

3.3.7.0

Milestone 3.3.7

Enhancements

- **New cylindrical_light:** There is a new type of light available to lighters through a new node called `cylindrical_light`. Like their cylinder shape counterparts, these lights are defined with two endpoints and a radius. These new lights should fully support all of the features available to lights in Arnold like motion-blur and MIS, excepting volumetrics/atmospherics which should will be supported in a future version. (trac#2195)
- **MIS support in diffuse layer of standard shader:** The necessary functions for MIS to work with the Oren-Nayar BRDF used by the standard shader have been implemented. Apart from being used internally by this shader, these functions are also available to shader writers in `ai_shader_util.h`, where the rest of the API-provided MIS functions are located. (trac#2251)
- **MIS support in distant_light:** Distant lights now correctly implement the required functionality for MIS to work with this type of lights. This should allow for cleaner specular highlights in glossy reflections of distant lights with angle > 0. (trac#2164)
- **Optimized importance sampling table precomputation:** The precomputation of the importance sampling tables for `quad_light` and `skydome_light` is now multithreaded. Any rendering threads which sample a `quad_light` or `skydome_light` that has not completed this precomputation will aid in computing the table, rather than wait for one thread to compute the table (as was the previous behavior). (trac#2258, trac#2266) The precomputation was made deterministic under threading and more numerically stable (output may change slightly), and is now safeguarded against NaN texels. (trac#2292)
- **Added ray differentials for all camera types:** Non-perspective camera projections such as fisheye and cylindrical now set derivatives properly, which allows texture filtering, wireframe shading and bump mapping to work correctly for shaders seen through these types of cameras. (trac#1234)
- **improved SSS precomputation with motion blur:** Arnold now uses a global absolute time to perform pointcloud lighting calculations. Previously the first key in the mesh would be used, but this can lead to artifacts when using centered frame motion blur for example because the first key is the position of the model at the previous frame. Now the mesh is positioned according to the global option `sss_time` which default to 0 both for backwards compatibility and more consistent behavior with centered frame motion blur (trac#2284)
- **Environment variable expansion in searchpaths:** The `texture_searchpath`, `procedural_searchpath` and `shader_searchpath` options now support expansion of environment variables delimited by square brackets. This works both using the API and inside `.ass` files as shown below: (trac#1816)

```
AtNode *options = AiUniverseGetOptions();
...
AiNodeSetStr(options, "procedural_searchpath", "[MY_ENVAR_PATH]/to/somewhere");

options
{
  AA_samples 4
  GI_diffuse_samples 4
  ...
  procedural_searchpath "[MY_ENVAR_PATH]/to/somewhere"
}
```

- **pykick enhancements:** There are several enhancements in how `pykick.py` is used and deployed: (trac#2271)
 - `pykick.py` is now an executable Python script which can be invoked directly via a new symbolic link `pykick` located in the bin directory. The symbolic link is only available in Linux/OSX.
 - `pykick.py` now points to the location of the Arnold Python bindings, so the user is no longer required to include that location in the `PYTHONPATH` environment variable.
 - Since the Arnold Python bindings use the `ctypes` module (introduced in Python 2.5), `pykick.py` now checks the current version of the Python interpreter used.

API additions

- **AiColor():** Added new convenience function that takes a float value and creates a color with all components set to that value (trac#2268)

Incompatible changes

- **auto_transparency_probabilistic now enabled by default:** The default value has changed from OFF to ON. This could affect the look of scenes containing semi-transparent or `min_pixel_width` curve shapes. (trac#2276)

Bug fixes

Ticket	Summary	Component	Owner	Priority	Version	Created
#2255	kick should only close its window when pressing ESC	kick	xo	critical	3.3	3 weeks
#2191	Random crashes with points.min_pixel_width > 0 and 1-element array in points.radius	arnold	oscar	major	3.3	2 months
#2245	Intel compiler floating point inconsistencies in Windows	arnold	xo	major	3.3	4 weeks
#2247	Problem loading some .OBJ format files	arnold	angel	major	3.3	4 weeks
#2249	Intersection precision problems in far away point primitives	arnold	xo	major	3.3	3 weeks
#2254	Intersection precision problems in far away sphere primitives	arnold	xo	major	3.3	3 weeks
#2259	incorrect MIS variable computation in textured, importance-sampled quad lights	arnold	alan	major	3.3	2 weeks
#2262	'min_pixel_width' in 'points' primitive is not properly working	arnold	oscar	major	3.3	2 weeks
#2265	AiEvaluateLightSample() is modifying lighting shader globals	arnold	alan	major	3.3	13 days
#2270	'maketx' and 'pykick' are not executables in linux releases	arnold	oscar	major	3.3	8 days
#2277	Buckets have different intensities when rendering with MIS and subdivision	arnold	xo	major	3.3	7 days
#2280	Linux/OSX kick process doesn't properly exit after display window is closed	kick	oscar	major	3.3	6 days
#2284	sss precomputation should occur at a fixed time	arnold	marcos	major	3.3	3 days
#2292	Crash when using a textured importance sampled light with NaN texels	arnold	xo	major	3.3	26 hours
#2293	Crash when opening a non-existent '.ass' file	arnold	oscar	major	3.3	25 hours
#2243	Rename "focal_distance" to "focus_distance" in "fisheye_camera"	arnold	oscar	minor	3.3	4 weeks
#2267	Windows kick process doesn't exit after display window is closed	kick	xo	minor	3.3	10 days