



MARI

NUKE<>MARI BRIDGE

VERSION 1.4V2

Nuke<>Mari Bridge™. Copyright © 2011 The Foundry Visionmongers Ltd. All Rights Reserved. Use of this user guide and the Mari software is subject to an End User License Agreement (the "EULA"), the terms of which are incorporated herein by reference. This user guide and the Mari software may be used or copied only in accordance with the terms of the EULA. This user guide, the Mari software and all intellectual property rights relating thereto are and shall remain the sole property of The Foundry Visionmongers Ltd. ("The Foundry") and/or The Foundry's licensors.

The EULA can be read in the Mari User Guide Appendices.

The Foundry assumes no responsibility or liability for any errors or inaccuracies that may appear in this user guide and this user guide is subject to change without notice. The content of this user guide is furnished for informational use only.

Except as permitted by the EULA, no part of this user guide may be reproduced, stored in a retrieval system or transmitted, in any form or by any means, electronic, mechanical, recording or otherwise, without the prior written permission of The Foundry. To the extent that the EULA authorizes the making of copies of this user guide, such copies shall be reproduced with all copyright, trademark and other proprietary rights notices included herein. The EULA expressly prohibits any action that could adversely affect the property rights of The Foundry and/or The Foundry's licensors, including, but not limited to, the removal of the following (or any other copyright, trademark or other proprietary rights notice included herein):

Mari™ software © 2011 The Foundry Visionmongers Ltd. All Rights Reserved.

Mari™ is a trademark of The Foundry Visionmongers Ltd.

In addition to those names set forth on this page, the names of other actual companies and products mentioned in this Getting Started Guide (including, but not the to, those set forth below) may be the trademarks or service marks, or registered trademarks or service marks, of their respective owners in the United States and/or other countries. No association with any company or product is intended or inferred by the mention of its name in this Nuke<>Mari Bridge.

Linux ® is a registered trademark of Linus Torvalds.

Mari software engineering: Jack Greasley, Kiyoyuki Nakagaki, Marcus Shoo, Kevin Atkinson, Tim Ebling, Jed Soane, Daniel Lond, Robert Fanner, Duncan Hopkins, Mark Final, Chris Bevan and Carl Rand
Product testing: Michael Zannetou, Mark Titchener, Robert Elphick and Antoni Kujawa
Writing and layout design: Jack Elder, Jon Hertzog, Eija Närvänen and Charles Quinn
Proof reading: Jack Elder and Eija Närvänen

Mari includes Disney technology licensed from Walt Disney Animation Studios.

The Foundry
6th Floor, The Communications Building,
48 Leicester Square,
London
WC2H 7LT

Rev: December 7, 2011

PREFACE

Mari is a creative texture-painting tool that can handle extremely complex or texture heavy projects. It was developed at Weta Digital and has been used on films such as District 9, The Day the Earth Stood Still, The Lovely Bones, and Avatar.

The name Mari comes from the Swahili 'Maridadi', meaning 'beautiful' and carrying connotations of 'usefulness'.

About this Manual

This manual provides you with the basic information you need to get started using Mari and Nuke to fix, enhance or create projection setups.

For further information on Mari and its functions, see the accompanying Mari *User Guide* and Mari *Reference Guide*.

Contact Customer Support

Should questions arise that this manual fails to address, you can contact Customer Support directly via e-mail at support@thefoundry.co.uk or via telephone to our London office on +44 (0)20 7968 6828 or to our Los Angeles office on (310) 399 4555 during office hours.

NUKE<>MARI BRIDGE

Introduction

Typically, visual effects involves creating digital environments, adding generated imagery and matte paintings to live action scenes, and creating clean plates by removing blemishes, wires, or rigs from existing footage. Building shots using a combination of these techniques can be a laborious task, especially if there is no effective bridge between applications.

Using the 2D and 3D environment in Nuke has helped speed up this process, but when composite projections need to be extended, blended, or recreated it becomes increasingly complex. Some projections may need to be warped to match other projections, or new projections may need to be created to fill in areas that are completely occluded by objects that have been removed from entire shots. Even when sufficient projection information exists, multiple projections may need to be blended together using paint tools to create a seamless composite.

A typical example of this is with matte painting. Once completed, a matte painting is often imported into a 3D system and projected onto a scenic model (such as a building or landscape). Unfortunately, as the 3D model stretches the painting, the projection only works well from the angle of the projection camera. Fixing this in a 3D system requires the creation of new projections where the stretching occurs and in turn, blending these projections with the original painting for every frame in a shot. Creating new projections in this manner and manipulating them on the fly is both technically and artistically difficult to achieve.

The Nuke<>Mari Bridge offers you a quick and easy way for fixing projection problems, converting projection textures to UV textures, and editing multiple projections in context using the rich and natural toolset of Mari.

Once installed, it's as simple as sending one or more **Project3D** nodes from Nuke to Mari (transferring all of the models, textures and projections in a shot) and using the Mari toolset to fix, enhance, or create projection setups. Once the work is finished in Mari, projection setups are sent back to Nuke where they appear immediately in the node tree.

The Nuke<>Mari Bridge handles all of the data management in the background and links both applications whether they are on the same machine, or on different machines on the same network.

Installation

To install the Nuke<>Mari Bridge, follow these steps:

1. Navigate to the following subdirectory of the Mari application directory:
 - **Media/Scripts/Nuke** (on Linux), or
 - **Bundle\Media\Scripts\Nuke** (on Windows).

Note *If you have any previous version of the Nuke<>Mari Bridge scripts installed, please remove them first.*

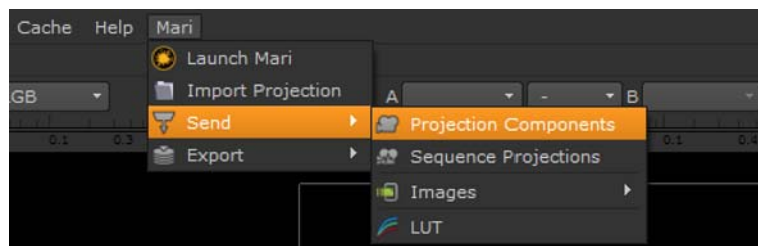
2. Copy the contents of this folder to the following directory:
 - **/home/login name/.nuke** (on Linux)
 - **\Users\login name\.nuke** (on Windows)
 - **/Users/login name/.nuke** (on Mac OS X)

Tip *These files can be installed anywhere in your NUKE_PATH, however the .nuke folder is convenient for most users. If multiple copies of the scripts are installed in multiple folders in your NUKE_PATH, you may encounter problems. Please ensure that only one copy of each script is present.*

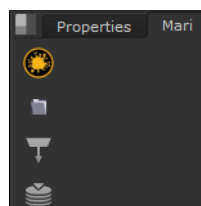
Note *If the .nuke folder already contains a file called menu.py, please open this file and paste the text from the menu.py file (from the subdirectory mentioned in step 1) anywhere into this file.*

Once this has been done, the following items appear in the Nuke UI when launched:

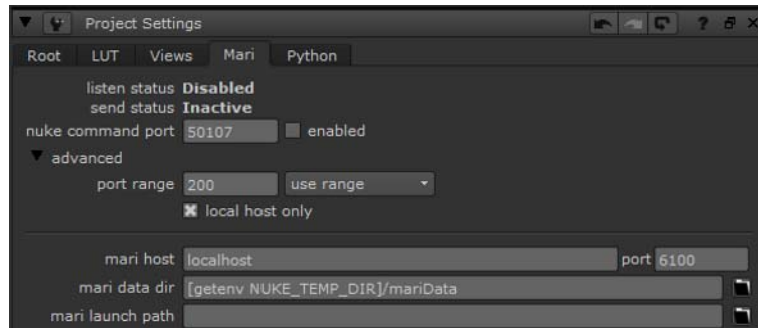
- A **Mari** menu item:



- A **Mari** tab in the **Content** menu under **Pane**:

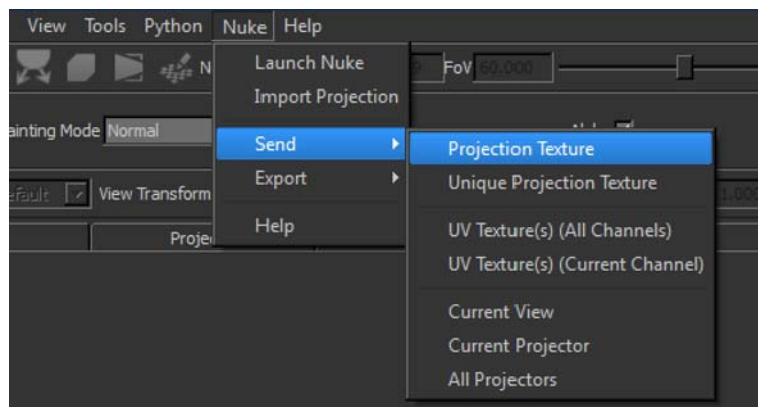


- A **Mari** tab under **Edit > Project Settings**:



In addition, the following item is now available in the Mari UI:

- A **Nuke** menu item:



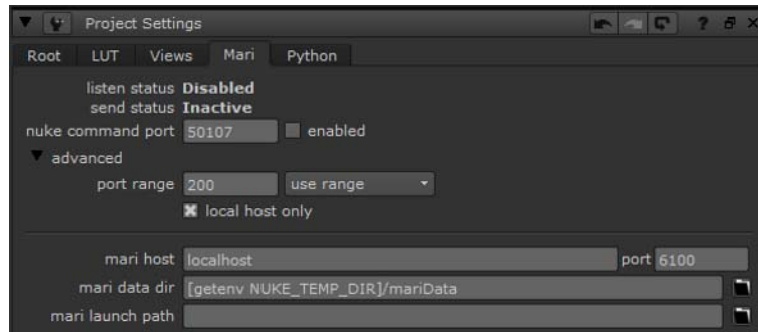
Connection Setup

Before we can send a projection to Mari, we need to establish a connection between the two applications.

Nuke

When Mari is launched from inside Nuke or a command is sent to Mari via the **Mari** menu or toolbar, Nuke's command port is automatically enabled. To manually set this, or adjust other settings related to the **Nuke<>Mari Bridge**:

1. Click **Edit > Project Settings > Mari**.



The Project Settings tab for Mari appears.

2. Check **enabled** and set the **command port**. When enabled, Nuke listens to Mari on the specified command port.

Note *Only one Nuke session can listen on the given port number. If the port number is already in use, Nuke attempts to listen on another port number within a range starting at the given port number. To change this behavior, click **Edit > Project Settings > Mari > advanced** and adjust both the **port range** and **use range** settings.*

3. If both Nuke and Mari are running on the same computer, you should leave the **mari host** field blank, and ensure **local host only** is checked. If they are running on separate machines, enter the machine name or IP address of the computer running Mari here and uncheck **local host only**. These settings are automatically set when a command is successfully sent from Mari to Nuke.

Note *Checking **local host only** tells the application to only listen for connections from the local machine. This can prevent accidental connections from different machines and is therefore more secure, but this must be turned off when you want to use the Nuke<>Mari workflow across the network.*

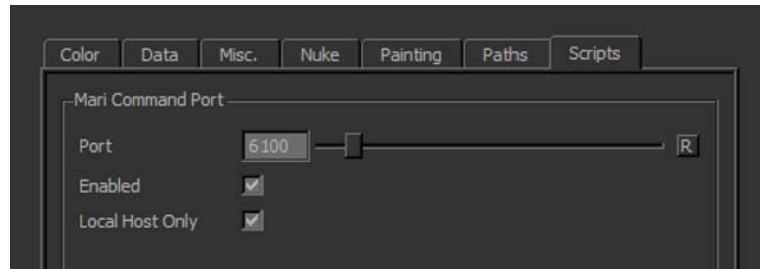
4. Enter the port number that Mari is set to receive commands on in the **port** field. This number should match the Mari **Command Port** number set in the **Scripts** section of the **Mari Preferences** dialog. This value should only be changed if the default is already in use by another application. This is automatically set when a command is sent from Mari to Nuke.
5. Set the **mari data dir**. This field specifies the location of the directory used for the data rendered and exported from Nuke and Mari.

Note *The settings for connecting Nuke to Mari are stored separately in each script.*

Mari

In Mari, there are a number of settings that we need to check before we can receive commands from Nuke.

1. Under **Edit > Preferences > Scripts** ensure that the **Mari Command Port** is **Enabled**. The default **Port** number should only be changed if it is already in use by another application. This number should match the port number set in Nuke under **Project Settings > Mari**.



2. As with the Nuke connection setup, if both Nuke and Mari are running on the same computer, ensure **local host only** is checked. If they are running on separate machines, uncheck this. Refer to the [Note](#) in the previous section for more information.
3. In the **Mari Preferences** dialog, click on the **Nuke** tab.
4. Enter the machine name or IP address of the computer running Nuke in the **Nuke host** field. If both Mari and Nuke are running on the same computer, you should leave this field blank. This is automatically set when a command is sent from Nuke to Mari.
5. Enter the port number that Nuke is set to receive commands on in the **Port** field. This number should match the Nuke **command port** set in Nuke under **Project Settings > Mari**. This value should only be changed if the default is already in use by another application. This is automatically set when a command is sent from Mari to Nuke.

You are now ready to exchange data between Nuke and Mari.

Workflow

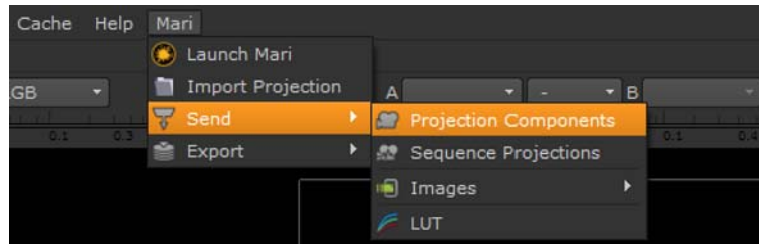
Sending a Projection Setup

To send a Nuke projection setup to Mari follow these steps:

1. Select the **Project3D** node and the **Geometry** node that the projection is applied to.

Note *If you want to merge the Nuke projection with a Mari project that is currently open, and the required geometry is already part of that project, you don't need to select the Geometry node.*

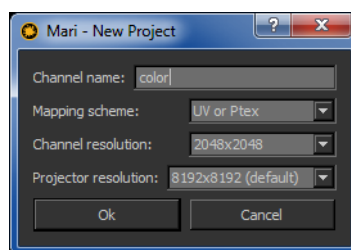
2. If Mari is not already running, select **Mari > Launch Mari** or click on the Mari icon in the **Mari** panel to launch it.
3. In Nuke, select **Mari > Send > Projection Components** or click on the Send icon in the **Mari** panel.



This pre-renders the texture to the location specified in **mari data dir**. The name of the pre-rendered file is derived from the node name and the current frame. If you sent the texture to Mari before, you are asked if you want to overwrite the file. If you choose **No** the existing file is used.

4. Mari now asks if you want to bake the incoming projections. If you plan on modifying the paint buffer before baking the texture (for example, if you want to use the **Spline Warper** tool first), choose **No**, otherwise choose **Yes**.
5. If there is no project open in Mari, you must create a new project and channel for your projection. You are prompted to enter a **Channel Name**, set the **Mapping Scheme**, provide a **Projection Size** and finally, choose the texture **Resolution** for the new channel.

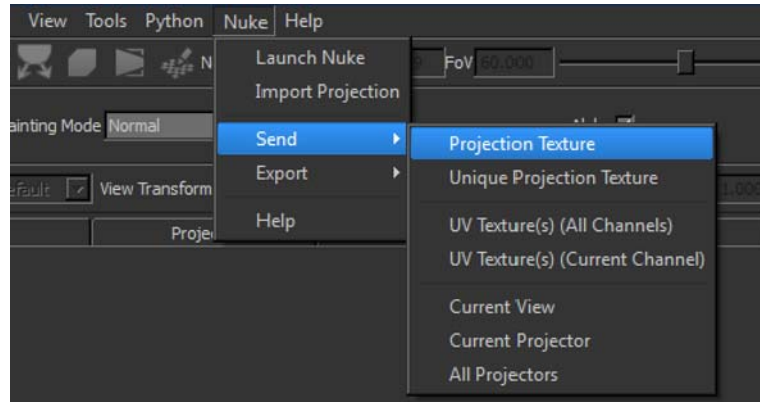
Note *Ensure that you don't compromise the incoming image's resolution by choosing a texture size that is too low.*



If you already have a project open, you are asked if you want to apply the incoming projection to the current project or open a new project.

6. If you chose not to bake the texture, you can now modify it in the paint buffer. The image is also imported into the **Image Manager** to keep track of it.

7. Once you have finished your texture work in Mari and have baked the texture, you have several options to send the data back to Nuke:



- **Projection Texture** — This unprojects the texture through the current projector. The texture is rendered to the Mari data directory and the filename is based on the name of the projector that it is unprojected from.

Once the texture has been written to disk, Nuke creates a **Read** node. This **Read** node should be connected to the **Project3D** node, replacing the original texture with the new texture from Mari.

- **Unique Projection Texture** — Like **Projection Texture**, this unprojects the texture through the current projector. It saves the unprojected texture with a unique file name to ensure that it does not overwrite (or be overwritten by) any other version of the texture that is sent to Nuke.
- **UV Texture(s)** — This exports a set of UV textures to the Mari data directory for the **Current Channel** or for **All Channels** in the project. The filename is based on the object (which in this case is the original **Geometry** node that you sent to Mari) and the channel name. Once the texture has been written to disk, Nuke creates a **Read** node. Since this is a UV texture and not a projection texture, you need to connect it directly to an **ApplyMaterial** node instead of the **Project3D** node.
- **Current View** — This creates a projector of the current view in Mari (Perspective View only) and sends it to Nuke along with the unprojected texture. In addition to the **Read** node, an additional **Camera** node is created that represents Mari's canvas at the time of the export.
- **Current Projector** — This sends the current projector in Mari's **Projectors** palette to Nuke along with the unprojected texture. In addition to the **Read** node, an additional **Camera** node is created that represents Mari's canvas at the time of the export.

- **All Projectors** — This sends all projectors in the **Projectors** palette and their unprojected textures to Nuke along with their unprojected textures. In addition to the **Read** node, a new **Camera** node is created that represents Mari's canvas at the time of the export.

Sending a Sequence Projection Setup

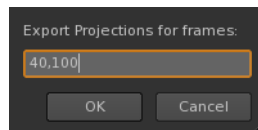
In addition to sending a single projection setup, Nuke can also export multiple snapshots of an animated projection to Mari. This can be used for painting clean plates by combining multiple frames of an image sequence and projecting them through the match move camera at their respective frames.

This workflow is the same as **Sending a Projection Setup**, with the exception of a **Frame Range** dialog that asks for the frames that the snapshots should be taken from.

Note *When planning to return a UV texture from Mari for a geometry node, you should uncheck the geometry node's **Image Aspect** checkbox at this stage. If you don't, the image size will change when plugging in the UV map later on, causing a line up error.*

To send a Nuke Sequence Projection Setup to Mari follow these steps:

1. Select the **Project3D** node and the **Geometry** node that the projection is applied to.
2. Click **Mari > Send > Sequence Projections**.
3. Enter the frame numbers in the **Frame Range** dialog.



4. Mari now asks if you want to bake the incoming projections. Choose **Yes** to bake the incoming projections.
5. Like before, if there is no project open in Mari, you must create a new project and channel for your projections. You are prompted to enter a **Channel Name**, set the **Mapping Scheme**, and choose the texture **Resolution** for the new channel.
6. Once your projections have been imported, Mari creates separate projectors for each frame in the sequence.
7. Proceed with the necessary texture work and, following the steps outlined in [Sending a Projection Setup](#), send the data back to Nuke.

Sending Image Nodes

You can also send any number of **Image** nodes directly from Nuke to Mari. To do this, select the **Image** nodes you want to send in Nuke, click **Send > Images** and choose from one of the following options:

- **All Read Nodes** — To send all **Read** nodes in the node graph.
- **Selected Read Nodes** — To send only the selected **Read** nodes in the node graph.
- **Selected Nodes** — To send all **Image** nodes in the node graph.

Sending a Lookup Table (LUT)

To ensure that images in Mari look the same as they do in Nuke, Nuke's current viewer LUT is approximated in a 10-bit 1D LUT and sent to Mari as a viewing LUT. If you change the **viewerProcess** menu in Nuke Viewer, you can update Mari's viewer LUT accordingly by clicking **Mari > Send > LUT**.

Exporting

Both Nuke and Mari offer an **Export** sub menu as an alternative to their **Send** menu.

This lets you send files between the two applications without a socket connection. When projections are exported to a file from either application, simply use the other application's **Import Projection** menu item to import the file.

Note *When exporting files to disk in Nuke, an **.nmb** extension is assigned (for Nuke<>Mari Bridge). When exporting from Mari, an **.mnb** extension is used (for Mari<>Nuke Bridge).*

Using the Nuke<>Mari Bridge over a Network

To use the Nuke<>Mari workflow between different machines on the same network, ensure that both machines have access to the folder where the shared data is stored. This means that the Mari data directory, under **Project Settings > Mari > advanced > mari data dir** in Nuke, should be set to a shared drive that both machines have access to.

Both computers must be able to access the folder with the same path — for example, two Linux machines might access a shared folder called **/mnt/netshare/Temp**. In this case, no further changes need to be made and the workflow should function correctly.

If the two machines access the shared folder through different paths — for example, **/mnt/netshare/Temp** on a Linux machine and

`\\server\netshare\Temp` on a Windows machine — then some adjustments must be made. The simplest way to account for these differences is to set up symbolic links between directories so the paths match. Your system administrator can help you with this.

File paths sent from Nuke on Windows to Mari on Linux or OS X may still need some modification, as the Windows paths generally start with a drive letter such as `C:`, which Mari on a different OS does not understand. An additional Python function, `mari_bridge.filenameFilter`, is available to help translate paths when sending files from Nuke to Mari. You can override this function in Nuke to modify paths as required.