Release Notes for Mari 2.0v1

Release Date

14 March 2013

Best Bits of 2.0v1

- Layers, accessed from either the Layers menu or from within the Layers palette, allow you to set up complex layer stacks for channels. For more information regarding layer integration, please refer to the Layers chapter in the Mari *User Guide*.
- Procedurals and adjustments can now be added using the Layers menu or palette. Procedurals and adjustments contain many of the former shader modules.
- Channels now contain layer stacks with mask, procedural, and adjustment data. Channels can be used in shader component inputs to specify how shaders impact individual channels.
- Shaders no longer operate on a modular system. Instead they use inputs to control how lighting, ambient occlusion, and specularity (among other components) interact with given channels.
- Channels, shaders, and layers can now be imported and exported, either in a full or flattened state.
- Sharing layers into a layer stack, or as a new channel, allows you to quickly transfer paint made in one layer to another.
- New selection methods allow you to select, lock and hide patches, objects, channels, and faces (or a combination of these).

System Requirements

Officially supported operating systems

- Windows 7 64-bit
- Linux 64-bit operating system (Fedora 12 or Red Hat Enterprise Linux 5.4)

NOTE: Some newer Linux distributions have changed their network device configuration handling and won't work with Mari without some configuration changes. Known affected distributions at this time include Fedora 16, Red Hat Enterprise Linux 6, and CentOS 6 on certain, notably Dell, hardware. Please see our FAQs for more information: http://www.thefoundry.co.uk/support/faqs/#question_97

Minimum hardware requirements

- Quad-core processor
- 250GB disk space available for caching and temporary files (or a minimum of 50GB if you're working on a small project)
- At least 4GB RAM
- Display with 1680 x 1050 pixel resolution
- An NVIDIA or AMD graphics card with at least 1GB of RAM and OpenGL 3.2 support (see Tested graphics cards below)

Virtual memory requirements

It is recommended that virtual memory be available for Mari to use, especially when handling large projects. The use of virtual memory improves stability, helping to prevent data loss on large projects.

Tested graphics cards

- NVIDIA GeForce GTX 480
- NVIDIA GeForce GTX 580
- NVIDIA GeForce GTX 680
- NVIDIA Quadro (Fermi Series) 4000
- NVIDIA Quadro (Fermi Series) 5000
- NVIDIA Quadro (Fermi Series) 6000
- AMD FirePro W7000
- AMD FirePro W8000
- AMD FirePro W9000
- AMD Radeon HD 7850
- AMD Radeon HD 7950

Please download and install the latest graphics driver for your card from the NVIDIA or AMD websites.

NOTE: Ensure that you are using 8.982.8.1 (AMD Catalyst 12.10) drivers or higher for AMD cards.

New Features

Layers

- Mari now has a more robust layer system that integrates with channels and shaders. Either from the **Layers** menu or from within the **Layers** palette, you can create:
  - layer stacks,
  - layer masks,
  - adjustment layers,
  - procedural layers, and
  - layer groups.
- Additional layer functionality allows you to:
  - cut, copy, and paste layers from one layer stack to another,
  - duplicate and merge layers,
  - cache large layer stacks or layer groups, and
  - use blend modes on specific layers.
- The **Layer Properties** menu option has been added to the context menu when right-clicking on a layer in the **Layers** palette.

  This option opens a new window displaying all the pertinent information about the layer's paint, including the file and color space formats, channel type, and any notes made about the layer.
- Mari now has advanced blending options for layers. This feature provides curves for adjusting the degree of blending using the intensity or individual color components of layers. These can be accessed from the right-arrow button to the left of the **blend amount** slider in the **Layers** palette.
- Color tags in the **Layers** palette allow you to assign tags to specific layers and filter the layer stack by tag color.
- **Share Layer** has been added to the context menu for the **Layers** palette. This can be accessed by right-clicking on a layer and navigating to **Sharing** > **Share Layer**.

For more information on using layers, please refer to the *Layers* chapter on page 223 in the Mari *User Guide*.

For information on the different procedural and adjustment types, please refer to the *Layers Palette* chapter on page 82 in the Mari *Reference Guide* or the section on *Layer Types and Examples* on page 244 of the *User Guide*.

Shaders

- Mari's shaders and lighting have been improved:
  - The default shaders **Current Channel**, **Layer and Below**, **Layer**, and **Paint Target** shaders support **Flat**, **Basic**, and **Full** lighting modes and have shader components that allow for channel inputs.
  - The number of lighting shaders available has been extended to include the Phong, Blinn, Beckman and Cook Torrance models. You can also mix-and-match different diffuse and specular models.
- **Cut**, **Copy**, and **Paste** actions have been added to the context menu for the **Shaders** palette. This allows you to cut, copy, and paste shaders within the same object and from one object to another.

For more information on using shaders, please refer to the *Shaders* chapter on page 281 in the Mari *User Guide*.

Channels

- Channels contain layer stacks with mask, procedural, and adjustment data. Channels can be used in shader component inputs to specify how shaders impact individual channels.

  Quick channels also allow you to quickly set up a channel from within the **Channels** palette without going through the **Add Channel** dialog.
- **Bleed Patch Edges**, available in the **Channels** menu or the right-click menu in the **Channels** palette, can be used on all channels or patches that shader inputs depend on.
- **Bleed Patch Edges** in Mari version 2.0v1 operates on an entire channel, that is, on all paintable textures including layer paints and layer masks contained in the channel.
- By design, Mari filters need to be applied to a single paintable texture, and not layered textures. From Mari 2.0v1 onward, channels can contain multiple layered textures in layers, so filters can no longer be applied at the channel level.

  To apply a Filter to a paintable texture, select the **Current Paint Target** option in the **Filters** dialog.
- The **Clone Stamp** tool now uses **Current Paint Target** in the **Source** dropdown instead of using Channels.

For more information on using channels, please refer to the *Channels* chapter on page 264 of the Mari *User Guide*.

Patches

- **Fill** menu options have been added to the **Patches** menu and the right-click context menu on the canvas under **Patches** > **Fill**. Options include:
    - **Black**,
    - **Gray**,
    - **White**,
    - **Foreground**,
    - **Background**,
    - **Transparent**,
    - **UV Mask**, and
    - **Wireframe**.

  The selected patches of the selected layers are filled with the selected color, transparency, or mask, and can be applied to an entire model by selecting all patches. Filling either the **UV Mask** or **Wireframe** applies only to selected patches of the current layer's current component, if paintable.
- **Transform** patch actions are now found on the **Patches** menu or by right-clicking on the canvas and navigating to **Patches** > **Transform**.
- **Mirror** patch actions are now found on the **Patches** menu or by right-clicking on the canvas and navigating to **Patches** > **Mirror**.

For more information on patches, please refer to the *Working with Patches* chapter on page 182 of the Mari *User Guide*.

Caching

- In the **Layers** menu and the right-click context menu in the **Layers** palette, there are options to:
    - **Cache Layer**,
    - **Cache Up to Here**,
    - **Uncache Layer**,
    - **Uncache Up To Here**.

  Caching or uncaching a layer also locks or unlocks it to prevent it from being edited. Caching or uncaching a layer "up to here" also caches/uncaches and locks/unlocks an entire layer stack up to the specific point.
- Indicators have been added to layer widgets in order to show the computational cost of the layer and the accumulated cost of any previous layers. The colors range from green to red, or blue if a layer has been cached.

For more information on caching, please refer to the *Layers* chapter, specifically the sections on layer caching on page 231 and 240 of the Mari *User Guide*.

Import and Export

- Importing and exporting channels has been updated. These options can be found under the **Channels** menu or by right-clicking in the **Channels** palette. Options for exporting channels include:
  - **Export Current Channel**,
  - **Export All Channels**,
  - **Export Everything**,
  - **Export Current Channel Flattened**,
  - **Export All Channels Flattened**, and
  - **Export Everything Flattened.**

  Exporting flattened exports the textures in a flattened state but does not affect the channels in the **Channels** palette.

  Options for importing channels include:
  - **Import into Layer Stack**, and
  - **Import into Channel**.
- Importing and exporting layers has been updated. These options can be found under the **Layers** menu or by right-clicking in the **Layers** palette. Options for exporting layers include:
  - **Export Selected Layers**,
  - **Export All Layers**,
  - **Export Selected Layers Flattened**, and
  - **Export All Layers Flattened**.

  Exporting flattened exports the textures in a flattened state but does not affect the layers in the **Layers** palette.

  Options for importing layers include:
  - **Import into Layer** and
  - **Import into Layer Stack**.
- The Import/Export Sequence dialog includes example templates.
- Mari's import/export workflow has been improved, including:
  - A new progress bar during exports, and
  - the **Mari Warnings** dialog has been renamed **Message Log**.
- The **Export** dialog has a field to **Use Template For**, allowing you to use a template for multiple files.
- When importing from **.psd** files, you can skip or update layers with names that match the ones in the existing layer stack.

For more information on importing and exporting, please refer to the *Exporting and Importing Textures* chapter on page 140, or for more specific information on importing or exporting channels or layers, please see the *Layers* chapter or *Channels* chapter in the Mari *User Guide*.

Project conversion

- When converting old projects to Mari 2.0, a dialog appears to warn you that conversions are permanent and cannot be undone. You cannot continue with the conversion until you acknowledge that you have read this warning.

  To avoid the risk of a project being permanently converted, regardless of the result, it is highly recommended that you create an archive of your project first.
- Mari warns you when converting projects with channels that the conversion process may take longer than expected, due to the RGB and RGBA conversion process. This process cannot be reversed.
- When converting old projects, shaders use the **Current Channel** default shader.

For more information on project conversion, please refer to the *Project Conversion* chapter on page 221 of the Mari *User Guide*.

Mari Preferences Dialog

- The graphics card preferences in the **Mari Preferences** dialog box have now been grouped into their own tab labeled **GPU**.
- The preference option **Tessellation Allowed** has been added to the new **GPU** tab in the **Mari Preferences** dialog. Disabling this option completely disables tessellation shader support. Mari then treats any project with displacement as if the graphics card does not support tessellation.
- The **Use HiRes Positions** option has been added to the **Mari Preferences** dialog box, found by navigating to **Misc** > **Input | Use HiRes Positions.** Turning this off may help users who experience problems using Wacom devices.
- You can reset the **Channel Cache** controls to the default settings by pressing the **R** button in the **Preferences** > **Channel Cache** dialog.
- Renamed the **Channels** path in the **Paths** tab of the **Mari Preferences** dialog box to **Channel Presets** to better reflect its purpose.

Miscellaneous

- In the UV view, overlay text showing the patch sizes has been changed. It now uses abbreviated notation, such as 4K instead of 4096x4096, for all of the non-current image sets, and displays all active patch sizes.
- New options have been added to the HUD Manager, and existing options have been reorganized.
- Mari's HUD includes an option to display the maximum number of used texture units.
- New icons have been added for several palettes, including the Snapshots palette, Python Console palette, Layers palette, and Shaders palette. Icons have also been added for some menu actions, such as Cut.
- Projectors now support unprojection from the default Current Channel, Layer and Below, Layer, and Paint Target shaders.

  You can also set the lighting (Flat, Basic, or Full) independently of the Lighting toolbar during uprojection.

Feature Enhancements

Nuke<>Mari Bridge

- BUG ID 31077 - The Nuke<>Mari bridge has been updated. The following changes have been made to the functionality:
  - All projector paint is now baked into its own layer, and as such, the Bake option has been removed. You can still use a projector to import paint to the paint buffer for warping and similar actions.
  - Each projector is baked into a layer with a white mask in the specified channel. These masks can be painted to blend between the different projector images.
  - On export to Nuke, the channel(s) being sent as UV textures are flattened.

Import and Export

- BUG ID 34276 - The exporter now replaces any characters that are invalid in the output file name with underscores. For example, invalid characters such as * or / in the file name are replaced with an underscore character, allowing it to be saved correctly.

Shortcuts

BUG ID 34126 - Old, redundant actions have been removed from the Manage Keyboard Shortcuts dialog and the Manage Toolbars dialog. These include:

- Add Channel, Remove Channel, Duplicate Channel, Merge Channel, and Convert Channel,
- Bake Attribute and Bake Shader,
- Alpha to Mask and Mask to Alpha,
- Export Modified,
- Save Shader, Load Shader, Duplicate Shader, Add Surface Shader, and Remove Surface Shader.

A few actions have also been moved within the action tree in the same dialogs. These actions include: Lock Channel and Unlock Channel.

UI improvements

- The Duplicate Channel menu option has been removed.
- The Copy Shaders menu option has been removed.
- The Match Channels menu option has been removed from the right-click context menu for objects in the Objects palette.

Miscellaneous improvements

- After converting a project to Mari 2.0v1, the previously selected shader in the 1.6 project is now the shader that's selected.
- The lighting mode has been removed from Quick Projections.
- BUG ID 11213 - The Palettes menu now displays the palette names in sorted order.
- BUG ID 13363 - The performance and frame rate for the Pinup tool has been improved.
- BUG ID 23073 - Tokens used in file templates, such as $CHANNEL and $UDIM, are now replaced using whole word matches, in upper case only. This allows you to use custom tokens with common sequences, such as $TEST and $TEST2, at the same time.
- BUG ID 23402 - You can now use $CHANNEL in turntable file names.
- BUG ID 27691 - You can now change the file filter to show all files in the Open Archive dialog box. If your archive doesn't end with **.mra** it can be selected once the wildcard filter is selected.
- BUG ID 28138 - The depth buffer rendering has been moved to the start of the deferred rendering chain in order to improve painting frame rates.
- BUG ID 30609 - The default lighting mode when a project is first loaded has been changed from Flat to Full (with diffuse and specular lighting).
- BUG ID 31245 - The Mari<>Maya script has been updated to use layers instead of image sets.
- BUG ID 32707 - Mari and Nuke now calculate grading without the minor dissimilarities that sometimes appeared before.

- BUG ID 32840 - The **Shaders** dropdown has been removed from **Render Turntable**, which defaults to the currently selected shader.
- BUG ID 32959 - The diagnostic turntable has been modernized so that the dialog lists 2.0v1 channels, and renders the checked channels.
- BUG ID 33552 - On the **Clone Stamp** toolbar, the **Source** dropdown menu alphabetizes the channel options.
- BUG ID 33664 - The **Fill**, **Transform**, and **Resize** patch functions only present a single dialog, instead of multiple requests to apply the operation to patches and layers.
- BUG ID 33689 - Improved the selection tool and tablet input context menu behavior. Mari now stops the selection being changed if the context menu appears.
- BUG ID 34014 - The Foundry EULA has been updated in the Mari *User Guide* in Appendix D.
- BUG ID 34406 - The lock and unlock options on the **Selection** menu have been renamed to **Lock Unselected**, **Lock Selected**, **Unlock Selected**, and **Unlock Entire Object**.
- BUG ID 34439 - The **Flow** procedural can now be cached, but only as part of a layer stack. It cannot be cached as a layer on its own, due to its similarity to adjustments. As adjustments are not free standing, the **Flow** procedural can only be cached in the layer stack. The cached version is the same as what you see when the **Animated** checkbox is not selected.

Bug Fixes

- BUG ID 27777 - The selection highlights on selected items disappeared when navigating.
- BUG ID 28645 - Mari crashed if an export string token, for example **$GEO**, contained characters that resulted in an invalid filepath.

  Mari now checks the file path for invalid characters and displays a warning if the path is not valid.
- BUG ID 28656 - The **Cancel** button on the **Fast Exporter** dialog did not work as expected.
- BUG ID 30546 - Mari did not interpret single-channel images correctly for the **.tif** file format.
- BUG ID 32537 - Mari occasionally crashed when using the **Magic Wand** selection tool.
- BUG ID 33063 - Using **Alt** + the right pen button on tablets changed the object/patch/face selections when finishing a camera navigation operation.
- BUG ID 33064 - The **None** blending mode should not have been included in the **Painting Mode** dropdown menu.

- BUG ID 33227 - An invalid user shortcut assignment for **Bake On New Channel** was linked to a non-existent action that caused Mari to overwrite a valid shortcut assignment.
- BUG ID 33355 - Shortcuts assigned to collection items (such as lights or projectors) displayed the assigned shortcut icon in the wrong location in the palette.
- BUG ID 33486 - Renaming a project in the **Projects** tab when the project was closed didn't assign the new name to the project in the title bar.
- BUG ID 33488 - The **New Project** dialog could be stretched over multiple screen widths after creating a new project containing multiple objects with long file paths.
- BUG ID 33548 - There was no warning that patches were locked when attempting to resize them.

  Mari now displays a warning dialog to notify you that a patch is locked.
- BUG ID 33694 - The **Toggle Hidden** option has been removed from the list of patch functions and the patches context menu.
- BUG ID 33697 - Duplicating a channel when non-sequential patches were used caused Mari to crash.
- BUG ID 33956 - The **Max Image Memory** preference under **Mari Preferences** > **GPU** had no maximum limit, allowing it to be set to unrealistically large values.
- BUG ID 34201 - When the **Save Changes Before Closing** dialog was visible, the cursor seemed to have disappeared from the canvas.

Known Issues & Workarounds

- BUG ID 11874 - Mari doesn't recognize 3-digit padded **.obj** sequences as animation.
- BUG ID 12102 - Current brush settings do not get saved as part of the project. Instead, Mari reverts to the default settings when you close and relaunch it.
- BUG ID 12567 - Enabling **Sync to VBlank** in NVIDIA settings can drastically reduce Mari's performance. If you experience very slow interaction, even with low-polygon models and one of the Tested graphics cards, navigate to:
  - Linux: **NVIDIA X Server Settings** > **X Screen 0** > **OpenGL Settings** and turn off **Sync to VBlank**.
  - Windows: **NVIDIA Control Panel** > **3D Settings** > **Manage 3DSettings** > **Vertical Sync** > **Force off**

  Then, restart Mari.

- BUG ID 13294 - Windows: Mari sometimes crashes when trying to load data on large projects due to the program exhausting all Window manager objects.

  To reconfigure the user object limit:

  - Open regedit and navigate to **HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\NT\CurrentVersion\Windows**, and
  - Edit **USERProcessHandleQuota** to a larger number.

  If this number gets too large, you may also have to modify **GDIProcessHandleQuota.**
- BUG ID 13394 - Using the **Select Items** tool with the **Facing** set to **Front** to select and hide a portion of faces causes some of the faces within the selection to remain visible when zoomed in.

  To catch all selected faces, either:

  - select **Facing > Through** instead of **Front**,

    OR
  - zoom in closer to the object.
- BUG ID 13481 - Making a selection using the **Marquee Select Tool** with the selection mode set to **Magic Wand** sometimes ignores regions that are completely surrounded by other selected regions, despite differences in color.
- BUG ID 13571 - Launching a new version of Mari for the first time, when a config file exists from a previous version, sometimes results in an object not appearing in the Ortho view.

  To solve this, close Mari, delete the following config file and relaunch Mari:

  - Linux: **~/.config/TheFoundry/Mari.conf**
  - Windows: **C:/Users/<login>/.mari/TheFoundry/Mari.ini**
- BUG ID 13640 - The **Blur** tool can be slow to use on the initial stroke.

  Wait for Mari to process the blur before applying a second stroke.
- BUG ID 13700 - Adjusting the **Camera > Perspective** settings for a **Projector** is not reflected on the canvas until the **Projector** is made Current.
- BUG ID 14201 - Linux only: Mari becomes unresponsive after the system is woken from sleep.
- BUG ID 14985 - There may be a slight pause after importing textures when creating new projects as Mari saves the project.
- BUG ID 15491 - Python: PythonQt bindings are missing for some enumerations and types such as QFileSystemModel. For enumerations, you can usually work around the problem by passing in the integer value instead.

- BUG ID 15810 - The black borders at the edge of the canvas in **Perspective** view are selected, if an object overlaps the borders when a selection is made using the **Marquee Select** tool, with the selection mode set to **Magic Wand**.
- BUG ID 16324 - Windows only: you cannot currently import an image into a channel using a relative file path. To work around this, use an absolute path when importing images.
- BUG ID 16616 - Python: PythonQt sometimes crashes when adding temporary objects to layouts, or compound widgets such as QTreeWidget. This is because of incorrect reference counting. To work around the problem, always store a reference to every GUI item until you are sure it is no longer needed.
- BUG ID 17618 - Ptex does not bake properly if the resolution of the face is too small.

  The work around is to increase the resolution of the selected faces you are having problems with.
- BUG ID 17626 - It can take a long time to import very large or very high polygon count Ptex models.

  The work-around is to assign a small uniform face size (1x1 or 2x2) on import, and then increase the resolution of the relevant bits of the model as necessary after loading.
- BUG ID 17690 - The **Tiled** shader module and the **Masked Tiled** shader module do not render correctly on Ptex channels. They are only intended for use with a UV mapped object.
- BUG ID 18457 - Using "Fermi Series" NVIDIA graphics cards with drivers older than version 270 results in various rendering issues when the **Virtual Texture Type** is set to **Half** or **Float**.

  To resolve this, please download and install the latest graphics driver for your card from the NVIDIA website.
- BUG ID 19780 - Nuke<>Mari Bridge: A projector created in **Ortho** view in Mari does not re-project correctly in Nuke.
- BUG ID 19829 - Nuke<>Mari Bridge: Unprojecting a displaced piece of geometry does not project correctly in Nuke.
- BUG ID 20021 - Textures in the canvas intermittently switch between lower and higher resolutions.

  This issue is more likely to occur if your virtual texture resolution is low, and you're working on a complex model with displacement. Possible work-arounds include increasing your virtual texture size, reducing the number of channels Mari has to access at once (for example, by reducing the number of channels required for the current shader), to reduce the patch resolution of patches in the channels used in the shader, or to use a smaller canvas window or monitor.

- BUG ID 20510 - If you find that the startup time for Mari is longer than usual, please check that the LIC files in your RLM licensing data folder do not refer to obsolete server ports. If they do, place them in another directory and restart Mari.
- BUG ID 22099 - The way that Mari converts quad faces to triangular faces is different to other applications. As a result, normal map renders from other applications may not always match.
- BUG ID 23010 - Nuke<>Mari Bridge: If Mari crashes when receiving incoming components from Nuke when the **Virtual Texture Type** is set to **Float**, lower the **Virtual Texture Size** to a value below 8192x8192.
- BUG ID 26055 - A project created before 1.4v3 with a shader module that uses image thumbnails may not recall its images.

  To remedy this, the shader module must be removed and added again.
- BUG ID 26460 - Painting a mask in a Mask Layer Group, sometimes results in unexpected paint results. To prevent this from happening, either:
  - Use a white "color" layer at the bottom of your mask stack. Any layer used over this initial "color" layer should then be fine,

    OR
  - If you want to create a mask in a Mask Layer Group, simply add another layer on your Mask Layer Group instead, and paint white into it to create a mask.
- BUG ID 29386 - When using the **Export for Maya** script, Maya's viewport may incorrectly show some patches as transparent. This can be resolved by selecting **High Quality Rendering** or **Viewport 2.0** from the Renderer menu within Maya.
- BUG ID 31946 - Sometimes paint is not baked because of memory management issues on the graphics card. This can be due to issues such as a high resolution paint buffer, a high bit-depth paint buffer, large virtual texture size, or even a large scale value on the paint buffer transform. These issues can usually be identified by **glError: 0x505 out of memory** messages in the log.

  Try reducing any or all of these values to prevent it happening. Graphics drivers are continually improving, so it's also worth checking whether upgrading your drivers may resolve the problem.
- BUG ID 33293 - Linux: Launching Mari with the language set to one without certain character symbols resulted in Mari failing with an error that the specified transform could not be loaded.

  To work around this, set the locale (language) to English.
- BUG ID 34679 - On extremely large projects, issues can arise with shader limits and reaching the maximum allowed texture slots available.

To avoid reaching these shader limits on large projects, try the following workarounds:

- hide groups and layers, or
- cache groups and layers.

- BUG ID 34690 - Flattening or caching layers or channels on complex projects may cause Windows to reset the graphics driver due to the long processing time. To work around this issue, you can try to flatten or cache fewer layers at a time, or reduce the value of the **Max Render Size For Baking** setting. This setting can be found under **Preferences** > **GPU** > **Baking and Projection**.

  Reducing this size breaks the flattening or caching operation up into smaller pieces, which individually take less time to calculate, and thereby avoids a Windows graphics driver reset.
- BUG ID 34729 - Mari displays a rendering error on the canvas when it is unable to create a shader. More information has been included to help you determine the cause of the error. Some solutions might be to hide groups and layers, or to cache parts of your layer stack until a shader can be created.
- BUG ID 34804 - An infrequent issue on Windows AMD graphics card drivers causes shader baking actions, such as flattening, merging, and caching, to swap around parts of the texture. We're working closely with AMD to resolve the problem.

  If you do find this occurring, increasing the **Max Render Size for Baking** setting under **Preferences** > **GPU** > **Baking and Projection** may help reduce the frequency of this problem.
- BUG ID 34923 - On the **Snapshots** palette, the ability to extract a snapshot has been temporarily disabled to prevent Mari from crashing.

Developer Notes

These are the changes relevant to developers.

New Features

- Added a createDuplicate set of functions for creating a duplicate shader input. However, this doesn't duplicate sharing.
- Added the Python function Projector.setUseShader() to set which shaders to use for unprojection.

Feature Enhancements

- The modular shader system from pre-2.0 versions has been replaced by the layers system. The modular shader interfaces have therefore been removed. Other related actions, such as shader baking and shader copying have also been removed.

- The functions findLayer() and layer() have been added to the LayerStack class in Python.
- Information regarding the old modular shader system in the Python API documentation has been removed.
- The new function GeoEntity.createPtexChannel() has been added.
- When Mari crashes on Windows, it now outputs the stack to the log file.
- BUG ID 32215 - You can now query an object to see whether it is UV or Ptex by using the function mari.geo.current().isPtex().
- BUG ID 33119 - The ability to query the resolution of ImageSets using the Python API has been removed, as it wasn't deemed to still be useful.

  Resolution should instead be queried on the actual images within the ImageSet.
- BUG ID 34139 - New functions LayerStack.exportImages(), LayerStack.exportImagesFlattened(), PaintableLayer.exportImages(), and ProceduralLayer.exportImages() have been added. These functions support exporting files using file name templates, as in the GUI exporter.
- BUG ID 34140 - New functions LayerStack.importImages() and PaintableLayer.importImages() have been added. These functions support importing files using file name templates, as in the GUI importer.
- BUG ID 34196 - The C API now supports adding metadata to objects loaded from custom plug-ins, using the attribute system. Please see the Mari C API documentation for more details.
- BUG ID 34260 - Access to alpha setting using the ChannelInfo function in the Python API has been removed, as the setting no longer exists.
- BUG ID 34268 - The new function mari.app.isProcessing() has been added.
- BUG ID 34740 - The linked patches Python example script - linked_patches.py - has been updated to allow patch linking setups to be copied and pasted between different paintable layers and paintable layer masks.

Bug Fixes

There are no bug fixes relevant to developers in this release.