

4.2.8.0

Milestone 4.2.8.0

Enhancements

- **Improved volume multiple scattering:** The importance sampling for volume indirect rays has been modified, increasing render time a little, but significantly reducing noise fireflies in various cases. (#4599)
- **Faster .ass writing:** Writing large .ass files is now 40% faster without compression and 8% faster with .gz compression. Limits on the length of string parameter values have been removed from the .ass parser as well. (#4780, #634)
- **Object space adaptive tessellation:** Polymeshes have a new subdiv_adaptive_space attribute for subdivision surfaces. subdiv_adaptive_space raster corresponds to the previous behaviour and subdiv_adaptive_space object corresponds to adaptive subdivision in object space. This is useful for instances or when adaptive tessellation is desirable regardless of a specific camera. In line with this new option, subdiv_pixel_error is now called subdiv_adaptive_error. Also, Arnold will warn when raster-space tessellation is used on more than one visible instance. (#4758, #4759)
- **Wireframe support for non-linear cameras:** Wireframe renders now preserve line width when using non-linear cameras such as the built-in spherical and fisheye cameras. (#4633)
- **Atmosphere shaders with RGBA output:** Atmosphere shaders can now output alpha through AtShaderGlobals.output.RGBA.a. This is only useful when implementing atmosphere shaders that act as mattes. (#4742)
- **image texture offsets:** The image node now has soffset and toffset parameters for offsetting the textures. This offset takes place before scaling, flipping, or swapping of the S and T coordinates. (#4788)
- **Opacity blending supported for FLOAT AOVs:** Opacity blending is now supported for AOVs of type FLOAT, in addition to the previously supported RGB and RGBA. (#4772)
- **Integrator details in logs:** The log files now include more information about the number of samples, GI depths and transparency. Per light volume sampling information was also added. Also, there have been a number of small cleanups and fixes in log files, like removing the legacy, detailed SSS stats, which should result in cleaner, shorter, more reliable and informative logs. (#4721, #4752)
- **Updated to RLM 11.3BL1:** We have upgraded the license server and the external library controlling the licensing subsystem from version 11.2BL2 to 11.3BL1, a more stable release fixing various crashes, bugs, hangs and memory leaks. (#4730)

API additions

- **AtString hash:** An AtString.hash() function and AtStringHash functor have been added, which return a precomputed string hash for usage in hash tables likestd::unordered_map. (#4748)
- **Unset message:** AiStateUnsetMsg*() functions have been added to remove shader messages from the message bank. (#4747)
- **Set metadata:** Metadata can now be dynamically set on node types by using AiMetaDataSet with an AtNodeEntry. Previously setting metadata was only possible innode_parameters. (#4766)
- **New procedural plugin methods:** Procedural plugins can now set three additional, optional methods. First, a bounds initialization method allowing the plugin to specify bounds automatically and efficiently for nodes it will generate, rather than relying on the user's specified bounds or on full expansion. Second, plugin initialization and cleanup methods run once for the render session. In order to get access to the plugin-wide data that can be set up during plugin initialization, an additional API function has been provided for the other procedural plugin methods. (#4320, #4769)

```
AI_API void* AiProceduralGetPluginData(const AtNode*);
typedef bool (*AtProcInitPlugin)(void** plugin_user_ptr);
typedef bool (*AtProcCleanupPlugin)(void* plugin_user_ptr);
typedef bool (*AtProcInitBounds)(AtNode* node, AtBBox* bounds, void** user_ptr);
```

Incompatible changes

- **Deep EXR files use (AR, AG, AB) as opacity channels:** Deep OpenEXR files were using the old (RA, GA, BA) names for three-channel opacity. We have switched to (AR, AG, AB) to follow the EXR standard. Note that Nuke (as of 9v6) does not yet correctly display these deep files anyway. (#4726)
- **single_channel texture lookups:** Instead of returning it in the R channel only, the single channel is now returned in the R,G,B channels as monochrome RGB. This affects AiTextureAccess, AiTextureHandleAccess and the image node. (#4741)
- **Default sky.intensity:** The default intensity for the sky background shader has been changed from 0.0 to 1.0. It's unlikely that this will affect old scenes for many users, as this shader was useless with the default intensity. (#4754)
- **Too-small procedural bounds now an error:** Bounds specified for procedurals that end up being too small (and therefore potentially clipping away parts of the object) have been upgraded from a warning to an error. With options.abort_on_error turned on, this will halt the render, which is safer and more reliable than random, non-deterministic, hard to debug clipping of these objects. (#4761)
- **Removed AI_LOG_SSS:** This bit, part of the AI_LOG_ALL bitmask in ai_msg.h, was unused since SSS pointclouds were removed years ago. (#4794)

Bug fixes

Ticket	Summary
#4734	Crash during IPR session with Maya on Linux
#3350	AiShaderGlobalsApplyOpacity() not converging on correct result with colored opacity
#4564	Escape sequences in .ass parser not working with string arrays
#4724	Crash with volumes when the P AOV is enabled and an object is behind or inside
#4725	Crash when using deep volumes without an RGBA AOV
#4726	rename "RA" "GA" "BA" OpenEXR opacity channels to "AR" "AG" "AB"
#4728	work scheduler lifetime extension causes crash in yeti
#4729	Rare random buckets rendering too bright
#4731	Missing header info in logs
#4737	Incorrect transformation for lights in motion blurred procedurals
#4741	return monochrome colors from image shader in single_channel mode
#4744	Built-in surface AOVs disappear inside atmosphere shader
#4746	Single channel alpha mismatch in deep driver for volumes
#4750	dPdv on curves is numerically unstable far away from the origin
#4753	BVH motion count in stats is wrong
#4755	fog is not always visible when seen from below the fog plane
#4762	AtShaderGlobals.area does not account for dOdx and dOdy
#4763	Occasional crash when curves generate invalid normals
#4764	crash in subdiv with vertex that exceeds the max valence of 255
#4768	Fix ray differentials in free mode
#4770	AiAOVRegister does not correctly detect unsupported types for AI_AOV_BLEND_OPACITY
#4771	AiMakeRay should compute a more numerically robust AtRay::mindist
#4773	non-deterministic behavior of user data declarations on cached procedural data
#4779	procedural cache wrongly propagates visibility overrides
#4780	Error writing long string parameter values to .ass file
#4792	Crash when re-creating a previously destroyed procedural
#4795	image node with UDIMs with alternate UVs are incorrect
#4798	Random crash with displacement and autobump
#4749	Renaming output in kick fails when a camera is specified in the output
#4765	Python API crash when AtMsgCallback goes out of scope