RELEASE NOTES FOR KATANA 1.5v1

Release Date

16 August 2013

System Requirements

- Katana 1.5v1 is tested and qualified on Linux 64-bit CentOS/RHEL 5.4
- A graphics card which supports OpenGL shader model 4.0
- A supported renderer (see below)

Supported Renderers

Katana 1.5v1 supports PRMan 17.0 and Arnold 4.0.11.0. The supplied renderer plug-ins are compiled and tested against these versions. Minor version increments of PRMan and Arnold may work, as long as they are API compliant with the supported versions.

To use a version of PRMan or Arnold other than those listed above, you may need to recompile the renderer plug-in.

To expose new features and portions, you may need to modify the renderer plug-in.

Using a version of PRMan or Arnold other than those listed above may produce unexpected behavior. Please note that we can only guarantee to respond to Katana bug reports when they are reproducible with the supplied versions of the renderer plug-in.

PRMan Scene Update

The Katana PRMan17 plug-in adds a new shelf script to update scenes containing PRMan16 PrmanGlobalSettings and PrmanObjectSettings nodes, to be compatible with the PRMan17 renderer plug-in. The update script is available in the PRman17 shelf. To update an existing scene made with the PRMan16 plug-in:

1. Make sure that the PRMan17 plug-in is loaded.

   See the *Installation and Licensing* chapter in the *User Guide* for more on this.
2. Click on the **Shelf Actions** button ⚙.
3. Select the shelf named **(other) PRMan17**
4. Click on UpdatePrmanSettingsNodes.

Third Party Dependencies

Katana 1.5v1 has dependencies on the following third party libraries:

- OpenEXR 1.6.1
- OpenSSL 1.0.0.a

These libraries are provided in the Katana distribution, in separate directories under ${KATANA_ROOT}/bin

An ABI-compatible copy of these libraries needs to reside on your LD_LIBRARY_PATH in order for many of Katana's plug-ins to run. The Katana application itself uses RPATHs to locate the required libraries.

NOTE: Katana's wrapper script ${KATANA_HOME}/katana prepends ${LD_LIBRARY_PATH} to ensure these libraries are visible to Katana plug-ins.

If you manage your own LD_LIBRARY_PATH or wish to expose these libraries to plug-ins by some other means, you can call the Katana binary directly using:
`${KATANA_ROOT}/bin/katanaBin`

Important Changes for 1.5v1

- The connections between input and output ports on ShadingNodes, such as PrmanShadingNode, can now be type checked. This is accomplished by tagging the ports using the assigned shader's **.args** file. An output can provide multiple tags, and all tags associated with an input must be present for an incoming connection to be valid. Examples of this are included with the PRMan co-shader example shaders' **.args** files. The syntax for the **.args** files is as follows:

  ```
  <input name="spec" tags="color|specular"/>
  <output name="out" tags="color"/>
  ```

  Note that multiple tags are separated by a | (pipe) character. Invalid connections are indicated by a red glow around related network material nodes, and a diagnostic error message is sent. Invalid connections are not disconnected, but it's not possible to create invalid connections using the UI.

  There are also two new NodegraphAPI functions in the Port class:

  Port.setTags() and **Port.getTags()**. These new functions allow you to set and get the tags on any node's ports. When the **Port.connect()** function is called from a script or the Python console, an invalid connection is possible, but the target node receives an error signal.

  In the future, the RenderInfo API will allow the renderer plug-in developers to define the shader input/output tags in addition to the current way of specifying them through **.args** files. This allows the renderer plug-ins to directly map the renderer's shader input/output and connection rules.

- BUG ID 34202 - Katana is now statically linked against Boost 1.46.0 for Linux, and the default visibility of all symbols is now hidden.
- BUG ID 34800 - OCIO library upgraded to 1.0.7
- BUG ID 34854 - Calls from C++/C into Python code could result in a deadlock within a render process.

NOTE: Please ensure that any attempts to access the Python C API are scoped with `PyGILState_Ensure()` and `PyGILState_Release()`

- BUG ID 36492 - Katana's PRMan plug-in was compiled against the versioned libprman-17.0.so library as opposed to libprman.so. This prevented the use of different versions of PRMan without re-compiling the plug-in or symlinking files.

  Katana's pre-compiled PRMan plug-in is now compiled against libprman.so to allow for greater flexibility out-of-the-box.
- BUG ID 33868 - There is a new **katana** script in the install directory. This prepends to the LD_LIBRARY_PATH the OpenEXR and OpenSSL library paths that Katana plugins require.

  If you wish to use your own versions of these libraries, then you should adjust the LD_LIBRARY_PATH settings accordingly.
- BUG ID 34130 - Katana previously created more file handles on the license server than required. Now only one RLM license is requested per Katana session. The new behaviour for each launch mode is as follows:
  - **Shell** - 1 x katana_r file handle per Katana session.
  - **Script** - 1 x katana_r file handle per Katana session.
  - **Batch** - 1 x katana_r file handle per Katana session.
  - **Interactive** - 1 x katana_i file handle per Katana session.

NOTE: Katana now shares the Interactive license for local Preview Renders, Live Renders, and Disk Renders.

- BUG ID 34367 - Division of integers was performed differently in the Python interpreter launched with the render process to the interpreter launched with Katana. This caused discrepancies between values computed in the main Katana process, and those computed in the render process.

TIP: It is possible to force old style division on a per module basis using double slash // division.

- BUG ID 35280 - A new environment variable flag has been added (KATANA_LEGACY_ARGS).

  If enabled, Katana searches for args files using the same technique as in Katana version1.1, based on the shader filename instead of the shader name.
- BUG ID 35957 - Temporary Katana scene backup files (crash files) were not named in an identifiable and sufficiently unique manner in all cases.

  The filename now contains all available information from the following list: Scene name, a short hash of the scene asset ID, the process ID of the Katana instance, the show name, and the shot name.
- BUG ID 36038 - The default behaviour for the CopyAndConvert render action (used by the Color render output type) now matches that of the Copy render action (used by the Raw render output type). This forces an overwrite of the render target file, even if the target is read-only (equivalent to **cp -f**).

  An option has been added to the Render Action base class to allow this behaviour to be disabled. Calling **setForceOverwriteTarget(false)** on a render action allows renders to fail if the target is read-only. This would be done by the Render plug-in in the function **configureDiskRenderOutputProcess( )**.
- BUG ID 36526 - Katana now links statically against ZeroMQ. The LD_LIBRARY_PATH modifications for ZeroMQ have been removed from the Katana wrapper script ${KATANA_HOME}/katana.
- BUG ID 37026 - PRMan only: Setting **displayOptions** attributes is now supported for all output types except **none** and **ptc**. Attributes set under that location are added to the display line.

  Example:

  ```
  renderSettings.outputs.<outputname>.rendererSettings.displayOpt
  ions. <optionname>
  ```

  This enables the use of options such as **asrgba**, regardless of the output type. This can be particularly useful for debugging or working around issues such as Bug 35122. See the Known Issues & Workarounds for more information.
- BUG ID 37110 - Katana now ships with the DSOs for ZeroMQ, which were removed from the Katana installation in a previous release. Although Katana statically links to ZeroMQ, customers may wish to compile plug-ins against the same ZeroMQ DSOs. Katana now also ships with the ZeroMQ static and shared libraries, and headers required to compile renderer plug-ins against ZeroMQ either statically or dynamically.
- Several environment variables have been deprecated in this release. For a full change list, including use and workaround information, see Deprecated Environment Variables in 1.5v1.

Feature Enhancements

- The job management system has been rewritten to provide a more robust multi-threaded environment.
- The Katana boot sequence has been rewritten to provide a single startup entry point. As part of the changes, a number of obsolete startup options have been removed.
- The CUE_THREADS environment variable has been replaced with the **--threads3d=X** command-line option in batch mode.

  The KATANA_RENDER_THREAD_COUNT environment variable has been replaced with the **--threads2d=X** command line option in batch mode.

NOTE: These command line options are only valid in batch mode. In all other modes, interactiveRenderThreads2d and interactiveRenderThreads3d preferences are used.

- A new UI for Python log messages has been added.

NOTE: You can find help on command line options using the following argument: **./katana --help**

- Documentation has been added for the Farm API. The documentation is an additional chapter inside the Technical Guide which can be found through either the **Help** > **Technical Guide** menu option, or located in the **docs/pdf** directory of the Katana install.
- The RenderOutputDefine node now queries the RendererInfo plug-in for its parameter policy and renderer specific settings instead of using the Python renderer plug-in interface.
- The renderer display drivers now use a unified RenderHost API for interactive rendering communication between the renderer and Katana.
- BUG ID 24238 - The use of the **FS** and **FE** (**Frame Start** and **Frame End**) environment variables to provide default project **inTime** and **outTime** values has been deprecated, as these can conflict with companies' internal environment variable names. This functionality can be restored by registering your own callback function, executed upon creation of a new scene:

```
import os
from Katana import Callbacks
def setFrameRange(**kwargs):
    from Katana import NodegraphAPI
    """
    Set the project InTime/OutTime, WorkingInTime/
    WorkingOutTime, and CurrentTime appropriately based
    on the environment variables 'FS' and 'FE'.
    """
    frameIn  = os.getenv('FS')
    frameOut = os.getenv('FE')
    if frameIn is not None:
        NodegraphAPI.SetInTime(frameIn)
        NodegraphAPI.SetWorkingInTime(frameIn)
    if frameOut is not None:
        NodegraphAPI.SetOutTime(frameOut)
        NodegraphAPI.SetWorkingOutTime(frameOut)
    inTime = NodegraphAPI.GetInTime()
    outTime = NodegraphAPI.GetOutTime()
    if NodegraphAPI.GetCurrentTime() < inTime:
        NodegraphAPI.SetCurrentTime(inTime)
    if NodegraphAPI.GetCurrentTime() > outTime:
        NodegraphAPI.SetCurrentTime(outTime)
Callbacks.addCallback(Callbacks.Type.onNewScene, setFrameRange)
```

Place the above script in a directory called **Plugins** within a path listed in your KATANA_RESOURCES environment variable. Alternatively you can add your script to an **init.py** placed in a directory named **Startup**, in the **.katana** directory found in your **Home** directory.

- BUG ID 25167 - Improved crash reporting:
  - Modified the crash report email subject line and body text to include the KATANA_RELEASE number.
  - Added environment variables to control crash reporting:
    - You can now set KATANA_CRASH_REPORT_DISABLE to 1 to disable crash reporting, including the sending of crash report emails.
    - You can also set KATANA_CRASH_REPORT_EMAIL_MODE to one of three values specifying the crash report email behavior:
      - 0: Skip the prompt and *never* send a crash report email.
      - 1: Skip the prompt and *always* send a crash report email to the addresses configured in KATANA_CRASH_REPORT_EMAIL.
      - 2: Prompt for user decision (default behavior).
- BUG ID 26086 - Added the ability to load the Node Graph when there are missing nodes in a Katana scene.

  Parameters on the node are viewable but cannot be edited and the node must be bypassed to allow the scene to be rendered. Edits can be made to other nodes and the scene can be saved to allow it to be opened as normal when the missing node types are found again.
- BUG ID 29425 - It is no longer necessary to set the element size when setting primitive scoped array data for arbitrary attributes. For example:

```
SetAttr("geometry.arbitrary.primitiveArrayTest.value",
    ScenegraphAttr.Attr("IntAttr", [0, 1, 2, 3, 4]) )
SetAttr("geometry.arbitrary.primitiveArrayTest.scope",
    ScenegraphAttr.Attr("StringAttr", ["primitive"]) )
```

No longer needs the addtional line:

```
SetAttr("geometry.arbitrary.primitiveArrayTest.elementSize",
    ScenegraphAttr.Attr("IntAttr", [5]) )
```

You can still use the element size attribute to override the automatic behavior.

- BUG ID 29841 - Support added for dicing camera using the camera.frameBeginCameraPath option in PrmanGlobalSettings.
- BUG ID 30523 - Any Viewer modifiers can now query input.getDrawOption("isPicking") to know, for example, whether to avoid enabling GL_BLEND during pick buffer drawing.
- BUG ID 31622 - Drag-and-drop from the **Attributes** tab into the **Python** tab now supports both explicit and implicit attribute representation.

  Explicit mode is triggered by pressing **Ctrl** while dragging.
- BUG ID 31885/36096 - The Python interpreter initialization has been modified to take into account different environment requirements, allowing Katana to initialize its environment safely, by avoiding inadvertent loading of unsupported modules.

  The KATANA_PYTHONPATH environment variable has been replaced by variables KATANA_DEBUG_PRE_PYTHONPATH and KATANA_POST_PYTHONPATH. The Katana Python modules search path is now built by Katana as shown below:

  KATANA_INTERNAL_PYTHONPATH =

  KATANA_DEBUG_PRE_PYTHONPATH :

  <Katana internal Python paths> :

  PYTHONPATH and site customizations :

  KATANA_POST_PYTHONPATH

NOTE: PYTHONPATH is not modified by Katana, and is inherited by child process as set in the user environment.

NOTE: Changes to **sys.path** included in **sitecustomize.py** do not affect the Katana internal Python paths.

NOTE: KATANA_DEBUG_PRE_PYTHONPATH is meant to be used solely for debug purposes, as it may lead to unexpected application behaviour due to non supported modules loading in place of the application's ones.

Changes to **sys.path** added through **sitecustomize.py** are now correctly taken into account.

- BUG ID 32555 - When using a custom Asset plug-in, to have that plug-in be the default, it was necessary to include a startup script with callbacks to set the Asset plug-in name, and File Sequence plug-in name.

Default Asset and File Sequence plug-ins are now defined in two environment variables:

KATANA_DEFAULT_ASSET_PLUGIN=yourAssetPlugin

KATANA_DEFAULT_FILE_SEQUENCE_PLUGIN=yourFileSequencePlugin

- BUG ID 32574 - Text with rich formatting was not converted to plain text when pasting into the command area of the **Python** tab, into script parameters of AttributeScript nodes in the **Parameters** tab, and into the text editor of BackdropNote nodes.
- BUG ID 32809 - The expression language function **getRenderLocation()** now returns a resolved filepath, if a render output is defined as a file sequence, and the current frame number is passed as an argument. For example:

  ```
  getRenderLocation(getNode('Render'), 'primary', frame)
  ```

  This allows users to more easily refer to previous outputs from dependent nodes.
- BUG ID 33142 - Katana now ships with the following OCIO configs:
  - aces
  - iif
  - nuke-default
  - spi-anim
  - spi-vfx

  Use the OCIO environment variable to control which config file is read. For example, if you’re running from a bash terminal:

  ```
  export OCIO=/myLUT/spi-anim/config.ocio
  ```

NOTE: The default OCIO config for Katana is **nuke-default**.

- BUG ID 33649 - A new Zlib folder has been added to Katana's bin folder. This folder contains Zlib DSO files compatible with Katana. By default, Katana uses the system version of Zlib, not the one included in Katana’s bin folder. However, on some Linux distributions the system Zlib conflicts with other libraries used by Katana. Users affected by this issue can now set the environment variable KATANA_USE_INTERNAL_ZLIB, which will make the Katana wrapper script ${KATANA}/katana load up the Zlib library shipped in Katana's bin folder instead of the system Zlib. Note that in general, you should not set KATANA_USE_INTERNAL_ZLIB unless advised to do so by our support team.
- BUG ID 34111 - The **material** group in the Gaffer Supertool is now expanded by default, saving a few mouse clicks for the user.
- BUG ID 34202 - Linux only: Katana is now statically linked against Boost 1.46.0 and the default visibility of all symbols is now hidden.

- BUG ID 34351 - Katana no longer tries to load incompatible render output types in the **Monitor** tab.
- BUG ID 34598 - Support has been added for Arnold Live Render updates for faceset materials.
- BUG ID 35173 - Sourcing a script from the **Python** tab no longer causes Katana to crash. This issue was seen in some window managers (notably Gnome).
- BUG ID 35771 - Added support for selecting subdivision surface scheme through PrmanObjectSettings nodes. The new parameter is **subdivisionMesh.scheme**. The available options for its value are:
  - catmull-clark
  - loop
  - bilinear
- BUG ID 35905 - PRMan user attributes are now declared with separate RiAttribute calls for readability.
- BUG ID 35962 - The ability to create debug output from a single Scene Graph location (using a command in the Debugging submenu of the context menu of Scene Graph locations in the Scene Graph tab) has been reinstated. This feature is currently only supported by PRMan.
- BUG ID 36095 - The PRMan **woff** option is now exposed through the PrmanGlobalSettings node.
- BUG ID 36401 - When clicking **Edit** > **Clear Catalog** in the **Catalog** tab, the FnDisplayDriver API was not sending the correct channel close message, which prevented Katana from releasing image data memory. If the render process exited unexpectedly, the image data was also not cleaned up correctly.

  This was fixed by using a ZeroMQ keep alive message which is sent periodically from the render side of the Display Driver, to instruct Katana not to clean up image data references.

  If the Katana Catalog Server does not receive a keep alive within a specified timeout, it cleans up references to any image structures it may have been holding on to in order to receive image data for the Catalog. By default, this garbage collection process occurs every 30 minutes, but can be controlled by setting the environment variable:

  KATANA_INTERPROCESS_GC_TIMEOUT_SECONDS
- BUG ID 36511 - During a PRMan Live Render, clicking **Rebind Shaders For Selected Locations** in the **Live Render Control** tab did not update the shaders.

- BUG ID 36535 - Previously, creating debug output for PRMan or Arnold aborted any currently active renders. Katana now treats Debug Outputs as a concurrent render and does not abort currently active renders.

NOTE: A render aborts a running debug output process unless concurrency is explicitly enabled.

Bug Fixes

- BUG ID 21834 - Context menu commands for disabling and enabling assets in Importomatic nodes in **Parameters** tabs used deprecated terminology **Ignore** and **Unignore** instead of **Disable** and **Enable**, and the key assignment **I** for Ignore instead of **D** for Disable.
- BUG ID 22039 - Node name wasn't preserved when applying undo after a "Convert to Group" command with LiveGroup nodes.
- BUG ID 26343 - The node names in parameter expressions were not being renamed correctly when a macro was imported into a recipe.
- BUG ID 27342 - Fixed issue with font families using Unicode characters in their names causing a crash.
- BUG ID 28250 - You couldn’t pick colors for shaders due to widget errors.
- BUG ID 28873 - The bounds attribute created by the IXformIteratorContexts in the Alembic_In node was of the form [minX, minY, minZ, maxX, maxY, maxZ] rather than [minX, maxX, minY, maxY, minZ, maxZ].
- BUG ID 28931 - Fixed a typo in the **Scripting using the NodegraphAPI** documentation.
- BUG ID 29028 - Loading the first recipe after initializing the application did not display the filename in the titlebar, it remained **Untitled**.
- BUG ID 29246 - Katana did not prompt to overwrite a Katana scene if the **.katana** extension was not entered explicitly by the user.

  This affected **File** > **Save As**, **File** > **Version Up**, and **Save** and **File** > **Export Selection** actions while using the default File asset plug-in.
- BUG ID 29303 - Master materials in the Gaffer node were not behaving as expected.
- BUG ID 29392 - While editing text, switching desktops resulted in the edits being lost.
- BUG ID 29404 - The **xform** attribute for viewer modifiers always returned as live when querying the ViewerModifierInput::isLiveAttribute method.
- BUG ID 29461 - **GetShutterOpen()**, **GetShutterClose()**, and **GetNumSamples()** were failing in AttributeScripts. This was due to the

architectural changes from 1.0 to 1.1. Further AttributeScript failures from the architectural changes were also discovered and fixed.

- BUG ID 29545 - Calls to Slo_SetPath from within the KatanaProcedural resulted in unpredictable behavior.
- BUG ID 29593 - Right-clicking on an unwanted Render node port and selecting **Delete Port** did nothing.
- BUG ID 29738 - Framing in the **Viewer** tab was not using the correct bounding box.
- BUG ID 29827 - Menus for shader parameters on nodes of type ArnoldShadingNode and PrmanShadingNode were not showing all available options.
- BUG ID 29831 - Shader UI hints in an **.args** file for ArnoldShadingNode nodes were being ignored after passing through a NetworkMaterial node.
- BUG ID 29845 - Parameters of LookFileMultiBake nodes were not displayed in the **Parameters** tab.
- BUG ID 29874 - Performing material edits or overrides on co-shaders that didn't exist resulted in all future co-shaders being sent to the renderer incorrectly.
- BUG ID 29881 - Katana hotkeys weren't functioning correctly under KDE.
- BUG ID 29981 - The Makefile for compiling the KatanaProcedural.so had incorrect dependencies.
- BUG ID 29990 - The **xAxis**, **yAxis**, and **zAxis** attributes of a point constraint were incorrectly defaulting to **false** when not found in a look file.
- BUG ID 30156 - The Makefiles for the renderer plug-ins contained unnecessary RPATHS. Also, a number of library dependencies were removed as part of the fix.
- BUG ID 30194 - Referencing materials from look files in the MaterialStack node didn't work. This was due to earlier changes to the Isolate node's behavior.
- BUG ID 30361 - The GeoMaker example plug-in Makefile was not working correctly.
- BUG ID 30451 - Right-clicking in the empty area of the **materials** list of MaterialStack nodes in the **Parameters** tab raised a Python exception.
- BUG ID 30452 - A number of menu options caused a Python exception, namely:
  - The **Tear Off Parameters...** option in the shortcut menu of a material in the parameters of a MaterialStack node.
  - The **Tearoff Parameters of Override...** option in the shortcut menu of an override in the parameters of a LookFileManager node.

- The **Go to [Node Name]** option in the pop-up menu of a node entry in the parameters of a ShadowRenderAssign node.

- BUG ID 30651 - The **Adopt Scenegraph Selection** menu option was missing from the popup menu of Scene Graph location editor widgets that were part of arrays of locations.
- BUG ID 30698 - When using the tile stitching command-line option, all tiles were stiched across the top row.
- BUG ID 30790 - PRMan visibility attributes were incorrectly converted from **float** to **int**, resulting in corrupted values.
- BUG ID 30802 - Calling **checkDynamicParameters()** caused a traceback when a **universalAttr** parameter was not passed.
- BUG ID 31009 - The tooltip of the **Add** button on LookFileMultiBake nodes used the old node name, KatanaStdBake.
- BUG ID 31076 - The red glow indicating an error on a node was, in certain cases, cleared upon switching of the viewed flag between nodes.
- BUG ID 31432 - The toggled menu items in the **Disk Render Dependencies Before** section of the context menu of nodes in the Node Graph tab did not apply to renders triggered using the **Render** > **Repeat Previous Render** command.
- BUG ID 31545 - The **Render Log** tab used a fixed-width font family that was not available on all systems. It now attempts to pick an available fixed-width font family.
- BUG ID 31802 - Katana crashed when calling the **insertArrayElement( )** function on a NodegraphAPI Parameter object with an invalid index, on an expressioned parameter or on a parameter belonging to a locked node.
- BUG ID 31910 - When the asset plug-in was changed in the Project settings, Katana displayed an error when the currently open scene was not compatible with the current asset plug-in. Removed this for cases in which the current scene is a valid file.
- BUG ID 31991 - Browsing for a LiveGroup asset from node parameter widget used a different file filtering than the **File** > **Import LiveGroup...** menu item.
- BUG ID 32096 - Arnold Preview Renders contained a single empty pixel row and column on the bottom and right edges of the image respectively. There was also a skipped pixel row at the top of the bottom-most row of render buckets, with all pixels below this shifted upwards.
- BUG ID 32555 - When using a custom Asset plug-in, to have that plug-in be the default, it was necessary to include a startup script with callbacks to set the Asset plug-in name, and File Sequence plug-in name. Default Asset and File Sequence plug-ins are now defined in two environment variables:

KATANA_DEFAULT_ASSET_PLUGIN=yourAssetPlugin

KATANA_DEFAULT_FILE_SEQUENCE_PLUGIN=yourFileSequencePlugin

- BUG ID 32681 - The FarmAPI method **IsSceneValid()** returned warning messages even if flags for alphanumeric, capital and unwanted symbols were not set.
- BUG ID 32684 - Python or C++ implementations of AssetAPI plugins did not have their **reset()** method called when the **Flush Cache** button was clicked.

  In addition, The AssetAPI python module now exposes the function ResetAllAssetAPIPlugins() which has the effect of calling reset() on all loaded AssetAPI plugins. This can be called on an adhoc basis by users.
- BUG ID 32714 - It was tricky to select text in the **Render Log** tab due to an offset in the position of the mouse pointer. Also, the **Render Log** tab did not show the text selection cursor.
- BUG ID 32807 - Using conditional visibility options for fileInput and asset widgets caused an empty file browse dialog to be displayed when browsing for a file or asset.
- BUG ID 32835 - Import LiveGroup command resulted in a node with name set to **unknown**.
- BUG ID 33257 - Caches from Alembic files, loaded using Alembic_In nodes, weren't flushed correctly when the result of the Alembic_In node was displayed in the viewer.
- BUG ID 33575 - An exception was raised when running the Renders shelf item script that prevented the selected Render nodes from being framed in the Node Graph tab.
- BUG ID 33607 - The ImportomaticAPI no longer contains old style Python classes.
- BUG ID 33726 - The title of the **Edit Shelf Item** dialog was misleading, as it does not provide editing capabilities.

  It has been renamed to **Shelf Item Source** and now also includes the name of the shelf item whose source is shown.
- BUG ID 33727 - The **Shelf Item Source** dialog (previously titled **Edit Shelf Item**) did not scale well.
- BUG ID 33805 - Katana attempted to set the now deprecated Arnold option, **unload_plugins**, causing a warning to be printed during shader loading.
- BUG ID 34009 - PrmanGlobalSettings node now exposes the ErrorHandler parameter that allows to set one of the default PRMan error handlers.
- BUG ID 34150 - ShadowBranch did not prune locations correctly for **none** and **lights and surfaces** prune modes.

- BUG ID 34189 - Changing frames when live rendering now triggers a new render to guarantee a consistent scene state.
- BUG ID 34238 - Some Material updates were not correctly passed to the PRMan renderer in Live Rendering.
- BUG ID 34259 - Opening documentation with Firefox through Katana sometimes failed due to legacy handling for old versions of Firefox.
- BUG ID 34381 - Added a makefile for the FileAsset plug-in.
- BUG ID 34398 - The default window size of the **Help** browser was too small for the **Examples** help page. The default window size has been increased to 1000x800.
- BUG ID 34589 - **resolvePath()** was being called inconsistently for Disk Renders.

  Now **resolvePath()** is called for all render output types.
- BUG ID 34691 - Several warnings were raised when compiling the Render APIs (uninitialized variables, unsigned/signed comparison).
- BUG ID 34905 - Katana crashed with a **Floating exception** when working with array parameter expressions.
- BUG ID 34930 - Array parameters with erroneous expressions defaulted to an invalid tuple size of 0, causing a crash in certain cases.
- BUG ID 34950, 35200 and 35689 - The **trace.decimationrate** option in the PrmanGlobalSettings node was defined with the wrong type (**float** instead of **int**).

  The **options.shade.cachemode** parameter of PrmanGlobalSettings nodes was not mapped to the to corresponding **_cachemode** RiOption correctly.

  Some parameters that are not currently - or are no longer - supported have been removed from the PrmanGlobalSettings node:

  - camera.frameBeginCameraPath
  - commandArguments.prmanVersion
  - commandArguments.prmanPluginVersion

  The **hair.minwidth** parameter in the PrmanGlobalSettings node was defined with the wrong type (**int** instead of **float**).

NOTE: Scenes created prior to the fixes to bugs 34950, 35200, and 35689 must be converted to the current standard using the PRMan 17 update script. See PRMan Scene Update for more on this.

- BUG ID 35054 - The popup of included outputs of a Render node in a Parameters tab used a wrong title.

- BUG ID 35056 - Fixed an exception when right-clicking a Render node with no input ports, but also disabled the menu option to delete or rename the first input port.
- BUG ID 35233 - Using a custom asset plug-in and saving a scene triggered the **Scene File Exists** override dialog.
- BUG ID 35335 - Overscan settings for color outputs were not respected in Disk Rendering.
- BUG ID 35447 - Overscan settings were not respected when Preview Rendering with Arnold.
- BUG ID 35533 - Katana crashed when choosing **Expression** in the context menu of array parameters in **Parameters** tabs.
- BUG ID 35545 - ExrCombine failed when merging outputs into a single image with overscan.
- BUG ID 35556 - Exceptions raised while evaluating AttributeScript were incorrectly converted to textual representation.
- BUG ID 35639 - Render outputs were resolved internally, in addition to in the **File** location plug-ins.
- BUG ID 35662/35686 - Some widgets in the **Parameters** tab were not disabled for parameters of locked nodes.
- BUG ID 35772 - With no Asset system loaded, Katana would not issue a warning dialog when overwriting a Katana file using **File** > **Save As...**
- BUG ID 35795 - No errors were reported when Live Rendering certain invalid render output configurations.
- BUG ID 35884 - The representation of Scene Graph locations of multiple dragged nodes inconsistently separated paths with commas instead of spaces, which lead to errors when dropping nodes into certain widgets, for example in the **Parameters** tab.
- BUG ID 36025 - The Arnold plug-in had unnecessary dependencies including expat and Python. Additionally, it was missing some C standard library includes.
- BUG ID 36026 - The Arnold plug-in improperly handled camera focal distance / shadow bias attributes.
- BUG ID 36089 - Args files on a RendererProceduralArgs node were not loaded using the File asset plug-in.
- BUG ID 36419 - Numeric values of arbitrary vector attributes were incorrectly converted to double values instead of float values by the Katana Arnold plug-in.
- BUG ID 36652 - Errors shown in the **Scene Graph** tab were not always displayed immediately as node errors in the Node Graph tab.

Known Issues & Workarounds

Asset API

- BUG ID 35334 - Katana gives an error loading a LiveGroup on startup when using an asset plug-in that was not previously selected.
- BUG ID 35321 - The PrmanObjectSettings node does not resolve an asset ID in a **ribInclude** when using Debug Output.
- BUG ID 35318 - The OCIOFileTransform node is not asset-managed.

Miscellaneous

- BUG ID 37256 - When using the RGBA widget type, RGB values in the color picker are reset to 0 when the alpha is set to the default value 0.
- BUG ID 36926 - When working with PRMan, RenderOutputDefine **rendererSettings** > **displayOptions** parameters cannot be set through the **Parameters** tab.

  The output display options can be set directly on the **/root** location, using an AttributeSet or AttributeScript node:

  `renderSettings.outputs.<outputname>.rendererSettings.displayOptions.<optionname>`
- BUG ID 36691 - State badges are currently shown for attribute values of dynamic array child parameters, even though only their parent array parameter should appear with a state badge.
- BUG ID 36566 - Moving lights in between rigs duplicates them during a PRMan Live Render.
- BUG ID 36520 - PRMan Live Rendering: The **Update Selected Coordinate Systems** action does not apply the correct transform to the coordinate system.
- BUG ID 36225 - The UpdatePrmanSettingsNode shelf script currently does not deal with expression parameters. If you have locally set parameter expressions on PRMan nodes, you must re-adjust them after running the shelf script.
- BUG ID 36176 - The 2D node **Disk Render Upstream Render Outputs** option does not use the batch render method, **batchRender**, for upstream render nodes, but uses **diskRender** instead.
- BUG ID 36170 - Control keys (notably arrow keys) do not function as expected in shell mode.
- BUG ID 36168 - Transformation of Gaffer light rigs while Live Rendering are not applied correctly.
- BUG ID 35122 - OpenEXR 1.6 restricts channel names to a length of 32 characters (31 characters plus null-byte). Katana's **exrcopyrenderattrs**

script, which runs as a post process, uses OpenEXR 1.6 and fails for names longer than 31 characters.

- BUG ID 34870 - Katana does not support render output directory creation for **shadow**, **merge** and **script** output types.
- BUG ID 34669 - Currently, **exrcopyrenderattrs** fails to copy EXR header information when tile stitching.
- BUG ID 34289 - The **Render Log** output may contain superfluous newlines.
- BUG ID 34274 - PRMan Live Rendering, Reyes mode: Lights are duplicated when sending Live Render updates.
- BUG ID 34256 - Preview Rendering 2D nodes fails when **Disk Render Upstream Render Outputs** is set to **Unless Already Cached**.
- BUG ID 33242 - Live Rendering is not designed to work with rapid updates to region of interest (ROI), and attempting rapid ROI changes may cause unexpected behavior.
- BUG ID 32770 - Due to a known issue with OpenEXR versions 1.6.1 and earlier, EXR channel names cannot exceed 31 characters in length.
- BUG ID 32230 - Parent transforms are not applied to lights after the first pass of a Live Rendering (re-rendering) session.
- BUG ID 32159 - The Alembic library does not support multiple process / thread access to an Alembic file. This means that modifying an Alembic file outside Katana, while it's loaded in an open Katana scene, results in a Katana crash.

  To avoid this, either close any Katana scenes before modifying included Alembic files, or collapse the Scene Graph of any Katana scenes to **/root**, and click **Flush Caches** before attempting to update any modified Alembic files.

TIP: To collapse the Scene Graph to **/root**, right-click on **/root** in the **Scene Graph** tab and choose **Close All**.

- BUG ID 32125 - Live Rendering complex scenes using anything other than **Selected Scenegraph Locations** can cause Katana to crash.
- BUG ID 27147 - Due to an internal bug in Qt 4.7.2 an error message similar to the following may occasionally be displayed in the console:

  ```
  Application asked to unregister timer 0x13000076 which is
  not registered in this thread. Fix application.
  ```

  https://bugreports.qt-project.org/browse/QTBUG-16175
- BUG ID 22296 - Due to a known issue with PRMan, Katana does not currently support PRMan Live Rendering with AOVs. As a workaround, to allow Live Rendering, disable AOVs in interactive renders by applying an

interactive render filter, using a RenderSettings node with the interactiveOutputs parameter set to just **primary**.

NOTE: In a scene with AOVs defined, attempting PRMan Live Rendering without disabling them using the workaround described in Bug ID 22296 produces unexpected results, and does not render the AOVs.

Deprecated Environment Variables in 1.5v1

The following table lists all environment variables which have been deprecated in this release, what the purpose of the environment variable was, and any workarounds that can be used instead.

Environment Variable	Purpose	Workaround
KATANA_FLUSH_SLM_DSOS	Used to flush/delete plug-ins and reload.	This is now done through **clearDsoCache**.
KATANA_EXIT_ON_ERROR	Used to determined if IMGLIB exits on detection of an error.	N/A
KATANA_DEBUG_PRMAN	Used to determine whether the **globalSettings.debugStream** is pointer at **std::cerr**	N/A
KATANA_MEMORY_DEBUG_PRINT	Used during debugging to print usage in 2D subsystem.	N/A
PYCHECKER	Used to circumvent usage of particular modules due to conflicts with PyChecker.	Conflicts were resolved.
KATANA_GLM_DSOPATH	Used to specify a file path to prepend to a dso name.	KATANA_RESOURCES provides this now.
KATANA_FORCE_GLSL	Used to force setting of graphics card support even if GLEW_EXT_geometry_shader4 check failed.	Graphics cards must support GLEW_EXT_geometry_shader4.
GINSUD_TMPDIR	Used to specify GINSU daemon temp directory.	N/A
CUE_FRAME_ID	Used to specify the frame ID to be sent to in-house render farm system used by Sony.	N/A
SPLAT_VERSION	Used to specify an in-house volume renderer used by Sony.	N/A
SVEA_VERSION	Used to specify an in-house volume renderer used by Sony.	N/A

Environment Variable	Purpose	Workaround
SVEA_PYTHONEXEC	Used to determine which Python interpreter to use.	Use The Foundry's digitally signed interpreter.
PYTHONEXEC	Used to determine which Python interpreter to use.	Use The Foundry's digitally signed interpreter.
SHADOWCONVERT_VERSION	Set an application version for **vshd** shadow types.	See CUE_THREADS and KATANA_RENDER_THREAD_COUNT under Feature Enhancements for more information.
KATANA_PROFILE_PRMAN	If set, it would pass **-timer** to the PRMan render process during rendering.	Use PRMan built-in profiling.
MAYA_VERSION	Used when Viewer functionality was provided by Maya.	Use the Viewer.
SPIMPORT_DEBUG	Sony specify variable, caused debug versions of libraries to be loaded rather than the default which is release libraries for the **SpImport** module.	Debug libraries never loaded.
KATANA_IGNORE_USER_RESOURCES	Cause $HOME/.katana to be omitted from the resource file search path.	N/A
FS	Frame Start - defines the current shot frame range.	See bug #24238 under Feature Enhancements.
FE	Frame End - defines the current shot frame range.	See bug #24238 under Feature Enhancements.
KATANA_RIBDIR	Used to set where renderable RIB files were written to.	This is now KATANA_TMPDIR.
SPIMPORT_PACKAGES	Sony specific variable.	Use new PYTHONPATH variables as described in bugs 31885/36096 under Feature Enhancements.
KATANA_PYTHON_LIBS	Used to point to dynamically loaded components.	N/A
COLOR_OVERRIDE	Used to point to a new colorspaces xml file.	N/A
COLOR_HOSTMASK_DISABLE	Used to localhost support in colorspace parser even if localhost is not in the host mask file.	N/A
CUE_THREADS	Defined the number of 3D rendering threads.	Set **--threads3d=X** command line option in batch mode, or Set **interactiveRenderThreads3d** in UI/Script/Shell mode.

Environment Variable	Purpose	Workaround
KATANA_RENDER_THREAD_COUNT	Defined the number of 2D rendering threads.	Set **--threads2d=X** command line option in batch mode, or Set **interactiveRenderThreads2d** in UI/Script/Shell mode.