RELEASE NOTES FOR NUKE 6.3v4

Release Date

23 September 2011

Supported Operating Systems

- Mac OS X 10.5 "Leopard" and 10.6 "Snow Leopard" (64-bit)
- Windows XP64, Windows 7 64-bit
- Linux RHEL 5.4 64-bit

Note *Mac OS X 10.7 "Lion" is currently not supported as an operating system for Nuke.*

New Features

There are no new features in this release. For more information on changes in

- Nuke 6.3v3, see "6.3v3" on page 11.
- Nuke 6.3v2, see "6.3v2" on page 15.
- Nuke 6.3v1, see "6.3v1" on page 17.

Feature Enhancements

Documentation

- The Nuke Reference Guide now covers these nodes:
  - Filter > BumpBoss
  - Filter > Defocus

Bug Fixes

- BUG ID 21707 - The Convolve node caused Nuke to crash with a customer script.

Known Issues and Workarounds

This section covers current known issues and gives workarounds for them.

Known Issues Specific to Nuke 6.3

- BUG ID 16816 - Using certain graphics cards configuration under Linux, interlaced stereo views might be flipped. A new Viewer preference was added for manually flipping interlaced stereo views as necessary.
- BUG ID 20407- Changing the file name in a ReadGeo node does not currently update in the Viewer.
- BUG ID 20431 - CameraTracker: It's not possible to pick colors in the Viewer with the control panel open.
- BUG ID 20489 - Linux: Nuke may crash when running a script with the **texture mode** control set to **multiframe** in the Preferences. The workaround is to set **texture mode** to **classic**.
- BUG ID 20550 - Mac OS X 10.7 (Lion): A segmentation fault occurs when exiting Nuke.
- BUG ID 21663 - Read: After reading in a stereo/multiview EXR file and choosing not to add new views to the project, subsequent reads of any stereo/multiview EXR files won't give the option to add new views.

AudioRead

- BUG ID 18213 - Changing **Draw Style** in the Curve Editor or Dope Sheet doesn't redraw the curves correctly.
- BUG ID 18217 - Redo keyframe generation doesn't replace the keys.
- BUG ID 18451 - Flipbooking doesn't honor the time range knob.
- BUG ID 18465 - The **time range** control doesn't limit the range when an **endtime** is specified.
- BUG ID 18666 - Changing the sample rate has no effect on playback in a Flipbook.
- BUG ID 18924 - Changes to the **rate** control value are not honored when **ratesource** is toggled between **file** and **custom**.

Deep Compositing

- BUG ID 17436 - DeepRecolor doesn't normalize correctly when the alpha from a flat image doesn't match up with deep opacity.

Particles

- BUG ID 17104 - If a Constant node is used for the **particle** input, its color is not shown in the 3D view.
- BUG ID 17243 - Using sprites instead of geometry representations causes particles to render behind the 3D grid lines.
- BUG ID 17520 - Geometry representation textures are displayed even when **display** is set to **off**.
- BUG ID 18044 - Emitting from **edges** or **faces** using the **uniformly** or **in order** options, still emits particles from **points**.

- BUG ID 18268 - Enabling **color from texture** always renders a solid white alpha for the texture.
- BUG ID 19965 - You cannot currently change the opacity of geometry particles over time.
- BUG ID 19977 - ParticleBounce: Internal bounce can leak particles outside the limiting primitive when two ParticleBounce nodes are used simultaneously in a Particle system.
- BUG ID 20057 - Attaching a Cube to the ParticleEmitter **particle** input adds a cube frame to the Viewer overlay.
- BUG ID 20171 - Scrubbing performance is occasionally poor when viewing particles in 2D.

PlanarTracker

- BUG ID 19240 - When Viewer **downrez** is greater than 1, the **clearTrackDataForward** button currently offsets bezier shapes to fit the root format.

  To return the shape to its original size and position, advance the sequence one frame.
- BUG ID 19562 - When you delete a PlanarTrackLayer from the Roto curve list you cannot undo using the Roto Properties undo button.

  Use **Ctrl/Cmd+Z** or the **Edit** > **Undo** function.
- BUG ID 19563 - PlanarTracker: Detected or tracked features are not offset by the origin of the input image's bounding box.

  To avoid this, you can add a Crop node before PlanarTracker's associated Roto node.
- BUG ID 19683 - After adjusting the planar surface using the **Correct Plane** Viewer button, you have to undo twice for the planar surface to return to its original position.
- BUG ID 20208 - There is currently no option in the PlanarTracker Viewer toolbar CornerPin2D export button for a CornerPin2D (relative) node that links to the PlanarTracker node. The **motion blur** control in a CornerPin2D (relative) node created from this Viewer toolbar button does not produce any motion blur as there are no animated **to** point position values.

  As a workaround, use the **Create** button in the relevant PlanarTracker node's **Export** section with the **export** option set to **CornerPin2D (relative)**. The animated **to** values are populated, resulting in motion blur if the appropriate option is utilized.

Warpers

- BUG ID 17697 - GridWarp: In stereo mode, the right view is labelled as **default** when you split controls into separate views.

- BUG ID 18012 - GridWarp: The grid can flicker between white and grey when zooming in and out of the Viewer, particularly when the grid is subdivided.
- BUG ID 18019 - GridWarp: You cannot select all grid points in the Curve Editor using **Ctrl/Cmd+A** shortcut.
- BUG ID 18203 - When **persistent preview** is active, the Viewer always shows **rgb** when **channels** is set to different sets.
- BUG ID 18270 - SplineWarp: Expression linking Tracker transform values to control points produces incorrect translations.
- BUG ID 18304 - GridWarp: Existing control point keyframes are not cleared when you draw a new grid using the **Draw Boundary** Viewer button.
- BUG ID 18342 - SplineWarp: The spline keyframe + button does not add keys to the Curve Editor or Dope Sheet.
- BUG ID 18405 - SplineWarp: Smoothing or cusping a source curve also affects the destination curve if both are set to **visible**.
- BUG ID 18497 - SplineWarp: Right-clicking on a single control point in multiple selections only affects the single point, which is not consistent with RotoPaint.
- BUG ID 18709 - GridWarp: Rotating the transform jack with both grids selected, but in different positions, does not undo as expected.
- BUG ID 18712 - GridWarp: The timeline currently shows both source and destination keyframes, even if a grid is not visible in the Viewer.
- BUG ID 18949 - GridWarp: The Viewer does not take downstream warps into account, only showing the first warp function.
- BUG ID 19012 - SplineWarp: It is currently difficult to drag control points when both the source and destination points are in the same position.
- BUG ID 19079 - SplineWarp: The **C** hotkey does not currently select the **Add Correspondence Point** tool.
- BUG ID 19148 - SplineWarp: Rendering fails if a curve is reduced to a single point.
- BUG ID 19386 - GridWarp: All keyframes are removed from the Dope Sheet after undoing, rather than just the latest, and the keyframes remain on the Timeline and in the Properties pane.
- BUG ID 19565 - GridWarp: The Viewer does not show the correct mix amount when in **morph** mode with **mix** set at an intermediate value.
- BUG ID 19755 - SplineWarp: Placing correspondence points at each end of an open spline causes rendering problems.
- BUG ID 19783 - SplineWarp: Clicking on pins in the Viewer does not select them in the Properties curves list.
- BUG ID 19835 - SplineWarp/GridWarp: The Viewer LUT is incorrect in **morph** mode, when **mix** is set at an intermediate value.

- BUG ID 19995 - GridWarp: Locked source and destination grids still allow adding and removing grid lines.
- BUG ID 20000 - GridWarp: When using cropped input, moving control points causes the Viewer to display the cropped image and the black area between the crop box and the format when merged over a background.

Other Known Issues

Mac OS X version

- On Mac OS X 10.5 (Leopard), when the Viewer is set to the **OpenGL stereo** stereo display mode, Nuke may trigger an OS X bug that causes a kernel failure. This is due to a bug in OS X 10.5 to do with stereo OpenGL support. For this reason, we do not recommend using the **OpenGL stereo** viewing mode in Nuke on Leopard at this time. The bug has been registered with Apple as bug number 5897735.
- BUG ID 11776 - Node text appears aliased, unclear, or garbled at certain zoom levels.
- BUG ID 12048 - Nuke crashes if you activate screen sharing when there is no screen plugged in.
- BUG ID 13638 - The following graphics cards are not currently supported for Mac:
  - ATI Radeon X1600
  - ATI Radeon X1900

  Users with these cards will be able to render from the command line, but in GUI sessions, the Node Graph will render incorrectly due to a requirement of OpenGL 2 drivers.

Render codecs

There have been issues where rendering with certain codecs makes Nuke crash. Due to this, we recommend the following:

- If you're using the Sorensen Video codec, it's recommended you use the Sorensen Video 3 codec instead. If you're unable to switch to Sorensen Video 3, try using a format smaller than 2K for better performance.
- If you're experiencing crashes with Cineform HD, try updating your Cineform codec to version 5 or above. You may need to download the Neoplayer at http://estore.cineform.com/neoplayer.aspx.
- If you're using Avid Meridien, you should only write out in NTSC and PAL.

FrameCycler

- BUG ID 15204 - FrameCycler: There are currently some graphical glitches occurring when flipbooking certain images. The workaround is to change the cache precision in FrameCycler. To do this, bring up the settings menu in FrameCycler (press **S**), go to **Options** > **Cache** and change the **Cache precision** option to either **16 bit float** or **8 bit int + 16 bit float**.

- Flipbooking with FrameCycler 2009 may not work on some older Intel and AMD processors with certain graphics card configurations. You can set the FC_PATH environment variable to point to a previous version of FrameCycler (that is FrameCycler 2008 found in Nuke 6.0v7). For information on setting environment variables, please refer to the *Configuring Nuke* chapter in the Nuke User Guide.

  For example, you can point FC_PATH to the following locations (check the correct path to the installation on your machine):

  - **Mac**:

    /Applications/Nuke6.0v7-32/Nuke6.0v7.app/Contents/MacOS/FrameCyclerOSX/bin/FrameCycler
  - **Windows**:

    Program Files\Nuke6.0v7\FrameCycler\Windows\bin\FrameCycler.exe
  - **Linux**:

    /usr/local/Nuke6.0v7-32/FrameCyclerCentOS4.4/bin/framecycler

Note *If you do not have Nuke 6.0v7 or earlier installed, you can download it from The Foundry's Web Archive at the following address:*

http://www.thefoundry.co.uk/products/nuke/product-downloads/

- We direct FrameCycler to write to the user's Nuke temp directory (NUKE_TEMP_DIR) for its user settings files. You can redirect this by modifying the FrameCycler/settings/Global_Settings.xml file that can be found within your Nuke installation.
- BUG ID 15046 - Flipbooking to FrameCycler with images greater than 4K in width, and with a non-1 pixel aspect ratio, skews the images (in FrameCycler). This is an issue with FrameCycler 2009, and we are awaiting a fix.
- BUG ID 17199 - There aren't appropriate LUTs in 6.2v3 for flipbooking some colorspaces including rec709, Gamma 1.8, Gamma 2.2, Panalog, REDLog, ViperLog and REDSpace. As a workaround, you need to select **Burn in the LUT** in FrameCycler for an authentic color reproduction.

RotoPaint

- The foreground onion skin overlay updates as you paint. This will change so the overlay only updates with the new stroke on pen up.
- It is not currently possible to clone RotoPaint nodes.
- Interactivity of laying down strokes/shapes in the Viewer may be faster when motion blur is disabled on the layer you are working in.

- BUG ID 9238 - Painting on Mac OS X and Linux is slower when the paint cursor is near the edges of the screen.
- BUG ID 11524 - Adding strokes/shapes in RotoPaint is slow when there is another RotoPaint after it.
- BUG ID 20195 - Using the new No animation on all knobs on a Roto(Paint) node, displays a PlanarTracker related error.

  You can click OK on the message to remove it, but sometimes multiple instances need acknowledgement.

Python

- Nuke sometimes reports errors in Groups and Gizmos, appearing similar to the following:

  groupName.NodeName.knobname: unexpected 'k' in '0.knobname'

  The problem is most likely that there is an expression that is using the input TCL command and doesn't validate that there is an input connected. An example expression:

  [input parent 0].translate.x

  The input command returns 0 when it can't find the requested input, which generates an expression of '0.knobname' that doesn't refer to anything. The fix is to restructure the expression to use the value TCL command and specify a default value to return in the case that the expression is invalid. It takes the form:

  [value [input parent inputnumber].knob defaultValue]

  Here is the modified example:

  [value [input parent 0].translate.x 0]

  The modified example will return 0 in the event that there is no input 0, and no longer result in an error message.
- There is a Python syntax conflict when assigning knob names on the fly with nuke.nodes.<node>() if the knob is called 'in'.

  For example, this will give a syntax error:

  nuke.nodes.Shuffle(in = 'depth')

  while this works because 'in' is a string here and not a keyword:

  sh = nuke.nodes.Shuffle()

  sh['in'].setValue('depth')
- BUG ID 6455 - You should not call the Python command nuke.restoreWindowLayout() from the Script Editor as that can cause Nuke to crash. Instead, you can use the same command from your menu.py, restore layouts by selecting **Layout** > **Restore Layout**, or use a custom menu or toolbar item.
- BUG ID 8063 - Creating many new nodes with nuke.createNode() and the in-panel argument at default (True) may crash when too many node

control panels are created too quickly. The workaround is to pass the in-panel argument as False or else use **nuke.nodes.NodeClass()** (where NodeClass is the type of node to create) to create the node and then connect it to the currently selected node manually.

Miscellaneous known issues

- File types in Windows and Mac OS X are associated with the standard version of Nuke by default, so if you save a script on NukeX using features that are only included in NukeX (such as CameraTracker or FurnaceCore) and then double-click on the script icon to open it, it will open in standard Nuke instead of NukeX.
- If you have trouble with FBX files, it may be because they were written with an older version of FBX. If they load very slowly, it is also possible that they are ASCII rather than binary. To get around these problems, you can use the FBX converter on the Autodesk web site. It converts between various different formats, including older FBX versions, ASCII, and binary, and is available on Windows, Mac OS X, and Linux.

  To download the FBX converter:

  1. Go to http://usa.autodesk.com/adsk/servlet/pc/item?siteID=123112&id=10775855 (or click here)

  2. Scroll down to **FBX Converter** and click on one of the links to start the download.
- CameraTracker: Canceling lens distortion initialization results in corrupt tracks.
- 3D Camera: If you want to navigate through the 3D point cloud created by the CameraTracker node when using **Create Scene**:
  - Select the Camera that was created in the 3D view when using the **Create Scene** button.
  - Press **F** to focus on the selected Camera. You can now navigate around the cloud. Do not try to focus (using **F**) on the point cloud. The resulting tumble camera movement is likely to be jumpy.
- PrmanRender: In the Nuke camera, the **window roll** control is not yet mapped to RenderMan.
- BUG ID 5083 - Flipbooking the output of the Anaglyph node asks which view you want to render. This question is unnecessary as the result is an anaglyph image. Irrespective of what view you choose, the flipbook output is the same.
- BUG ID 5690 - Windows run-time libraries are not packaged properly with Nuke.

  Nuke runs correctly from a network install on Windows without specifically installing the run-time libraries, though we still recommend that you do so as there will still be some minor problems without them. For details, please see *Installation on Windows* in the *Installation and Licensing* chapter of the Nuke User Guide.

- BUG ID 5922 - At the moment, cloning does not work properly with all OFX nodes. This affects, but is not restricted to, any nodes that have an analysis pass.
- BUG ID 9521 - Currently, the Nuke Viewer cannot cache very large plate sequences in float. The limit per frame is 50MB. If your frames are larger than this, you may need to switch to proxy mode for the caching to work.
- BUG ID 11620 - In the 3D Viewer, there is currently a conflict between 3D geometry selection and points drawn with RotoPaint. This only occurs if you have two Viewers open, one in 2D mode and the other in 3D mode, and you have the panel for the RotoPaint visible.
- BUG ID 12424 - Ultimatte: Overlays are not updating correctly or reverting when panning or zooming.
- BUG ID 12505 - Motion Vector output has been improved, but still doesn't work properly because some large polygons are clipped by the front camera plane.

  You can minimize this effect by increasing the geometry **tessellation max** parameter.
- BUG ID 17550 - The 3D Viewer does not always honor the **display** setting for geometry when you click on the Node Graph.
- BUG ID 18030 - PrmanRender: Rendering OBJs and FBXs without texture coordinates causes Nuke to crash.
- BUG ID 18271 - The Error Console occasionally hangs or crashes with certain node errors and has to be forcibly closed.
- BUG ID 18649 - The transform jack is currently scaling incorrectly from the corner pivot point.
- BUG ID 19162 - Setting a Read node to **cache locally** > **always** occasionally causes Nuke to crash when performing frame analysis, such as when you're using the CameraTracker node to track a moving camera.
- BUG ID 19185 - Attaching an FBX ReadGeo to the Viewer occasionally causes a slight graphical glitch in the Properties pane.
- BUG ID 19933 - ReadGeo: Geometry occasionally doesn't display as a solid until you click in the Viewer.
- BUG ID 20059 - Displacement: When applying displacement to scaled up geometry, tears in the surface of the geometry may appear from certain camera angles, leaving gaps in the rendered 2D output.

  To avoid this problem, either use geometry of the right size or render out a prescaled version of the required geometry.
- BUG ID 20081 - Toolsets: Local Toolsets currently take priority over built-in and facility-wide Toolsets when loading.

- BUG ID 20204 - Multitexturing: When **Preferences** > **Viewers** > **Multiframe** is enabled, increasing **downrez** in the Viewer toolbar can cause textures to flicker in the 3D Viewer.

  You can switch back to **Classic** mode or avoid using proxy in 3D to work-around this issue.

6.3v3

Version

Nuke 6.3v3

Release Date

21 September 2011

Supported Operating Systems

- Mac OS X 10.5 "Leopard" and 10.6 "Snow Leopard" (64-bit)
- Windows XP64, Windows 7 64-bit
- Linux RHEL 5.4 64-bit

Note *Mac OS X 10.7 "Lion" is currently not supported as an operating system for Nuke.*

New Features

Nuke has been updated to use RED SDK reader version 4.1, adding support for the Epic Camera and HDRx files. This change adds a **RedColor2** option to the **decode colorspace** control and a **RedGamma2** option to the **gamma curve** control on the Read node properties panel. HDR decoding on RED Rocket cards is also now enabled.

Note *When you're working with HDRx files, it's worth noting that Nuke currently only decodes the individual HDRx frames, but does not automatically mix them so you may need to mix your frames manually.*

For more information on new features in Nuke 6.3v1, see "6.3v1" on page 17.

Feature Enhancements

- BUG ID 4337 - Read: When reading in stereo EXR's, Nuke now gives the option to automatically set up the required stereo views.
- BUG ID 5629 - PYC files are now generated for Python scripts during Nuke installation.
- BUG ID 8256 - DPX colorspace and transfer characteristic header information is now written in the file metadata.

- BUG ID 13563 - A new **offset negative display window** checkbox was added in the Read node for dealing with EXR files with negative display values.
- BUG ID 13649 - Python: You can now split string knobs and set values for them using Python.
- BUG ID 15133 - Python: New functions were added to Nuke's Python Menu class (for example **setScript**, **setIcon**, **setShortcut**).
- BUG ID 16579 - Chroma sub-sampling options were added for writing JPEG files.
- BUG ID 16816 - A **flip interlaced stereo views** checkbox was added on the **Viewers** tab in the Preferences for flipping interlaced stereo views.
- BUG ID 19430 - In the node rename dialog, you can now press the **Esc** key to exit the editing mode, restore the previous value and close the dialog.
- BUG ID 19467 - A function called **print_callback_info()** has been added to misc.py in nukescripts. It allows you to return information on currently active callbacks.

Documentation

- The Nuke Reference Guide now covers these nodes:
  - **Draw > Noise**
  - **Draw > Radial**
  - **Draw > Ramp**
  - **3D > ProjectionSolver**

Bug Fixes

- BUG ID 6431 - An expression in the file path should have been maintained as long as the path continued to match the file.
- BUG ID 8068 - Nuke could not open or execute a script from the command line when the file path was longer than 199 characters.
- BUG ID 10351 - It wasn't possible to set an expression for both views using Python.
- BUG ID 12767 - Flipping an image with the Transform node produced a different result from previous versions of Nuke.
- BUG ID 13377 - Mac OS X: Tablet errors occurred when drawing a paint stroke that continued outside the tablet surface.
- BUG ID 16715 - Python: The help for **Pulldown_Knob** has been updated.
- BUG ID 16857 - The Precomp node wasn't looking for Write nodes inside a group.

- BUG ID 17317 - A customer script with a GeoOp connected to a Viewer would crash Nuke.
- BUG ID 17737 - The AddTimeCode node didn't permit setting an expression for the **start code** control.
- BUG ID 17849 - Animation in **Local Matrix** wasn't showing keyframes on the timeline.
- BUG ID 18098 - Particles: The bounding box for particles rendered through ScanlineRender was slightly too small causing their colors to streak across the format.
- BUG ID 18277 - Stretched pixels occurred in a customer script with a ScanlineRender node and a ReadGeo node.
- BUG ID 18416 - Setting up stereo views after creating a node set the **views** control to **none**.
- BUG ID 18492 - Python: A segmentation fault occurred when using **nuke.executeMultiple** with **views** set to **none**.
- BUG ID 18583 - Python: The **File_Knob** had no **splitView** and **unsplitView** methods.
- BUG ID 18584 - Viewer: The **apply LUT to color channels only** control failed when used with a customized Viewer process.
- BUG ID 18811 - SplineWarp: Incorrect rendering occurred with a two-point curve when the BBox Boundary Curve was disabled.
- BUG ID 19059 - DPX: The transfer field in metadata was set according to what the Write node's colorspace was by default.
- BUG ID 19092 - Moving control points or handles in the 3D Viewer mode when a GridWarp or a SplineWarp node was attached to a Card created flashes in the Viewer.
- BUG ID 19198 - Transforming objects in the Viewer was showing a zoomed in bounding box.
- BUG ID 19241 - Nuke crashed when starting up on Mac OS X 10.7 (Lion).
- BUG ID 19790 - Local file caching: The Node Graph indicated that a file was being cached after its local file caching options were set to **never**.
- BUG ID 19858 - Tracker: Clicking **Cancel** in the tracking process dialog grayed out the button and didn't stop the tracking.
- BUG ID 20058 - The file browser had a 5-7 second delay when loading file contents from a network location.
- BUG ID 20181 - Particles: There was no way to cancel particle calculations, which sometimes caused Nuke to become unresponsive.
- BUG ID 20229 - GridWarp: The destination grid didn't resize automatically.
- BUG ID 20254 - RotoPaint: Painting over existing paint strokes caused glitches in the Viewer.

- BUG ID 20381 - GridWarp: It was very slow to drag a large grid, undo a drag, and move keyframes on the Dope Sheet.
- BUG ID 20461 - GridWarp: The **submesh resolution** control wasn’t clamped at 1.
- BUG ID 20540 - SplineWarp: After drawing a point and undoing it, the Bezier tool still appeared selected even though the selection mode was on.
- BUG ID 20616 - RotoPaint erroneously displayed the **Frames to be visible on** dialog when removing animation from all controls.
- BUG ID 20645 - Incorrect proxy behavior occurred with a customer script.
- BUG ID 20647 - When you **Ctrl/Cmd**+dragged to rotate a 3D object in the 3D Viewer with a light node, the lines on the light appeared thicker.
- BUG ID 20705 - Writing a stereo EXR from the ScannedGrain node hung Nuke.
- BUG ID 20708 - The CopyMetaData node didn't write DPX header metadata correctly.
- BUG ID 20810 - The 3D Viewer was multi-texturing even though the **classic** mode was selected on the **texture mode** dropdown in the Preferences.
- BUG ID 20992 - Nuke crashed using Mac OS X 10.7 (Lion) and an ATI graphics card.
- BUG ID 21013 - RotoPaint: After painting a clone brush stroke, the onion skin overlay disappeared.
- BUG ID 21038 - Setting the **vertex emission rate** to 0 on the ParticleEmitter node caused Nuke to hang.
- BUG ID 21054 - The DeepFromImage node added black samples when the alpha value was zero.
- BUG ID 21121 - PyQt panels only expanded halfway when they were docked.
- BUG ID 21155 - The ScanlineRender node caused 3D geometry objects connected to a Scene node with a Camera to have the wrong projection.

6.3v2

Release Date

29 July 2011

New Features

There are no new features in this release.

Feature Enhancements

Documentation

- *The Script Editor and Python* chapter has been removed from the Nuke User Guide and all information in it has been included in the improved Python documentation. For a closer look at the Python documentation, click **Help** > **Documentation** in Nuke.
- The Nuke Reference Guide now covers these nodes:
  - **Time** > **AppendClip**
  - **Time** > **FrameBlend**
  - **Time** > **Retime**
  - **Time** > **TimeWarp**
  - **Transform** > **Crop**
  - **Transform** > **LensDistortion**
  - **Transform** > **Reformat**
  - **Transform** > **Tile**
  - **Transform** > **Transform**
  - **3D** > **CameraTracker**
  - **3D** > **Modify** > **DisplaceGeo**
  - **3D** > **Modify** > **Normals**
  - **3D** > **Shader** > **BasicMaterial**
  - **3D** > **Shader** > **Displacement**
  - **3D** > **Shader** > **Phong**

Bug Fixes

- BUG ID 14707 - Linux and Mac OS X: Starting Nuke using a foreign language operating system caused Nuke to crash on start-up with a **Bad Format** error.

- BUG ID 20242 - Scripts containing animated geometry, including particles, did not correctly report memory usage, resulting in slow rendering and playback after a few frames.

6.3v1

Release Date

18 July 2011

New Features

Deep Compositing Nodes

The new Deep Compositing nodes in Nuke enable processing of deep pixel images. These are images that have more than one color sample at each pixel. The concept of deep pixel images differs from the concept of channels and in fact, each channel can have multiple samples. There is a new Deep menu in the toolbar, and a new deep sampling widget in the bottom of the Viewer window for showing the samples under the mouse cursor as you move it across a deep pixel image. For more information, see p. 509 in the Nuke User Guide.

Particle System (NukeX)

The new Particle node set is a solution for creating particles in a 3D environment. The Particle nodes are grouped under the Particles icon in the Nuke toolbar, and you can use them for emitting, manipulating and displaying limitless types of particles in your 3D scene. The essential node to start with is the ParticleEmitter node. To render in 2D view, you need to connect the output of your particle system to a ScanlineRender node. For more information, see p. 695 in the Nuke User Guide.

AudioRead Node

You can use Nuke's AudioRead node (**Other** > **AudioRead**) to read in an audio file to your project, use it to synch visual and auditive cues and then flipbook it with your footage. You can read in uncompressed WAV and AIFF files, and you can display an audio waveform for your audio clip in the Curve Editor and Dope Sheet. Furthermore, you can access its animation curve through expressions. For more information, see p. 390 in the Nuke User Guide.

New SplineWarp

The existing Spline Warp node (**Transform** > **SplineWarp**) has had a major makeover. The new SplineWarp shares a foundation with RotoPaint so that splines can be copied between nodes. For more information, see p. 393 in the Nuke User Guide.

New GridWarp

The existing GridWarp node (**Transform** > **GridWarp**) has been rewritten. Grid control point positions can now be expression-linked and edited in the Curve Editor and Dope Sheet and the grid resolution can be dialed "virtually" with new subdivisions previewed and created when the slider is released. For more information, see p. 393 in the Nuke User Guide.

Displacement Shader Node

The new Displacement shader node (**3D** > **Shader** > **Displacement**) creates geometry on the fly in the ScanlineRender node and has a number of different tessellation modes which you can switch according to the type of displacement you're creating. This displacement is multi-threaded, and in most cases should be faster to process for the equivalent amount of detail than the DisplaceGeo node. For more information, see p. 480 in the Nuke User Guide.

Local File Caching

Nuke's local file caching feature is a new way of working faster by copying files used in a script to local disk for faster reading and reduced demands on your network. There are new **Local File caching** preferences for your cache folders, new options are available in the Read node control panel, and there's a new top level **Caching** menu. For more information, see p. 133 in the Nuke User Guide.

LGPL Qt 4.6.2

This release of Nuke links dynamically with an LGPL build of Qt 4.6.2. This allows developers to link plug-ins to Nuke's Qt, and create custom widgets that operate smoothly within Nuke.

Node Presets

This is the ability to create node presets that can be selected from the new presets menu on the control panel of a node. This allows you to create time-saving presets of the node settings that you use often.

PlanarTracker (NukeX)

Using the new PlanarTracker (**Transform > PlanarTracker**) node, you can now track areas in your image that lie on a plane. Planar tracking derives a transform based on matching a large image area's position from one frame to the next, as opposed to small features relative to a reference pattern as used in point tracking. As a result it's less susceptible to jitter and inaccuracy due to grain, motion blur, and camera shake. PlanarTracker is available as both a stand-alone node and integrated into Roto. For more information, see p. 714 in the Nuke User Guide.

ToolSets

A new ToolSets menu in the toolbar provides easy access to store and recall groups of nodes. Like node presets, these are stored in your NUKE_PATH so they can be added locally or site-wide. For more information, see p. 58 in the Nuke User Guide.

Denoise (NukeX)

A new noise and grain removal node (NoiseTools > Denoise) is now available in NukeX for analyzing the noise in your input image. You can use it to calculate a noise profile and remove noise from digital or film footage. For more information, see p. 689 in the Nuke User Guide.

Caching

Nuke 6.3 is much more aggressive with memory caching, without going over the Nuke image buffer memory limit. In previous versions of Nuke the image memory cache on a node would be cleared when you tweaked a parameter, and all caches would be cleared when you flipped frames. In this version the memory caches are kept, meaning undo, and tweaking parameters on different frames is much faster. When Nuke reaches the memory limit the caches are dropped until Nuke is under the image buffer memory limit set in the Preferences. Nuke will also clear all the memory caches if you select Cache > Clear Buffers or press F12.

Because this changes the memory profile significantly, there's a new option in the Preferences to explicitly enable this behavior. The option, off by default, should only be enabled on machines with 8GB's of RAM or more. On machines with less memory, enabling it can cause extreme slowdowns, unless the memory use limit is set to 20 - 30%. Even then, it can cause problems if you have multiple instances of Nuke running, or you're using background rendering.

New LUTs and Colorspace Conversions

Sony S-Log, Alexa LogC (V3), and a Josh Pines-style pivoted LogLin have been added to the root LUTs and to the Colorspace node. A standalone node for the pivoted log-lin conversion is also available (Color > PLogLin).

Toe

A new Toe node has been added to the Color menu. Toe lifts the black levels, in a similar way to gain controls, but with a rolloff so that whites are mostly not affected. See p. 207 in the Nuke User Guide for more information.

Documentation

New documentation is available with Nuke 6.3 on the **Help** > **Documentation** menu.

- **Nuke Reference Guide**—a reference for all controls within each node in Nuke, though currently, some nodes only include a brief description. With later versions, we aim to provide a complete list of controls for all nodes.
- **Python Developers Guide**—a developer's guide to using Python in Nuke.
- **Nuke Developers Guide**—a developer's guide to the NDK.

Feature Enhancements

- The SplineWarp node's algorithm has been enhanced to enable more localized warping results. There's also a new **Classic warping** box in the control panel that you can check to use the old, more widespread warping method.
- BUG ID 4403 - CameraShake's **cs-center** indicator now defaults to the center of the supplied input. Motion blur is now calculated from the Transform node (rather than VectorBlur and TimeBlur), improving smoothness and render times.
- BUG ID 5488 - You can now pick a static frame to generate the postage stamp from. This is a per-node setting on the **Node** tab and there's a global preference in **Preferences** > **Node Graph** > **postage stamp mode** to enable or disable it globally.
- BUG ID 5827/17036 - CornerPin now has controls to copy **to** values to **from** and vice-versa. You can also reset **from** to the input format or bbox.
- BUG ID 6357 - The **-sro** flag was added for command line rendering to force the render order set by Write nodes to be honored.
- BUG ID 7797 - The Flipbook dialog now supports multiple ranges.
- BUG ID 8582/15162/16944 - S-Log, LogC, and a Josh Pines-style LogLin LUTs have been added. See New LUTs and Colorspace Conversions above.
- BUG ID 9267 - A new preference **show transform preview** has been added to the **Viewers** tab. You can disable this to prevent previewing in the Viewer when using nodes like Transform or CornerPin.
- BUG ID 9936 - Nuke now reports to the console if you request a frame during render that you failed to notify Nuke of using **getFramesNeeded**.
- BUG ID 11925 - In the Node Graph when using the **Tab** menu to select nodes, if only one option is available, you can use **Return** to select it.

- BUG ID 15222 - The Viewer right-click menu **Stereo Modes** now contains checkboxes to display the current stereo mode.
- BUG ID 15828 - An **assume zero padding on files** option has been added to the **Preferences** > **Appearance** tab to intelligently apply frame substitutions (# or %d) to sequences.
- BUG ID 15864 - A **disable GPU dithering** preference has been added in the Viewers tab for removing dithering when using a half float depth in the Viewer.
- BUG ID 15997 - Windows only: Multiple instances of Nuke now all use the same terminal window.
- BUG ID 17224 - You can now customize the display colors in the Dope Sheet using the **Preferences** > **Appearance** tab.
- BUG ID 18001 - Incoming EXRs now have a metadata flag detailing whether or not they are tiled.
- BUG ID 18759 - A 4x4 matrix was added to the CornerPin node.
- BUG ID 19302 - We've added a new right-click option on nodes in the **Properties** tab called **No animation on all knobs** to remove all animation from a node.

Bug Fixes

- BUG ID 3413 - File Browser completion was case sensitive.
- BUG ID 6612 - Transform overlays only appeared in one view in stereo viewing modes.
- BUG ID 11355 - Creating a node and a control from init.py caused Nuke to crash.
- BUG ID 12452 - Scripts containing layers with channels called **none** were causing Nuke to crash.
- BUG ID 13851 - Postage Stamp updates were occurring when scrubbing in the Viewer.
- BUG ID 14455 - Flipbooking didn't work properly if no nodes were selected.
- BUG ID 14774 - A **Nothing is named "parent"** error occurred when expression linking in RotoPaint.
- BUG ID 15231 - Flipbooking in the background was not working correctly.
- BUG ID 15362 - Rendering four Card nodes one in front of another with the ScanlineRender node had errors even though rendering separately and merging worked well.
- BUG ID 15615 - Text in the Viewer was unreadable when using the **Preferences** > **Presets** > **Silver** color scheme.
- BUG ID 15649 - Mac OS X only: Nuke dominated the screen when switching Applications.

- BUG ID 16042 - OpenGL Stereo was not disabled when switching into the 3D Viewer.
- BUG ID 16101 - The camera did not appear in the Viewer when you connected it to a Project3D node.
- BUG ID 16305 - Write nodes were cropping footage when used mid-tree.
- BUG ID 16414 - Keylight: Scaling a Transform node when using Keylight caused a **Black in root got request for zero or negative-sized box** error.
- BUG ID 16513 - Write: Nuke looked for both views even when the **-view** parameter was set to just one in the command line.
- BUG ID 16545 - Write nodes were occasionally generating a popup dialog containing **could not find knob** errors.
- BUG ID 16673 - The UnmultColor node **bg color** control had an inconsistent default value.
- BUG ID 16728 - Python: Extra knobs were changed when the Read node properties panel was opened.
- BUG ID 16736 - Running from the command line in NonGUI mode and using **nodeCopy** caused Nuke to crash.
- BUG ID 16761 - Copying nodes in large, continually splitting-and-merging scripts was excessively slow.
- BUG ID 16819 - Rendering simple scripts containing a Defocus node was causing Nuke to crash.
- BUG ID 16830 - RotoPaint: A crash occurred with the Smear tool when the wacom pen pressure was set to affect size.
- BUG ID 16864 - Python: Using **nuke.env** dictionary **.get** method crashed if you used it twice.
- BUG ID 16875 - Evaluating an **EvalString_Knob** when it was not attached to a node caused Nuke to crash.
- BUG ID 16887 - A discreet LUT was evaluating incorrectly with the ColorLookup node.
- BUG ID 16968 - ScannedGrain did not comply with Nuke's way of passing requested channels upstream.
- BUG ID 17074 - Camera: Keyframe values were truncated at 6 digits when exporting channels.
- BUG ID 17086 - The ScannedGrain node could only render one view at a time.
- BUG ID 17092 - The right-click menu in the Node Graph didn't synchronize properly with the menu bar when disabling an item.
- BUG ID 17169 - RotoPaint: Duplicating a shape from the Viewer omitted a letter from the shape name each time.
- BUG ID 17194 - Re-running all or part of a certain script was causing Nuke to crash.

- BUG ID 17230 - Write nodes were channel shuffling in certain cases.
- BUG ID 17237 - RotoPaint: A crash occurred when selecting strokes in a customer script.
- BUG ID 17267 - RotoPaint: Nuke hung when deleting strokes or shapes when there were many strokes.
- BUG ID 17376 - Executing **nuke.makeGroup** from the Script Editor didn't work.
- BUG ID 17387 - Dope Sheet: Using **Edit** > **Generate...** with a Read node open in the **Properties** tab caused Nuke to crash.
- BUG ID 17401 - Write: NukeQuickTimeHelper-32 processes were not removed when the associated Write node was deleted.
- BUG ID 17412 - **Edit** > **Generate...** was generating keys for all nodes, not just the selected node in the Dope Sheet and Curve Editor.
- BUG ID 17418 - The Plug-in Installer was not launched by the installer or **Help** > **Plug-in Installer** menu option.
- BUG ID 17450 - Python: A crash occurred with a segmentation fault when exposing knob in the Precomp node and then reloading the script.
- BUG ID 17483 - Setting Viewer **downrez** >1 with **show overscan** enabled caused artifacts in the Viewer.
- BUG ID 17484 - Filtering didn't work after opening a script in the File Browser.
- BUG ID 17491 - ColorLookup: Channels were shuffled when **use precomputed table** was disabled.
- BUG ID 17500 - Labels were missing in the **RotoPaint.curves** expression dialog.
- BUG ID 17507 - Using DisplaceGeo in certain scripts caused Nuke to crash.
- BUG ID 17526 - FBX specific controls were visible on a ReadGeo node when an FBX file was read in using the **R** shortcut.
- BUG ID 17529 - Write nodes did not accept format specific parameters in Python render mode.
- BUG ID 17654 - In certain scripts, connecting a PositionToPoints node to a Viewer caused Nuke to crash.
- BUG ID 17845 - The ModifyMetaData node did not refresh after undo.
- BUG ID 17924 - CurveTool: Settings **Curve Type** > **Max Luma Pixel** mode without any input caused Nuke to crash.
- BUG ID 17940 - Editing in the ModifyMetaData node wasn't working as expected.
- BUG ID 17947 - Expression linking tracking values confused the track points and did not affect the stroke/shape point position.

- BUG ID 17993 - A customer gizmo containing a linked control was causing the Curve Editor to behave strangely or disappear entirely after saving and reopening the script.
- BUG ID 17997 - Windows only: Removing a keyframe in the Dope Sheet displayed random values and caused Nuke to crash.
- BUG ID 18064 - Clicking **File** > **Clear** reset custom LUTs to linear.
- BUG ID 18089 - LevelSet: Certain control value and input combinations were causing Nuke to crash.
- BUG ID 18093 - HistEQ: When used in conjunction with certain other nodes, bounding box x/y values were not taken into account when calculating proportions, causing Nuke to crash.
- BUG ID 18121 - Changing the **channels** control in a Blur node didn't update the Viewer.
- BUG ID 18177 - Write nodes were displaying the render time in the terminal when rendering in the interface.
- BUG ID 18237 - Windows only: Every Write node producing MOV output had its own active NukeQuickTimeHelper-32.exe process running.
- BUG ID 18398 - ColorLookup: Deleting a channel (delCurve) and adding it with an expression (addCurve) resulted in the default value.
- BUG ID 18415 - Keylight: Enabling stereo mode and scrubbing the **Screen Balance** control caused Nuke to crash.
- BUG ID 18441 - Reading in FBX files through a Camera node was unreliable.
- BUG ID 18444 - Roto: **extra matrix** expression linking between different Roto nodes didn't work as expected.
- BUG ID 18511 - Certain expressions, such as **input.width** and **exists parent.input**, were causing a customer script to crash or hang on render.
- BUG ID 18549 - Create, undo, create operations in certain scripts were causing Nuke to crash.
- BUG ID 18554 - When loading a certain script, the Viewer was not drawing as expected when viewing a ColorCorrect or Merge node downstream of FrameHold nodes.
- BUG ID 18609 - Camera: Using right-click **set to default** on split Camera controls caused Nuke to hang.
- BUG ID 18673 - Project3D **front** mode showed black instead of the color in the 3D view.
- BUG ID 18747 - RotoPaint: No output was displayed when **output** was set to anything other than **rgba**.
- BUG ID 18818 - Keylight: Scrubbing in the Viewer when viewing a Keylight node caused Nuke to crash.

- BUG ID 18821 - Calling a Python **node.metadata()** function from an expression on an OFX node caused Nuke to crash.
- BUG ID 18831 - The Nuke user interface was not obscuring RotoPaint operations. For example, clicking a Viewer tool created a new stroke instead of selecting the tool.
- BUG ID 18928 - Python: Nuke froze when using **nuke.selectConnectedNodes()**.
- BUG ID 18932 - ScanlineRender: Diagonal lines were appearing in the 2D Viewer when projecting onto a surface.
- BUG ID 18946 - Dragging an animation curve control from RotoPaint's **Transform** tab to a control point crashed Nuke.
- BUG ID 19019 - Write: Duplicate controls were added when **read file** was enabled using the EXR format.
- BUG ID 19068 - PrmanRender: Setting the **projection mode** to **orthographic** caused Nuke to crash.
- BUG ID 19176 - DiskCache: A crash occurred when you precached a simple script with a transform.
- BUG ID 19178 - The orthographic **projection mode** produced unexpected results on ScanlineRender.
- BUG ID 19421 - When rendering from the command line, a customer script was reporting incorrectly that **There are no Write operators in this script** when a Write node was present.
- BUG ID 19510 - Viewer: Overscan did not retain the pixel aspect ratio.
- BUG ID 19521 - Camera: When **projection** was set to **orthographic**, the **depth** channel was incorrect.
- BUG ID 19665 - There was no notification when background renders failed.
- BUG ID 19892 - Transform controls in certain nodes, such as Camera or Axis, were not disabled when **read from file** was enabled.
- BUG ID 19895 - ReadGeo: The **read from file** control was disabling controls it shouldn't affect.

Developer Notes

Here are the changes relevant to developers.

Changes for Nuke 6.3v1

- Everything in the NDK is now in the **DD::Image namespace**. Assuming that developers have **namespace DD::Image** at the top of their source files, then this shouldn't affect them. If they have NDK objects declared/ referenced in header files, they will most likely need to properly scope them with **DD::Image::** (for Nuke 6.3 and onwards). The changes that will most likely affect NDK developers are:
  - Everything in DDImage/Thread.h (Guard, Lock, etc)
  - Everything in DDImage/gl.h (glColorf, gl_text, etc)
  - Everything in DDImage/MetaData.h, which means that any classes that derive from **DD::Image::Op** and **override _fetchMetaData()** will have to scope the return value of the function

- We've added the following new knobs/methods to create them:
  - **TransformJack_KnobI**, created using **TransformJack_knob()**, which allows for creation/manipulation of a Transform Jack in the Viewer.
  - **Cached_File_knob**, which is the same as a regular file knob, except that it automatically has an option to cache the file locally.
  - **Read_File_knob** which is for Read file knobs. This takes in a **Read-FileKnobOwner** class, which developers can derive from to provide their own file read associated functionality.
  - **Write_File_knob** which has two strings for a file, one for the full res and one for the proxy.
  - **PositionVector_knob**: This takes in 6 floats, 3 for the start xyz position, 3 for the end xyz position, and creates the appropriate handles in the 3d viewer to manipulate those positions. This knob is used by the particle gravity op.
  - **ControlPointCollection_knob**: This knob is used extensively by the GridWarp. It allows for collections of control points and tangents to be manipulated.

- Geometry selection has been refactored. It is not just a typedef now, but a class with methods. Details on how to use it are available in the NDK Developer's Guide.
- A host of interfaces have been added for dealing with Deep data.
- Python on Windows now includes the **ctypes** module.
- All leaked symbols to ilm libraries related to OpenEXR have been removed. This means that NDK developers can now easily link to different versions of the OpenEXR libraries.

- Added a reference to **Histogram_Data**, which can be used with the Histogram knob.
- NDK Documentation: **disable()** was not defined in **Iop.h** but it was referenced in the NDK documentation.
- We added a new method (**nuke.ofxAddPluginAliasExclusion**). Calling this with a fully qualified Ofx node name will disable Nuke from creating a short alias. This allows third party developer's to decide if they want Ofx nodes (such as those in FurnaceCore) to be created with calls to **nuke.createNode()** using a short name. For more info, including an example, do help(**nuke.ofxAddPluginAliasExclusion**) in the script editor.
- **nuke.rawArgs** was added. This gives all of the arguments passed to Nuke. This was done so that clients can access all of Nuke's arguments in Python, without introducing new problems for clients that were already using sys.argv.
- The ability was added to specify knob defaults for non-file format specific knobs in Write and Read nodes. So for example, **nuke.knobDefault**("Read.exr.colorspace", "sRGB") will work now.
- When the file format is changed, the default, if it exists, overrides what you had previously selected for that knob.
- Nuke Write always executed even with **validateFilename** python callbacks. A C++ callback has been added to validate filenames and we've added functions (registerFilenameValidateCallback, unregisterFilenameValidateCallback) to DDImage/FileFuncs.h. These allow developers to add C++ filename validation callbacks, to replace the Python validation callbacks. The C++ versions are much more consistent, and work in all cases (unlike the Python filename validation callbacks, which only work when clicking on the **Render** button of a Write node).

Other Change Information

- The Viewer cache class may change in subsequent maintenance releases. Please do not sub-class the Viewer cache class, or you may have to recompile your plugins for subsequent maintenance releases.
- BUG ID 10809 - Python: **sys.argv** doesn't include all **nuke** arguments. We've added **nuke.rawArgs** to allow devs to get at all of the arguments and maintain backwards compatibility.
- BUG ID 11338 - Nuke now links against LGPL Qt 4.6.2 on all platforms and developers can now create their own widgets in Nuke, on the main thread.
- BUG ID 14453 - Python: **nuke.toNode** now works for a **PyScript_Knob** when run from **PythonPanel**.
- BUG ID 15331 - Python: PyQt panels can now be docked in Nuke.

- BUG ID 15792 - NDK: The **SourceGeo::selectable_ boolean** member function has been removed—it was already in the superclass GeoOp.
- BUG ID 16099 - OFX: The creation of unknown OFX parameter types no longer crashes in paramDefine.
- BUG ID 16755 - TCL: Control expressions referencing width/height now automatically get input.width/input.height instead, which is much safer and doesn't cause recursive expressions that cause crashes.
- BUG ID 17052 - Python: Adding a control to multiple nodes through Python now produces an error message and doesn't add the control.
- BUG ID 17075 - Python: **nuke.Undo** objects now allow the text shown in the **edit** menu to be set correctly.
- BUG ID 17257 - NDK: The DDImage headers now (optionally) produce a compiler error when building under Visual Studio versions other than 2005.
- BUG ID 17595 - NDK: Nuke Write always executes even with **validateFilename** callbacks in Python. Devs can now register filename validation callbacks in plugins (in DDImage/FileFuncs.h).
- BUG ID 17652 - NDK: **String_Knob** now respects the **Knob::NO_MULTIVIEW** flag.
- BUG ID 17817 - NDK: **DD::Image::Quaternion** now handles rotations between two vectors pointing in opposite directions.
- BUG ID 17821 - Python: The **_ctypes.pyd** file was missing from the DLLs folder on Windows7.
- BUG ID 17896 - Python: **nuke.Node.forceValidate** has been added, which forces an update and can be used before calling **nuke.Node.opHashes()** for consistency.
- BUG ID 17926 - Various minor errors have been fixed in **flipbookingExample.py**.
- BUG ID 18041 - Python: We've added **nuke.Node.resetKnobsToDefault()** which provides a node method of resetting all knobs to default.
- BUG ID 18094 - Python: **afterFrameRender** callbacks now have the correct context (they were previously called with info for the next frame).
- BUG ID 18272 - Python: General controls on Read/Write nodes can now have control defaults set that are file type specific.
- BUG ID 18399 - Python: A Python method **editCurve()** has been added to LUT objects to enable the editing of curves.
- BUG ID 18513 - NDK: We've added **GeoInfo::setVertexCount** and **Primitive::getPrimitiveType()**.

Note: ***Primitive::getPrimitiveType()*** *is meant for internal Nuke use, and subject to change in future. Do not use it for non-Nuke types.*

- BUG ID 18862 - NDK: A **Histogram_Data** definition has been added, for use with the Histogram knob.
- BUG ID 19782/19787 - Python: We've added a **nuke.getPaneFor()** method to dock a panel next to an existing panel and a **panels.registerWidgetAsPanel** helper function to ease creation of dockable widgets.