and other auxiliary programs now support DeepEXR files natively. dsview has also received several improvements. RMAN-21298

Image Output and Displays

- Added new LPE flags: 'worldtransform;' / 'objecttransform;' / 'cameratransform;' which will transform the output to the desired space. The transform will operate as if the output is a position unless a 'normal;' or 'vector;' flag is also given. The 'normal;' flag will also normalize the output which helps with certain cases in Lama where layering normals can give unhelpful results. RMAN-20972
- decidither level controls for XPU Progressive Pixels mode are now supported.
- RenderMan now includes an ACES-1.3 OCIO config file. Note, this config will only work with OCIO 2.0 and above. RMAN-19996

Geometry

- Improved time to first pixel for scenes with many instances by around 5x to 7x or more in both RIS and XPU. In addition, interactive editing and deleting operations on these instances have also been sped up substantially, in some situations by an order of magnitude or more. RMAN-19096
- A precision issue that could lead to self intersection on flat polygons with user normals in the opposite direction of the renderer computed normals has been fixed (both RIS and XPU). RMAN-20963

Shading

- Due to an improved string hashing function, the results from the OSL hash() shadeop when applied to strings have changed. Any shaders which depend on this function (including PxrRamp, when using Random Source = "Object Name") may compute significantly different results.
- Fixed a crash when a varying knot-count was used with the OSL spline() shadeop. PRMAN-2868
- OSL `gettextureinfo()` can now query metadata in textures. This is limited to only OpenEXR format textures, and will fail if it can't be done at shader JIT-compile time and constant-folded, which means it's largly incompatible with file-name substitution tokens.
- PxrSurface subsurface "continuationRayMode" 2 (and 1) for "subsurfaceType" 0, 2, 3, and 4 now receives subsurface scattering from the last hit which was previously missing under some circumstances. And the subsurface scattering noise may be reduced.
- The LPE for **direct** subsurface scattering in RIS has changed from two bounces `C<TD3><TD2>L` to a single bounce `C<TD3>L`. However, the commonly used LPE for subsurface scattering -- `CD3[DS]*L` -- (which includes optional secondary bounces) works as before.
- Reduce fireflies in diffusion subsurface scattering near 90 degree angles -- as on cubes etc.
- UDIM expansion inside <primstr:> texture substitutions are now supported. RMAN-18303
- The PxrRamp pattern now has a more intuitive 0-1 default. RMAN-21815
- Fix PxrColorCorrect Contrast Banding. This will cause a look difference between RenderMan 25 and 26. RMAN-21770
- When LamaHairChiang's "mollification" parameter is on, the three longitudinal roughness values ("ltRoughnessR/TT/TRT") were mollified too much, leading to hair that looked too diffuse. This has been fixed in both RIS and XPU. PRMAN-2805
- LamaIridescence albedo and normal AOVs now have prefix "Iridescence" -- to be more consistent with the AOVs of other Lama nodes and the PxrSurface iridescence lobe. RMAN-20871
- Lama specular albedo and normal AOVs now have prefix "Specular" -- to be more consistent with the "DiffuseAlbedo" and "DiffuseNormal" names and match PxrSurface AOV names. RMAN-20871

Statistics

- Added a new Option “statistics” “string jsonFilename” for generating a JSON report from the new stats system.
- Telemetry Listener console output now include elapsed time and metric type (SAMPLE or EVENT).
- The CSV output from the Telemetry Listener is now sorted by time.
- stportal now supports loading JSON files from the command-line.
- JSON report viewing in stportal now includes search and filter options in the file comparison tab.
- Fixed a timing bug which caused the renderer to produce inaccurately high timing stats on the Linux platform when running on some recent CPUs.

Texturing

- The OSL shaders PxrTexture, PxrMultiTexture, PxrLayeredTexture, PxrBump, and PxrNormalMap have a new int parameter: "smoothRayDerivs". When 1 (which is the default value), derivatives and hence texture filter widths at ray hit points are computed in a more robust manner, providing consistent widths independent of the orientation of the underlying surface geometry. When 0, the old formula (using straight Dx() and Dy() OSL derivatives) is used for the texture filter widths.
- The PxrTexture parameter "maxResolution" now works more reliably, both in RIS and XPU, and for both ray hit shading and displacement shading. (Same for PxrMultiTexture, PxrLayeredTexture, PxrNormalMap, PxrBump, and PxrProjectionLayer.)

RIS

Geometry

- Fixed bug when accessing P during cached opacity shading on a pretessellated subdiv with additional user vertex primitive variables. PRMAN-2882
- Fixed a bug where bad geometry could unpredictably increase the scene bounds. RMAN-21562
- A bug in the shadow and reflection trace bias for round cubic curves in RIS has been fixed. The bias is now consistent with flat and linear curves, and also consistent with XPU. The fix causes changes in shadows and shading on round cubic curves, particularly near other objects. To get the old behavior: Option "trace" "int incorrectCurveBias" [0|1]. The default is 0.
- Added support for a new spline-based opacity mode to RiPoints. Previously, falloff could be specified for RiPoints as a single power value. A primvar 'falloffpower' controlled this. 0 meant no falloff, any other value meant apply a power function using that power value to control the shape. RiPoints prims now support three falloff modes:

    - "none" - no falloff. This is equivalent to the old approach with falloffpower = 0.0
    - "power" - power based falloff. This is equivalent to the old approach.
    - "spline" - spline based falloff. This is a new mode which supports multiple types of spline shapes, matching the OSL spline implementation.

    A new primvar [constant string falloffmode] ["none|power|spline"] selects which mode to use. (If this primvar is absent, compatibility code will ensure the previous behavior is followed.) If spline mode is selected, the following three primvars control the falloff shaping:

    - [constant string falloffsplinetype] ["constant|linear|bezier|bspline|catmull-rom|hermite"]
    - [constant float[N] falloffsplineknots] [<array of N knot values between zero and one>]
    - [constant float[N] falloffsplinevalues] [<array of N values between zero and one>]

    Values of N between 1 and 16 are supported. Extra knots will be inserted at the beginning and end of the curves to ensure the spline tangents are horizontal at the start and end. The spline evaluation can be expensive, so a table lookup is used by the implementation to reduce costs. 1k entries are used with a linear lookup, which in testing has been found to be artifact-free for all reasonable point sizes.
- Users can now use namespaced primvars. Primvars which start with the namespaces used by standard Renderman primvars will still be culled (for example the ri/grouping/dice/trace namespaces). RMAN-21578

Image Output and Displays

> ❗ The "asrgba" parameter to the OpenEXR and DeepEXR display driver now correctly defaults to 1, matching the Args file. This only changes behavior in RfK or RIB files where the asrgba parameter is not explicitly set. RMAN-21883

- Fixed an issue where changing the pixel variance in Solaris had no effect in RIS renders. RMAN-21296
- Added "raytracebakedisplacement" option for baking. This option helps prevent artifacts when baking using trace() calling patterns, such as PxrDirt and PxrOcclusion. RMAN-20743
- OpenEXR and DeepEXR autocrop can now use any AOV which has alpha as its source. DeepEXR will also no longer crash if autocrop is enabled without an alpha channel. RMAN-21546
- Add two new parameters to the OpenEXR driver: "string asrgbacolorchannel" and "string asrgbaalphachannel". If supplied, these strings automatically force "asrgba" to be 1, and then serve as the actual channels that get converted to "RGB" and "A" instead of "Ci" and "a".

Lights

- Greatly improved the response of live render light edits in the face of thousands of lights in the scene. PRMAN-2761

Volumes

- Attribute "volume" "temporalmethod" is no longer supported. The only method of generating temporal data for volume rendering is the Eulerian method.
- Fixed a crash that would happen when the "Light Source" parameter of a PxrVolume is enabled while the stats json listener is also enabled. RMAN-21054
- Reduced memory usage of volumes rendered using the impl_openvdb plugin. RMAN-20942

Integrators

- PxrUnified now supports Attribute "trace" "reflectsubset", "reflectexcludesubset", "transmitsubset", and "transmitexcludesubset" in fashion identical to PxrPathTracer.

Stylized Looks

PxrStylizedCanvas:

- New Display Filter for layering, background colors and textures

PxrStylizedControl:

- Toon Artistic Mode control (Light Rotation vector, Trace Shadow) has been added
- New Color Ramp in Toon Artistic Mode
- New Toon Step Mode for Steps vs Color Ramp
- Conditional visibility on UI for Toon_Step_Mode
- Comp modes for Sections Map (Replace/Multiply/Plus/Screen/Difference)
- Quantize Curvature for Wireframe Lines
- Bend Normals towards Light
- Hatching Frequency Scaling control (by UV/ST/Facing Ratio)
- NPRtoonOut AOV

- Result Visualizer (I vector, UV, st, Curvature)
- Sections HSV Switch

PxrStylizedToon:

- New Artistic Mode (from Display Filter and PxrStylizedControl override)
- Step Modes integrating Color Ramp
- HSV on all Step Modes to match Lights & Darks colors
- Compositing Modes (Over/Plus/Multiply/Screen)
- New Smooth functionality to antialias between Toon steps & light terminator
- UI rework

PxrStylizedHatching:

- New Ignore Alpha switch for Canvas BG Color & Texture
- Color Blend Signal in Result Visualizer (from switch)
- Defaults updated for layering
- New Frequency Scale by PxrStylizedControl
- UI rework

PxrStylizedLines:

- New Activation Function "None"
- New Activation Function for Edge detectoin: Slope Threshold (in addition to existing Spline and Sigmoid)
- Line Detect Multiplier extended to all Activation Modes (None, Spline, Slope/Threshold, Sigmoid)
- New HSV Albedo to tint for line color (outside of Light Response)
- Fix NAN's on Distort when pulling data off-render
- New 2x2 filtering when Distort enabled for smooth lines
- Mix feature in Compositing section

Shading

- Integer primvars can now be properly accessed from OSL. RMAN-20966

Textures

- The scalability of the texture cache when using many threads on RIS has been greatly improved. On machines with many cores, this may result in a substantially faster time to first pixel for scenes with many textures. RMAN-20729
- An issue with suboptimal file performance using OpenEXR textures, especially noticeable when using network-mounted file systems, has been addressed. RMAN-21083

Statistics

- The (new stats) memory tracking label for the RIS raytracing component has been renamed from "rman/ris/rrrenderer/..." to "rman/raytracing/..."
- The progress outputs during baking have been improved: they are now more informative about how many objects and baking tasks remain.

APIs

Please consult API Changes from 25.X to 26.X for information on updated developer APIs between release 25 and release 26.

Bridge Tools

RenderMan For Katana

- New features:
    - **XPU selection for batch rendering**
        - RfK now lets users set the renderVariant setting in PrmanGlobalStatements to either XPU - CPU or XPU - GPU for cases where you need to render with only the cpu or only the gpu.
        - Note that the settings in the Preferences menu will still take precedence for live and preview renders, unless the new "interactiveXPUSettingsOverride" preference is disabled.
        - For batch renders, we also added a way to override the renderVariant and the gpu selection using command line options, i.e. katana --batch --katana-file=myfile.katana --render-node=Render -t 1 -- --prmanRenderVariant=xpugpu --prmanGpuSelection=0. Please note that the -- needs to precede the prman command line arguments.
    - **Support for Katana 6.5**
        - The new openvdbasset style of volumes is supported. This enables importing volumes with UsdIn.
        - RenderMan Hydra Render Delegates are supported in this version (not supported in the RenderMan 25 version of 6.5)
    - **Interactive Denoiser** - You can now use denoise live renders in the Katana Monitor when rendering with RIS. To enable interactive denoising, enable the checkbox PrmanGlobalStatements. Extra settings for the denoiser are available in the renderman section of the Katana Preferences.
- hdPrman in the Viewer Fixes:
    - Fixed a crash when switching between different RenderMan delegates in the Viewer.
    - Fixed bug where image files without the .tex extension would sometimes be ignored.
    - Fixed an issue that prevented usd lights from rendering in Katana 6.0
- Other fixes:

    - Fixed the paths in the denoise_teapots.katana example file so that it works on Windows.
    - Fixed an issue where the RenderMan Denoiser window would freeze the Katana UI when opened with the shelf script.
    - The PrmanStylizedLooks macro has been updated to support new features. You will need to recreate the node in your scenes to get the new required AOVs.

RenderMan For Houdini - Solaris

- New Features:
    - Added new PxrLightFilter, PxrDisplayFilter, PxrSampleFilter, PxrIntegrator LOP nodes
    - Added Support for PxrMeshLight in Solaris (Linux and macOS only for now). There is a new LOP node to set a mesh as a mesh light
    - Support pinned curves in Houdini 19.5
    - Visualizer node (pattern soloing) now supported. This also supports soloing Lama BXDF nodes in a network.
- Changes
    - Nested instancing is now supported for instancer nodes, which can bring significant performance improvement.
    - Significantly improved performance of processing of many instances at the start of a render.
    - Improved support for velocity motion blur.
    - Husk --snapshot 0 disables checkpointing.
    - New "import into selected material library" behavior in the preset browser
    - Rendersettings should match RenderMan defaults more closely.
    - Volume "fieldIndex" attribute is now supported.
    - No longer need array node to set MatteID.
    - Support for rendering OpenEXR images with overscan.
    - Support velocity and angularVelocity on point instancers.
- Bug Fixes
    - UsdSkel no longer culls time samples after 4.
    - For the RenderMan LOP, fixed LPE Lobe Mappings parameters so they are correctly passed to the USD Stage.
    - "light group" and "visible in refraction" light attibutes now correctly working.
    - Fixed a bug where materialX shaders wouldn't be correctly generated if an upstream node was both part of a pattern network and also directly connected to the material node.
    - Fixed a bug in UsdPreviewSurface support for normal maps, which could lead to artifacts.

RenderMan For Houdini - Classic

> The utilityPattern of PxrSurface now accepts both int and int array inputs in order to support MatteID and Stylized Looks in Solaris. However, due to a bug in the Houdini UI, existing int array connections are lost and must be reconnected. If an array is not necessary, we recommend connecting the integer output directly.

- Bump to roughness textures correctly processed in RfH.
- A bug that caused RIB generation to fail via hbatch has been addressed.
- Stylized AOVs are added in UI.

RenderMan For Maya

- Bug Fixes:
    - Fixed a crash when unable to checkout a license for the material viewer and swatch renders.
    - Fixed a bug that was causing a wrong filename for the denoiser when illegal characters (ex: ':') were in the camera name.
    - Fixed bug where rman materials would lose connections when exported from maya as USD.
    - Fixed a bug where presets saved for stylized looks wouldn't import correctly.
    - Fixed a bug where mayabatch renders were not respecting the integrator type from the globals.
    - Fixed a bug where on Windows textures weren't always converted properly for shading networks generated by Substance for Maya.

RenderMan For Blender

- New Features:
    - Blender 3.6 and 4.1 are now officially supported. Note, the Qt framework is not supported in 4.1
    - Viewport renders can now use the RenderMan AI denoiser as well as the OptiX denoiser. See the Blender denoising page for more information.
    - Add a new menu item in the shader graph to convert selected Cycles image nodes to PxrTexture nodes. This is useful when dragging and dropping image files from the file
      browser to the shader graph.
- Changes:
    - We now pay attention to the Viewport and Render levels when using the subdivision modifier. Note, the subdivision scheme setting on the object data properties still takes precedence.
    - A new json filename statistics option has been added to the workspace editor. This should be used in favor over the older XML filename.
    - The drawing the barn light filter projection in the viewport when in physical mode has been disabled
    - A LamaSurface node will automatically be created for you, when attach a Lama node bxdf to your object.
    - The addon will now present a dialog when importing presets that include display filters.
    - The "volume:temporalmethod" primvar has been removed.
    - The package scene operator will now print an error message if the directory selected is not empty.
- Bug Fixes:
    - Fix a bug where importing presets with array parameters failed.
    - Fix the orientation of the PxrEnvDayLight. The sun was rising and setting in the wrong directions. Note, older scenes will change with this fix
    - Fixed a bug where RIB files were not getting created correctly on Windows.
    - Fixed a bug with relative paths not working with OpenVDB files, on Windows.
    - Fixed a bug where lights, lightfilters and RenderMan objects were not created at the cursor location.

  - An issue that caused a slowdown when rendering hair curves object has been addressed.
  - Fixed a bug where using "name" for the layer parameter in PxrCryptomatte didn't work.
  - Fixed a bug where the "attribute" parameter would not show up when using PxrCryptomatte.
  - Fixed a bug where trace sets didn't updated correctly during IPR.
  - A bug that caused metaballs to not render when using Blender 3.6 has been fixed.
  - A bug that caused an error when dealing with meshes with multiple materials and using boolean modifiers has been addressed.
  - Fix a bug where making the wrong connection to a RenderManOutput node would lead to a crash
  - Fixed a stretching issue with dome light texturing in the viewport when using a newer Blender (3.3 and above)
  - A bug that caused a crash when using a hair particle system with equal root and tip diameters has been addressed.
  - A number of bugs when using the PxrOSL node have been addressed, including IPR not updating correctly when a parameter value was changed.
  - A bug that caused material updates to not work on a Volume Box object has been addressed.
  - A bug that caused hair to look shorter than the GL view when using the newer hair curves node has been addressed.
  - Fixed a bug when the preset browser would fail to import the Stylized Toon 2 Steps preset.
  - A bug that caused "it" to display the wrong Pixel Variance value when rendering with IPR has been addressed.
  - A bug that caused PxrRamp nodes to not update during IPR when they are inside of group nodes has been fixed.
  - A bug that caused assets to fail to export in the preset browser, when you have a color to float connection has been addressed.
  - A bug that caused meshes with the subdivision modifier attached to not render properly when the Blender viewport GPU subdivision option was turned on has been worked around.
  - A bug that caused chaining of PxrOSL nodes to fail has been fixed.

Known Limitations

Installation + Licensing

- macOS 11 and beyond: The installer can download the packages from Pixar, but can't run them to get them unpacked. You must manually double-click on the .pkg files that the installer leaves in the Downloads folder. RMAN-18802

Interactive/Live Rendering Limitations

- Bucket order or size cannot be changed during live rendering.
- Changes to Presence do not update when using the opacity cache option (RIS only).
- Objects are not re-diced during interactive camera edits.
- Mesh lights cannot be interchanged as geometry without restarting.

Denoiser

- macOS/Apple Silicon: The denoiser is not supported on Apple Silicon, even with Rosetta. RMAN-20408
- If your image has NaNs from either RIS or XPU, the denoiser will fail on macOS and Windows.
- The interactive denoiser currently is not supported in XPU.

XPU

- The GPU portion of XPU can render artifacts with trace:reflectexcludesubset. RMAN-20393
- Support for homogeneous single scattering in PxrSurface is incomplete. Renders will be noticeably incorrect after the first bounce off the interior of a closed object. RMAN-16374
- Lights inside objects with homogeneous single scattering in PxrSurface will not compute the correct shadows. RMAN-16367
- Due to differences in how XPU computes derivatives for normals, XPU may compute different results from RIS for bump mapped subdivision surfaces in areas of highly concave curvature, combined with a high bump scale. This happens especially when the bump mapping cannot reasonably approximate the displacement. Note that this difference does not occur with displacement. The issue can be ameliorated by reducing the bump scale. RMAN-21977
- Please refer to the XPU section of the documentation for a more detailed current list of limitations.

RIS and XPU

- Color Management: Although we have upgraded to the VFX Reference Platform 2023, there is an outstanding issue with the Image Tool described below.

> The Image Tool is currently unable to read OCIO v2 configurations and is locked to the $RMANTREE/lib provided configuration. If you have a v1 config you would like to use set the environment variable IT_OCIOV1.

- MaterialX: Material layering is not supported. RMAN-20365

RIS

- Shading
  - PxrUnified integrator does not yet support all the standard rendering features.
  - <primstr:nameofvalue> substitution of data from a constant primvar or user attribute on an object for dynamic filename substitution is not yet working for the gettextureinfo() OSL call.

    - Using the `'.'` character in the handle for an OSL shader could cause unpredictable results during re-rendering.
    - The RixTexture::TextureDerivatives calls always returns RixTexture::InvalidFile for non-texture-atlas files. This will be restored in 26.1.
- General rendering
    - Load on demand procedurals are not supported anymore, all procedurals are now loaded immediately.
    - RenderMan does not read point data from OpenVDB files.
    - Per-Instance baking is not supported, only the reference instance.
    - PxrBakePointCloud cannot directly render ptex.
    - Sample + Display filter plug-ins do not have access to lighting services for light dependent effects, e.g. lens flare.
    - Adding new mesh light on existing geometry during IPR results in double geometry.
    - Motion blurred polygons do not motion blur normals when deformed. Use Subdivision meshes instead.
    - When attempting to access an array primvar, you must first check the size of the array primvar and allocate the appropriate space. Not doing so may lead to a crash.
    - Points and curves cannot be used as geometric lights.
    - Analytical lights placed inside non-aggregate volumes may yield artifacts when made visible to the camera. As a work around, the light camera visibility should be turned off, and a geometry with a similar shape should be used (visible to camera, invisible to transmission and indirect rays), with the proper emissive bxdf.
    - Indirect Path Guiding in the PxrUnified integrator causes a crash.
- Bridge Products:
    - RfH: Soloing MaterialX Lama nodes in complex shading networks can give an incorrect result.

RenderMan for Houdini

- When batch rendering from a RIB exported from Houdini, the searchpath may contain an invalid path that causes RenderMan to load the wrong OpenVDB shared library, causing batch renders to fail. The workaround is to edit the "procedural" searchpath in the RIB to $RMANTREE/lib/plugins instead of the one in the RenderMan for Houdini directory. RMAN-20297
- RfSolaris: Texture memory limits are not properly respected. RMAN-20563
- Denoise ROP does not work in H20, py3.10. In py3.9, it will work if loppath on the usdrender_rop LOP node is set to opinput('.', 0) - RMAN-21926

RenderMan for Maya

- F-Stop is divided by 10X with PxrCamera. RMAN-20475
- Overriding PYTHONPATH can cause denoising to fail. RMAN-20546