

Segment 01 FLUX: An approximate image based lighting tool for Maya – *Plug in Development*

Technology C++, Maya API, MEL, Boost, OpenGL, OpenMP, OpenExr

Description **Main Features:**

- Less Noise in Render
- Less Render-time
- Editable and Key able
- Precise Control over the result
- Ideal for lighting Fur, Hair, Fluids and matching lights with Live action
- Combines and interpolates multiple sets of HDRI
- Pipeline friendly; Exports centroids data to Maya's native data structure
- Artists friendly; Customized "Attribute Editor Templates" for easy node connection and traversal
- Fast and responsive; Multi-threaded and accelerated using built-in kdTree
- Interactive modification of lights properties
- Output statistics, verbose, diagnostic data

Flux Pipeline:

Stage (1) - Importing data

- **[fluxLoader]:** Supports all type of openExr files (Tiled, Scanline, Multichannel) with channel shuffling and extra bit info; Built-in support for Latitude-Longitude and Cube environment maps.

Stage (2) - Pre-processing

- **[fluxLuminance]:** Computes luminance energy of input data with compensation at poles
- **[fluxSat]:** Accelerate computation using SAT (Summed Area Tables)

Stage (3) - Centroids Generation

- **[fluxMedianCut]:** Adaptive and fixed importance sampling using MedianCut algorithms with an optional pre-processing Barlet filtering
- **[fluxHammerslayHalton]:** Importance sampling with Hammerslay / Halton sequences and computing energy based on Voronoi patterns
- **[fluxUserInput]:** Generates centroids based on user's input, UVs, Radius, Falloff; Computing luminance/color by using Mean, Normal or Sum operator.

Stage (4) - Visualizing

- **[fluxCentroidsViz]:** Visualizes centroids data in viewport including Color, Luminance scale, Magnitude, IDs, Centroids orientation

Stage (5) - Centroids Editing

- **[fluxCentroidsTransform]:** Centroids transformation including translation, rotation, radius and orientation; Meta information including Location, description and centroids count
- **[fluxCentroidsField]:** Generates centroids field by combining multiple sets using weight balance or union interpolation
- **[fluxCentroidsEdit]:** Clustering and selecting centroids by Luminance, Color, UV ranges, Volume and IDs; Cleanup and deleting centroids; Editing centroids color and luminance using stackable operators with priority feature; including Generic, Set-Range, Normalize, Clamp, Override (Axial) and remapping in RGB or HSV color space

Stage (6) - Exporting data

- **[fluxCentroidsExport]:** Exports centroids data to Maya's native data structure including Matrices, Colors, Energies, UVs; Ideal for transferring data to shaders, lights, etc.
- **[fluxGlobal]:** Global container for flux's IBL settings, lights rigs and flux's Mentalray shader
- **[fluxMentalrayIBL]:** Utilizing centroids data inside Mentalray
- **[fluxLightsGeneration]:** Generates light rigs based on centroids data and global container nodes

Segment 02 Tornado: Case study - *Plug in Development*

Technology C++, Maya API, MEL

Description **[dnaTransferFluidAttributes]**: Transfers particle's scalar and vector including customPP attributes to fluid containers within Add, Replace and Multiple modes. Interpolates data using linstep, smoothstep and hermite functions.

[dnaTransformInterp]: Interpolates transformation matrix between arrays of matrices

[dnaRampInfo]: Maps array of attributes using curves

In this example **dnaTransferFluidAttributes** was used to transfer particles' attributes from tornado rig to fluid container; **dnaTransformInterp** was used to interpolate surface between main controllers (Stretchy Setup) and **dnaRampInfo** was used to add extra details to the surface of tornado.

Segment 03 Mathematic and Utility nodes for Maya - *Plug in Development*

Technology C++, Maya API, MEL, Boost

Description It is composed of a set of mathematics and utility nodes (over 60 functions) for Maya to control shading networks, Key frame and procedural animation, character or vehicles rigs without writing expression. This will result in a faster evaluation of DG network; Also supports Maya's native renderer and Mental ray.

- **[Angular]**: Computes value to radian/ degree/ minutes/ seconds
- **[AnimcurveSpeed]**: Computes instant and constant speed of animcurve based on given input time
- **[Cardinal]**: Computes Cardinal and Hyperbolic Cardinal
- **[ConditionRange]**: Tests the relation between input value and specified range (Inclusive and exclusive)
- **[Constant]**: Pre-defined constant including pi, e, golden ration, etc.
- **[Curvefunc]**: Smooth, incrementing transition between values (Linstep, Smoothstep, Polynomial hermite)
- **[Exponential]**: Exponential functions including exp, log, log10, cubed root, square root, etc.
- **[Factorial]**: Computes rising and falling factorial of base input
- **[Limit]**: Imposes limit on values including abs, ceil, floor, sign, trunc, fraction, round
- **[LimitAnimCurve]**: Returns min or max value of animcurve keyframes
- **[LimitMinMax]**: Finds minimum and maximum of value(s)
- **[Parabolic]**: Solves general form of 2nd degree Polynomial equation including root, delta, vertex
- **[TLGamma]**: Computes true, log and log derivative gamma of input base
- **[Trigonometric]**: Computes trigonometric value(s) of input, including sine, sine hyperbolic, arc sine, cosine, cosine hyperbolic, arc cosine, tangent, tangent hyperbolic, arc tangent and hypot.

Segment 04 meshToXML - *Plug in Development*

Technology C++, MAYA API, MEL, LibXml

Description Imports / Exports geometry description including Multiple UV and Color sets and Blind data to custom formatted XML structure. meshToXML can be used for:

- Archives geometries in standard format compatible with database engines and applications
- Exchanges geometries with 3D Applications
- Transfers game Levels to game engines
- Shares 3D meshes across the web