

JEFECORP

JefeCheck

PLAY.PROCESS.SHARE

User's Manual

JEFECORP

JefeCheck User's Manual

Document for JefeCheck Version 1.4.03

© JefeCorp

Mexico City

Porfirio Díaz #39
Mexico City, Mexico 56503
Phone (5255).5563.8686 • Fax 203.555.0168

Los Angeles

1041 North Formosa Avenue
Fairbanks Building, suite 1
West Hollywood, CA 90046
Phone. (323) 850 3553

Foreword

Thank you for buying or evaluating JefeCheck from JefeCorp! This user's manual will help you understand what JefeCheck is all about and have you using it like a pro in no time.

The manual is divided in three sections, for beginners, check out **Section 1: Play**. To use the more advanced features in JefeCheck look at **Section 2: Process** and **Section 3: Share**.

Have Fun!

Daniel Gollás, Developer of JefeCheck

Table of Contents

FOREWORD	I
TABLE OF CONTENTS	I
INTRODUCTION	1
HOW TO READ THIS MANUAL	2
icon key	2
SECTION 1: PLAY	3
Tracks and Viewports	3
Basic User Interface	4
The Load Window	5
Sequence Info Controls.....	5
Range Controls.....	7
Load Time Modification Controls.....	8
Scale	9
Filter	9
Crop.....	9
BitDepth	9
Channels.....	10
Start All and Playlist Buttons.....	11
The Main Window	11
The Control Bar	13
Timeline and Track Controls.....	13
Playback Controls.....	17
Viewport Controls	19
The Viewports Area	22
Viewport Behavior	22
Viewport Interaction	27
The Menu Bar	28
File Menu	29
View Menu	29
Dialogs Menu	30
Help Menu.....	31
Playlists	31
SECTION 2: PROCESS	34
Real Time Modifications vs Load Time Modifications	34
The LUT Manager Window	35
Default LUT.....	36
Visualizing LUTs.....	36

SECTION 1: PLAY

Using LUTs 37

The FX Manager Window 38

The FX Control Window 38

 Common FX Controls..... 39

 Performance..... 40

 FX Control Window Menu 40

SECTION 3: SHARE 42

Remote Sessions 42

The Remote Session Manager 43

 Connecting to a Server..... 43

 Starting a Server..... 44

 Participating in a Remote Session..... 45

 Loading Sequences in a Remote Session..... 45

 Chat 47

 Remote Pointers..... 47

SECTION 4: OTHER STUFF..... 49

Preferences Window 49

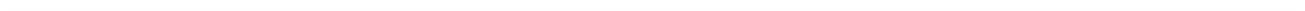
Rendering 53

 The Render Window..... 54

Included FXs 56

Creating custom 1D/3D LUTs..... 58

Creating new FX plug-ins..... 61



Introduction

What is this software for anyway?

JefeCheck is targeted at many different people inside the film and TV industry. Particularly those working with **frame based data**.

The advantages of working with frame based data are many, but a big disadvantage is the fact that there is no straightforward way of playing them back in sequence in their native format. A jpeg for example has no idea that it belongs to a series of files.

That is where image sequence players come into the game. But JefeCheck is a lot more than just an image sequence player!

JefeCheck has three big features, that we call **Play**, **Process** and **Share**.

Play: The Core of JefeCheck. Load up an image sequence and **play it back at full resolution**. Load more than one, load up to four. Look at them, compare them, move them around, look closer, play them faster, play them slower, scrutinize every frame to make sure you are looking at exactly what you expected to see.

Process: The first thing that sets JefeCheck apart from the rest. Process your images (non destructively of course) in any way you can think of with a powerful, easy to use plug-in system based on a well known shading language. Apply 3D Lookup Tables, adjust brightness, contrast, saturation and gamma, composite one sequence over another, do a preview green screen extraction and composite over a background sequence, apply primary color correction, create difference mattes etc.... **All in real time and in full resolution!**

Share: So you're having fun playing back and modifying your images, but playing alone is no fun. Fortunately **JefeCheck is multiplayer!** Start a remote session with any other JefeCheck enabled computer over a local area network or over the Internet² and play together. It's like being in your own virtual screening room. Everything you do is mirrored on their JefeCheck and vice versa. Text chat, plug-in and Lookup Table sharing, remote pointers and other tools help you show everybody in the remote session exactly what you want them to know.

¹ All participants in a remote session need to have the images stored locally, no image information travels over the Internet, so all your information is safe.

² At least one of the participants over an Internet remote session must have a public IP address.

SECTION 1: PLAY

What JefeCheck is NOT:

JefeCheck offers a lot of image processing power to alter an image in real time, but JefeCheck is not a finishing tool. Although you can use JefeCheck to review the final output of your work, you should not rely on its image processing features to complete your work. Although it is indeed possible to render out whatever you see on the screen, you should only use these tools as means to preview your work. For example, you can do primary color correction on your frames to get an idea of changes that you would like to make using programs that you might not have immediate access to.

Using the remote collaboration capabilities of JefeCheck, someone a long distance from you might be able to give you an idea of changes that they would like to make to your sequences, but for final approval of such delicate subjects such as color correction, there is really no comparison to having a real live meeting where both people are looking at the exact same monitor with the exact same color temperature etc.

How to read this manual

This manual is written in a way that you can read it cover to cover, but if you need to know about something in particular you can just jump to any of the three sections (Play, Process or Share).

It is recommended that you understand the sections in order, since each one assumes you are familiar with the terms described in the previous one.

ICON KEY	
	Summary
	Shortcuts
	Power Tip

You will also see some paragraphs that contain special information, identified with a special icon. These paragraphs usually give extra tips or a short summary of what was just discussed. The icon key on the left shows what each icon means.

There is also an index if you need to look up a particular concept.

Finally, there is an appendix, which shows you how to build custom plug-ins and LUTs for JefeCheck.



SECTION 1: PLAY

In this section you will learn the basics of JefeCheck.

At its core JefeCheck is an image sequence player, that is, it allows you to load a bunch of images, named in a sequenced fashion (image.0001.jpg, image.0002.jpg ... etc) and play them back at a given frame rate.

JefeCheck allows you to load up to **four different** image sequences, and then watch them on screen in several different ways.

Tracks and Viewports

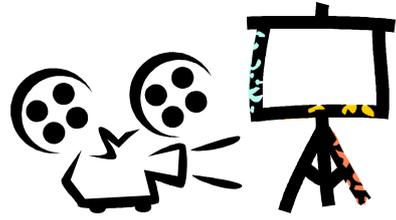
There are two basic concepts you need to grasp to use JefeCheck: Tracks and Viewports. Tracks are where you load the sequences, and Viewports are where the images are actually drawn for you to see.

Although they are closely related, Tracks and Viewports are independent of each other and allow a great deal of flexibility. In a movie theater analogy, you can think of a Track as an actual film reel and a Viewport as the combination of projector and screen: A reel will always contain the same images, but depending on variables like the size of the screen or the lens on the projector, you will see the image in different ways. In JefeCheck, a Track can be seen in many different ways, simply by adjusting the properties of the Viewport on which you are looking it at. You can even see the same track on multiple viewports with different characteristics at the same time.

SECTION 1: PLAY



Track



Viewport

You can load up to 4 different tracks at once (Tracks A, B, C and D), and then watch them on up to four different viewports (Viewport 1, 2, 3 and 4).



So to sum it up...

You load sequences into tracks, and then you watch those tracks on one or more viewports.

The most basic way of using JefeCheck is to simply see one Track on one Viewport without changing anything else. To do this, let's first get acquainted with the basic user interface.



Basic User Interface

When you first open JefeCheck, you will see the **Main Window** and the **Load Window** on top. If you don't see the Load Window, you can bring it up by pressing the shortcut *Ctrl-L* (L is for load). As you might have guessed, the Load Window is used to load the image sequences into tracks.

SECTION 1: PLAY



Figure 1 The Load Window

The Load Window

The load window is shown on Figure 1.

The load window is divided into 4 sections. These four sections represent the four tracks into which you can load image sequences. You don't have to use all four, you can use as many as you want. Each section is exactly the same, so let's analyze what each control in the section does.

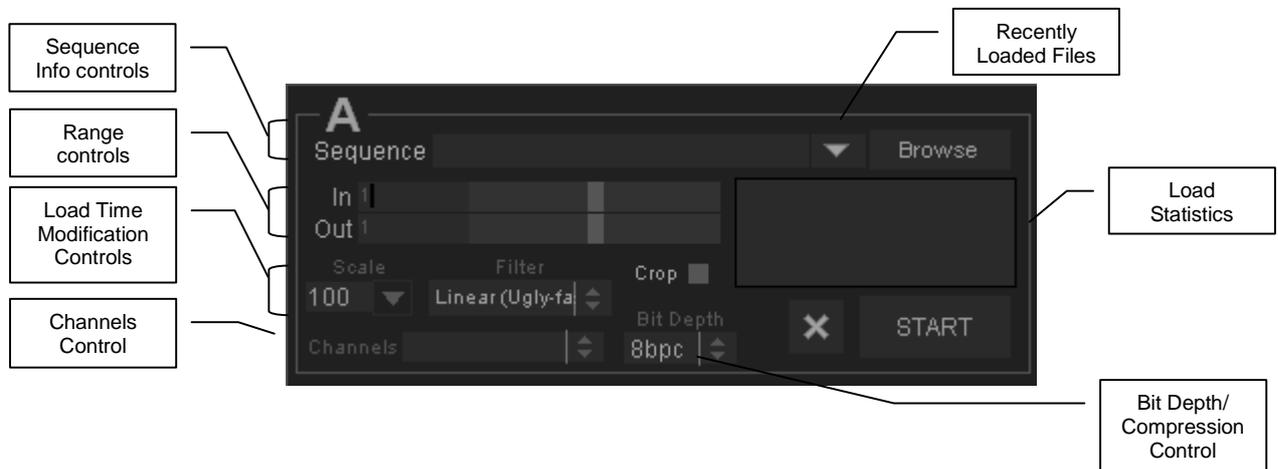


Figure 2 "Loading Parameters"

Sequence Info Controls

The Sequence Info Controls let you select what sequence you want to load. When you press the **Browse Button** a file choosing dialog appears as shown in Figure 3.

SECTION 1: PLAY

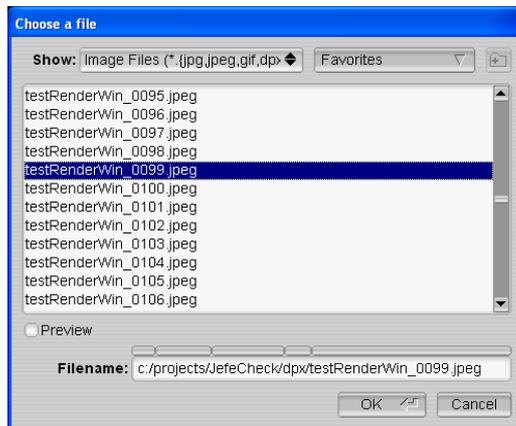


Figure 3 The file chooser dialog

From within the file chooser, you can navigate through all the folders and drives attached to your computer. To pick a sequence simply select any file that belongs to that sequence and click OK.

You can filter out the kinds of files that are shown by selecting an item in the **Show** selection box.

The bottom text box labeled **Filename** shows the current path to the selected file, and also serves as a navigation tool,

click on the box above any folder in the path to go directly to that folder. Finally, you can type the path in the filename box, and use the autocomplete function by pressing the tab key.



You can also drag and drop any frame in the sequence from your Windows Explorer, Mac Finder or Linux file browser onto the Main Window while the Load Window is open to load a preview. If the load window is closed when you drag a file on to the viewport, the loading of that sequence will start immediately. If you shift-drag a file, it will load the sequence at 50% resolution.

If you drag a folder, the first sequence that is found will be loaded.



When working on a project, you will probably be visiting the same folder many times a day. JefeCheck lets you store **Favorites**, which are paths to folders you use frequently. To add the current folder to your favorites click on the Favorites selection box and choose **Add to Favorites**. Now that folder will be available from the Favorites selection box and you can navigate directly to it simply by choosing it. You can also manage your favorites by clicking on the Favorites selection box and choosing **Manage Favorites**. You can also set the default browsing directory in the preferences window.

Once you have selected the file you want and click OK the file chooser dialog closes and the load window is populated with information relevant to the sequence you have chosen as shown in Figure 4. The statistics box shows how much RAM memory will be used up by each frame and by the whole sequence, as well as an estimated load time based on how long the preview frame took to load.

SECTION 1: PLAY

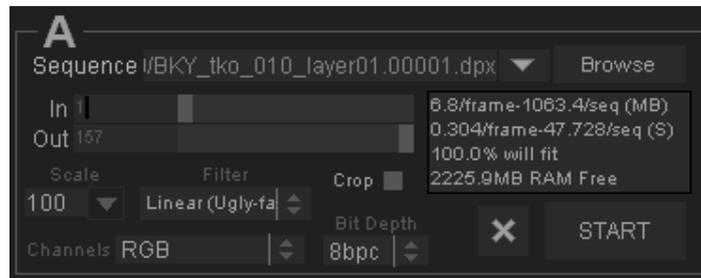


Figure 4 Populated Sequence Options



The most recent sequences you load will be available from the **Recently Loaded** drop down menu to the left of the browse button.

Even though you are not familiar with the Main Window yet, you will see that the image you selected will be loaded and previewed on it, right behind the Load Window.



You can get a better look at the image behind by moving the Load Window out of the way. You can **click and drag on any gray part of the window to move it.**

On top of that preview image you will also see some information on the sequence you are going to load, such as the filename, format, number of channels, bits per channel, Selected frame, Keycode and SMPTE Timecode (on DPX files that contain that information) and how many frames will be loaded as shown in figure Figure 5.

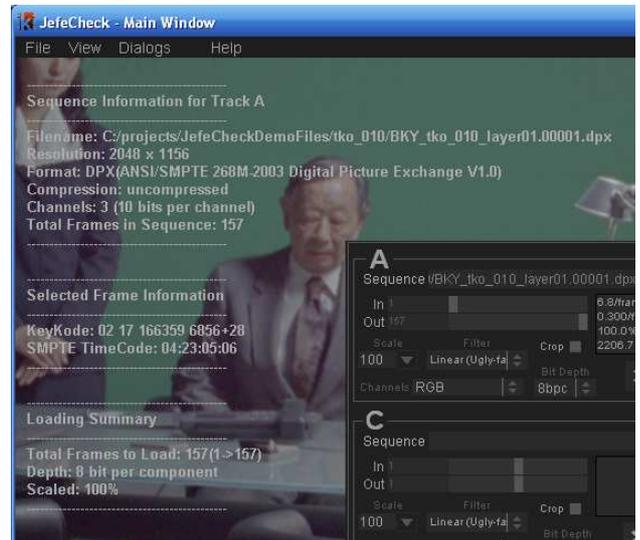


Figure 5 Preview information

At this point, you could press the **START Button** and JefeCheck would begin to load the sequence into the track for you to play.

But before we do that, let's review the other options you have at load time.

Range Controls

The Range Controls let you select what part of the whole sequence you want to load. You simply select **In** and **Out** frames. The numbers you see are directly related to the numbers on the file's name. So if a sequence starts with frame number 153, the

SECTION 1: PLAY

smallest value you could choose for the in frame, would be 153. To select in and out points, you simply drag the sliders on the right side of the numbers.

Load Time Modification Controls

The Load Time Modification controls allow you to alter certain properties of the image sequence as it is loaded into a track. Changes made at load time are “baked” into the track, as opposed to **Real Time Modifications** which are applied “on the fly” as the picture is being drawn on the screen (you will learn about Real Time Modifications later).



Let it be clear that neither Real Time Modifications nor Load Time Modifications change the original file on disk in any way.

A comparison between Load Time Modifications vs. Real Time Modifications is shown in Table 1.

Table 1 Load Time vs. Real Time Modifications

Load Time	Real Time
Loading a scaled image saves memory.	A scaled image uses the same amount of memory as a full resolution one.
Changes are baked in and can't be reversed unless sequence is reloaded.	Changes are done to the sequence on the fly, and can be turned on and modified instantly.
Loading takes longer	Loading is faster
Works well on old hardware	Runs best on newer hardware

Not all Load Time Modifications have a Real Time counterpart, and therefore using one type of modification should not exclude using the other. This will become clearer when we start to use and understand Real Time Modification Controls.

The Load Time modification controls are described below:

SECTION 1: PLAY

Scale

The scale input box allows you to load the images as proxies of the original one. This means that the image is resampled and loaded as a lower resolution version of itself. This saves memory and also helps performance on lower end hardware. You can choose from common options by clicking on the pop down menu (100%, 50%, 25%) or type in any number. When you modify the Scale value, the preview image in the main window changes accordingly.

Filter

This option box lets you choose between two resampling methods for when scaling an image, if you are loading the image at 100% scale, this option is irrelevant. Linear sampling is faster, but produces lower quality images, while Bilinear sampling is slower, but produces smoother images. If you are not that concerned with the quality of the proxies and want faster load times use Linear sampling, otherwise, choose Bilinear.

Crop

Crop is a load time modification that has no Real Time counterpart, and is great for saving memory and improving performance on lower end computers.

Crop allows you to specify a rectangular sub-section of the image you want to load, and only that part will be loaded into memory. When you enable the **Crop Checkbox** a blue overlay square will appear atop the preview image in the main window. Whatever is under this square will be loaded. **To move the Crop square, click and drag it, to resize the square, click and drag the corner squares..**



BitDepth

The bitdepth drop down box allows you to choose different formats to load the image into memory. The different formats have different tradeoffs as listed:

- 8bpc – This loads the image and transforms it to an 8 bit per component image to be displayed on normal monitors. This is the format you will usually use, as it offers a good balance of memory usage and provides perfect image quality unless heavy processing is applied later.
- 16bpc – This image format loads images in their native bit depth and transforms them to 16 bit floating point images. This uses up twice the amount of memory than the normal format, but allows greater flexibility during processing, preventing banding and other color artifacts when applying real time color modifications.

SECTION 1: PLAY

- HALF – This image format is specific to OpenEXR image files. If you select it while loading another image format, the effect will be the same as choosing 16bpc.

Loading OpenEXR images using the HALF bitdepth allows JefeCheck to display the image using its native format (16 bit half float numbers). Loading from OpenEXR files will be much faster as well, since no processing has to be done on the image.

When loading using the HALF format, you will have the entire high dynamic range of the image available for processing. You will usually need to use the Gamma and Exposure controls to view it correctly, since OpenEXR pixels need gamma correction to be viewed directly on an 8bit screen. This FX also allows you to play with different exposure levels in order to see all the details contained in the EXR image.

Loading in HALF format will result in each frame using double the RAM of a normal 8bpc image.

- 4bpc – This format has a limited amount of uses since it loads the image and transforms it into a 4 bits per component image. This introduces serious banding and color inaccuracies for all but the simplest images. The tradeoff is that you can fit twice as many frames into memory as you could with the Normal format. One of the uses of this format is to load as many frames into memory in order to check for black or missing frames in a sequence, a process where image quality is not important.
- S3TC – This loads the image and compresses it in order to fit more frames into memory. Not all hardware supports this format and it is a little slower to load. The compression is not lossless and introduces a small amount of artifacts, so if you are checking for pixel imperfections, this is not the way to go. **On the other hand, this format let's you fit up to eight times more frames into memory and should be used if you wish to view very long sequences and image quality does not have to be absolutely perfect (but still quite good).**

Unless you are having specific problems, or you need to load more frames into memory at the expense of image quality, **the Normal method is the recommended format for most uses (except for EXR images, in which case the HALF format is recommended).**

Channels

This option box lets you choose between any available channels present in the image. This is particularly useful for OpenEXR files. If you open a multichannel OpenEXR image, all the contained channels will be shown here so you can choose which one you wish to load into the track (e.g. Specular, Diffuse, RGBA, Y, Shadows, Depth, etc.).

SECTION 1: PLAY

Start All and Playlist Buttons

Once you have selected what tracks you want to load, adjusted their load time parameters (if needed) and you are sure of what you want to load, you hit the **START Button**. This will start loading the track in the background and you can continue using the load window to load another sequence into another track. If you setup all the tracks and want to start loading them all at the same time, you click the **Start All Button** at the bottom of the Load Window. This closes the Load window and the sequences will start loading into the tracks. If you decide you don't want to do anything, you can simply click the **Done Button** to close the Load Window.



The Add to Playlist button adds the current setup to the playlist. To display the playlist press Ctrl+p.

So to sum it up...



You use the load window to select what tracks you want to load, adjust how you want them to be loaded, and then click the **START** button to start the loading. All parameters have the most common values as default, so loading can be as fast as selecting a sequence and clicking **START**. Remember you can also drag a frame of the sequence onto any viewport in the Main Window to load the sequence.

The Main Window

The main window is where almost everything else in JefeCheck really happens. It is basically divided in three: the Menu Bar, the Viewports Area, and the Control Bar as you can see in **Figure 6**.

SECTION 1: PLAY

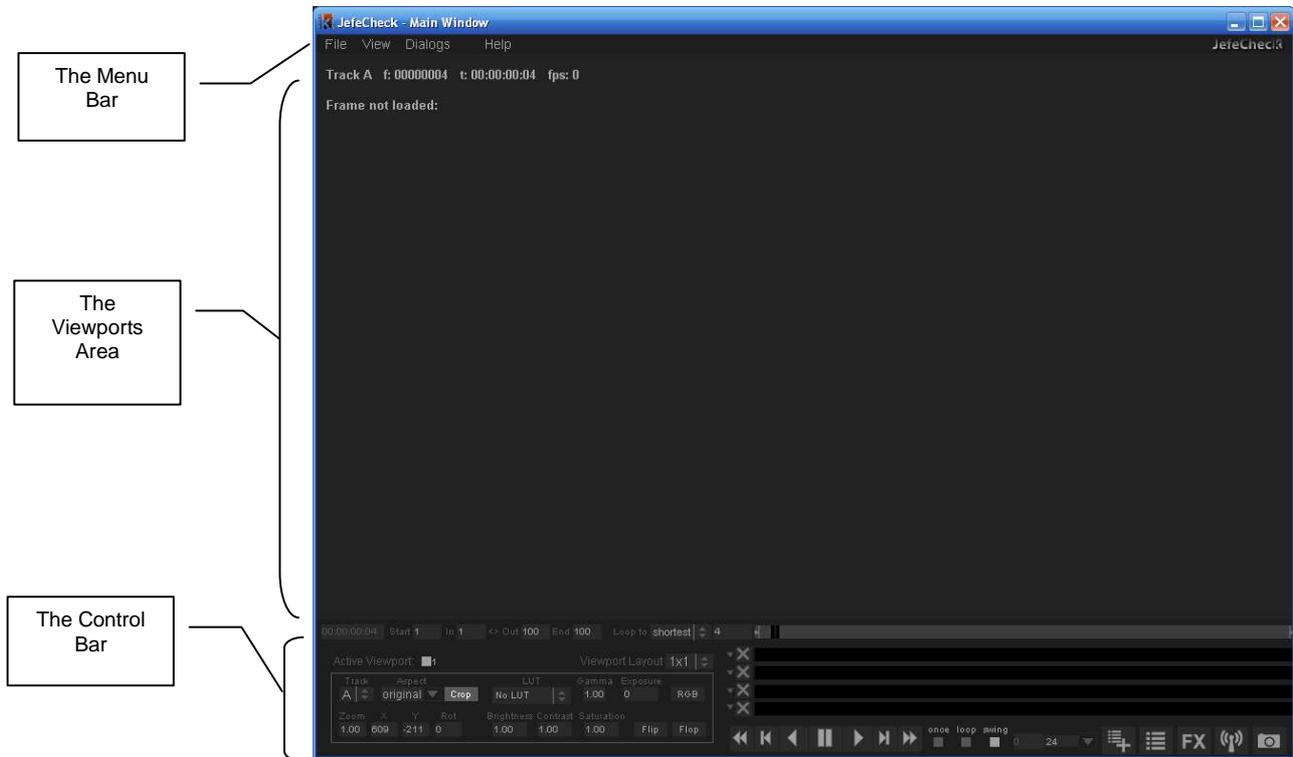


Figure 6 “The Main Window”.

- **The Menu Bar** is where you access other windows or adjust some settings.
- **The Viewports Area** is where the viewports are located and you actually see the image sequences you have loaded.
- **The Control Bar** is where you control almost everything related to the playback of the sequences you are watching and adjust many settings for the viewports.



You can hide the Control and Menu Bars by pressing the **ctr+alt+f** keys. You bring the controls back up by pressing **ctr+alt+f** again.

When you first see the Main Window, it is empty. What is shown in the viewports area varies depending on what windows are open, but this is the area where you will see all of the graphical data that JefeCheck can show.

When the Load Window is open, you will see a preview of the sequences that will be loaded if you click the START Button. When the Load Window is closed, you will see whatever track is loaded and assigned to each particular viewport.

SECTION 1: PLAY

Remember tracks are independent from viewports: tracks hold sequences, and viewports are used to view those sequences. One viewport can show only one sequence, but one sequence can be seen on multiple viewports.

Let's go over each element of the Main Window.

The Control Bar

Figure 7 Shows the Main Window Control Bar, which contains many different controls that can be grouped into three logical areas as shown in Figure 8. **Error! Reference source not found.**



Figure 7 The Main Window Control Bar



Figure 8 The three big logical areas of the control bar

The three big logical areas in the control bar are:

- Timeline and Track controls
- Viewport Controls
- Playback controls

Timeline and Track Controls

The Timeline and Track Controls show you information relevant to time and position of what you are seeing on the screen and they let you control what frames of the loaded sequences you are looking at.



Figure 9 Timeline Controls

The Timeline controls shown in Figure 9 from left to right are as follow:

1. **SMPTE Counter:** The SMPTE Counter shows the current time relative to the timeline starting at frame one, it does not show time code information

SECTION 1: PLAY

embedded in files. The Counter shows time in hours, minutes, seconds and frames in the following way: HH:MM:SS:FF. The amount of frames that have to elapse for the seconds counter to increase depends on the target frame rate, controlled from the Playback Controls (see Playback Controls)

2. **From-To , In/Out Markers:** The **From** and **To** markers specify the range of the timeline. You can force the timeline to playback only from a certain frame and up to another one using the In/Out markers. This is helpful if you only want to see a certain range of frames of the sequence at a time. This is different from only loading a certain range of frames, since you can change the From and To markers on the fly without having to unload or reload any frames. As with all numeric inputs in JefeCheck, you can click and drag to change the value or enter a number with the keyboard. You can also use the shortcuts *i* and *o* to set the in and out points. To reset them use *shift+i* and *shift+o*. Also, if you do *alt+i*, the in point will be set and the sequence will start reloading from that point on. You can also click and drag the In/Out markers directly on the timeline.

Whenever you load a sequence, the From/To, In/Out markers are updated in order to fit the longest loaded sequence.

3. **Loop Priority Control:** When you have multiple tracks loaded, each with a different length, the loop priority determines at what point on the timeline the playback will loop. The options are
 - a. Shortest – The playback loops around the sequence with the shortest number of loaded frames. This could mean not seeing all of the frames in the longer sequences.
 - b. Longest - The playback loops around the sequence with the longest number of loaded frames. This could mean seeing empty frames in the shorter sequences.
 - c. Timeline – The playback loops around the length of the timeline, no matter how many frames are loaded in any track.
4. **Current Frame:** Shows the current frame the timeline is at.
5. **The Timeline:** The timeline allows you to control and graphically visualize what frame of the sequence you are looking at. You can click and drag the knob on the timeline to scrub or jump to any point on it. The size of the knob will vary depending on the range the timeline is showing, defined by the From and To markers.

The Track Controls are shown in Figure 10 along with the Timeline.

SECTION 1: PLAY

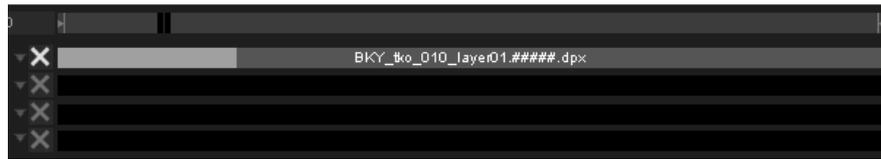


Figure 10 The Track Controls with the Timeline

You can see that there are four rows of controls under the Timeline. These four rows represent the four tracks that can be loaded. Figure 10 shows a single loaded sequence. Each row contains a **Track Options Button**, a **Cancel button** and a **Track Bar**.

1. **The Track Bar:** 

The Track Bar consists of a gray background that shows how long the track is in relation to the timeline and other tracks, inside the gray background, there is a slightly darker gray area, that shows the range of the track we wish to load (you don't always want to load the whole sequence), that gray range area lights up in green to show what frames are loaded in memory and ready to show. While a track is loading, you can playback whatever part is already in memory.

2. **The Cancel Loading Button:** 

The Cancel Loading Button turns red whenever its corresponding track is being loaded. This means that you can cancel the load at any point. When the track is completely loaded or no more frames can fit into memory, the loading stops and the Cancel Loading Button turns gray.

3. **The Track Options Button:** 

When you click on the Track Options Button a Track Options window appears as shown in Figure 11.

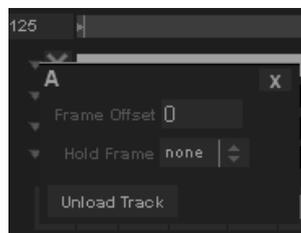


Figure 11 The Track Options Window for track A

The Track Options window shows several options for track behavior.

- **Frame Offset:** Allows you to offset the track a certain amount of frames in respect to the timeline and other tracks. This is useful if you

SECTION 1: PLAY

are comparing two tracks but the segment that you wish to compare is not necessarily in the same frames on both tracks. For example, if you want to compare and simultaneously play back the last 10 frames of Track A and the first 10 frames of Track B you could offset track B enough frames so that the first 10 frames align with the last 10 frames of Track A. Or you could offset track A in the reverse direction to obtain the same result.



You can also change the frame offset by simply clicking and dragging the track.

- **Hold Frame:** Many times, you want to simultaneously play back two sequences of different lengths. The Hold frame specifies what should be displayed if the current frame is outside of the loaded frames for a particular track. For example, if Track A is 25 frames long, and track B is 45 frames long, when the timeline is at frame 30, track B will show its own frame 30, but track A will show an empty frame since there is nothing loaded there. The Hold frame can be one of three values
 - None: No frame will be held. If the timeline goes out of the range of the sequence, no frame will be shown.
 - Current: No matter what value the current timeline frame is, the Track will show the frame position that was current when the **Current** option was selected. So if the timeline value is 5, and you select the Current Frame option, track A will always show frame 5.
 - Edge: Whenever the current timeline frame is out of the range of loaded frames for this track, the one on the closest edge to the actual value will be used, that is, if the timeline is at a later frame, the last loaded frame will be shown, if the timeline is at an earlier frame, the first loaded frame will be shown.
- **Unload Track:** You can unload a particular track in order to free memory or simply because you don't want to use it anymore. When you click this option, the track is unloaded but all the settings are kept in the load window, so the track will be reloaded next time you start a load. If you wish to unload the track and erase the settings in the load window, use the Unload button in the Load Window.

SECTION 1: PLAY



You can also reach the more options menu by right clicking on the track. Remember you can change the offset by dragging the track bar.



Many times a whole sequence won't fit into memory. You can tell JefeCheck that you want to unload the loaded frames and start loading the sequence from a different point by **alt-right mouse button** clicking on the track on the point that you would like to start loading.

If instead of alt-right clicking on the track bar, you do it on the timeline, then all tracks will unload and start loading from that point on.



So to sum it up...

You use the Timeline controls to change timeline range and current timeline frame, which controls what is displayed for each track according to its own settings. You can also use the trackbars and timeline to start loading the frames at a particular point.

Playback Controls

The Playback Controls let you control and decide how the timeline is to behave.



Figure 12 The Playback Controls

An explanation of each control (from left to right) follows:

1. **Rewind:** Returns the current timeline value to its first frame. The keyboard shortcut is **Z**.
2. **One Frame Back:** Turns the current timeline value one frame back. The keyboard shortcut is **x** or **left arrow**.
3. **Play/Pause Reverse:** Starts/Pauses playback in reverse direction. Shortcut is **”**.
4. **Pause:** Starts/Pauses playback in whatever direction we are playing. Shortcut is **Space Bar**.
5. **Play/Pause Forward:** Starts/Pauses playback in forward direction. Shortcut is **”.**

SECTION 1: PLAY

6. **One Frame Forward:** Turns the current timeline value one frame forward. The keyboard shortcut is **c** or **right arrow**.
7. **Fast Forward:** Turns the current timeline value to its last frame. The keyboard shortcut is **v**.



The keyboard shortcuts for the playback controls may seem a little weird, but if you **position your left hand with your thumb on the space bar, and your pinky finger on the z key, you will see that your other fingers will fall right in place with the other playback control keys**, allowing you to start/pause playback, fast forward, rewind, advance and recede frames with very little effort in a very intuitive way, all while leaving your right hand free to control the viewports.

8. **Loop Modes:** The loop modes control what happens when the current timeline value reaches its limit.
 - Once: The sequence plays once and then stops on the last frame. Keyboard shortcut **8**.
 - Loop: The sequence plays in the forward direction until it reaches the last frame at which point it starts over at the first frame, in an infinite loop until it is paused. Keyboard shortcut **9**.
 - Swing: The sequence plays in the forward direction, and when it reaches the last frame, it starts playing in reverse until it reaches the first frame, going back and forwards until it is paused. Keyboard shortcut **0**.
9. **Actual Frame Rate:** This control shows the actual frame rate the sequence is running at. Due to hardware limitations the playback frame rate may not match the target frame rate.
10. **Target Frame rate:** When you playback a sequence, JefeCheck will try to maintain the target frame rate. You can type in any integer value or click on the drop down menu arrow to select commonly used frame rates.

You can also scrub the timeline by clicking and dragging on the viewport while holding the shift key.

SECTION 1: PLAY



So to sum it up...

You use the Playback controls to start/pause the playback, determine the target frame rate, monitor the actual frame rate and select if and how you want the sequence to loop.

Viewport Controls

The viewport controls, shown in Figure 13 allow you to specify and control what viewports will be shown, and what track each viewport will display.



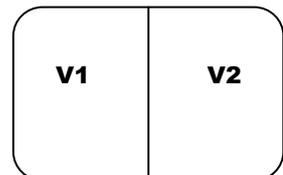
Figure 13 The Viewport Controls showing a 2x2 layout

The listbox on the top right shows the viewport configurations or Layouts. You can also access the configurations from the keyboard or from the View>Viewport Layout menu in the Menu Bar.

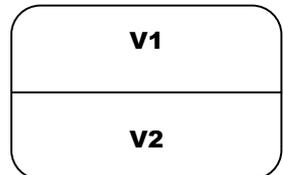
- **1x1:** A single viewport (Viewport 1), using up all the space in the viewport area is shown. Keyboard shortcut **ctrl+1**.



- **2x1:** Two viewports are shown side by side (two columns times one row), each using half the horizontal space and the full vertical space in the viewport area. Keyboard shortcut **ctrl+2**.



- **1x2:** Two viewports are shown one on top of the other (one column times two rows), each using half the vertical space and the full horizontal space in the viewport area. Keyboard shortcut **ctrl+3**



- **2x2:** Four viewports are shown, two side by side and another two under them (two columns times two rows), each using up half the vertical space



SECTION 1: PLAY

and half the horizontal space in the viewport area. Keyboard shortcut **ctrl+4**.

Figure 13 Shows the Viewport controls when all four viewports are showing in a **2x2 layout**. The controls that are shown correspond to the active viewport, to activate a viewport select it in the Active Viewport checkboxes or click on it in the viewport area.

Each control group affects how each viewport is displayed. A description of what each control does follows:

- **Track Selection Box :**



Select which track you want displayed on this viewport. It can be any of the four tracks. If no sequence is loaded in the selected track, the viewport will not show anything.



More than one viewport can show the same Track at the same time, so you could see the whole frame in one track, while looking at a close up of the same frame in another viewport.

- **Aspect Ratio Controls :**



You can change the aspect ratio of the sequence being shown on the screen in real time. The default value for the Aspect Ratio control is **original**, which means that the original aspect ratio of the image being shown will be used to display it. The drop down menu options will show commonly used aspect ratios such as 4:3, 16:9, 2.35:1 etc, **but you can type in any ratio you want, either in the x:y notation, or as a decimal number obtained by dividing x/y.**

By default, when you change the aspect ratio of a viewport, the image will be deformed to adapt to it. This is useful when viewing material filmed with anamorphic lenses. If you don't want to deform the material but still want to change the aspect ratio, you activate **crop button**.

When you activate the Crop Bars control and the Aspect Ratio control is set to anything other than "original", the aspect ratio of the image will be changed not by squashing or stretching the image, but by applying black crop bars to hide or compensate for a portion of the image. The opacity of the black aspect bars can be changed in the View>Aspect Bars opacity menu.

- **Flip/Flop Controls :**



When you activate the flip/flop controls, the viewport will reflect vertically (flip) or horizontally (flop) or both. Shortcuts are **ctrl+8** and **ctrl+9** respectively.

- **RGBA Mask :**



You can see any RGB color channel of the image in the viewport by

SECTION 1: PLAY

clicking the RGB color mask button. Clicking on the button will toggle through each channel. **The shortcuts for turning these controls on/off are r, g, b and a.**

- **Transformation controls:**



Zoom	X	Y	Rot
1.00	-2	-37	0

In JefeCheck you can interactively zoom in and out of a viewport by using the scroll wheel on the mouse, or by using the zoom control. A value of 1 means no zoom. As with other numeric controls in JefeCheck, you can click and drag your mouse to modify the value on it. The rate of change when dragging will depend on what button you click with. **You can also zoom in and out by holding down the ctrl key on the keyboard and left-clicking and dragging on the viewport.**

You can also interactively pan a viewport by left-clicking and dragging the mouse on the viewport, or by using the x/y position controls. X/Y controls with a value of 0 will result in a viewport centered within its allotted viewport area. Finally, you can rotate the frame an arbitrary number of degrees. To reset transformations use **ctrl+R** or **alt+ctrl+R** to reset transformations on all viewports.

- **Color Correction Controls:**

LUT	Gamma	Exposure
No LUT	1.00	0
Brightness Contrast Saturation		
1.00	1.00	1.00
		Flip

You can adjust several color aspects of the viewport with the Color Correction Controls which are applied in the following order:

LUT: Apply a 1D or 3D LUT to the viewport. Loading LUTs is described in part 2. To scroll through the available LUTs click **ctrl+Up/Down**. You can also select a default LUT that will always be applied in the LUT Manager (shortcut **F4**). This default LUT can be overridden by setting the environment variable **JEFECHECK_DEFAULT_LUT** to the name of the desired LUT.

Gamma: Adjust screen gamma. Shortcut **hold W and click drag on the viewport.**

Exposure: Adjust screen exposure. Shortcut **hold E and click drag on the viewport.**

Brightness: Adjust screen Brightness. Shortcut **hold Q and click drag on the viewport.**

Contrast: Adjust screen contrast. Shortcut **hold D and click drag on the viewport.**

Saturation: Adjust screen saturation. Shortcut **hold S and click drag on the viewport.**

To reset the viewport's color corrections click **shift+R**.

SECTION 1: PLAY

If you use a color correction frequently, you can save it as a Color Correction Favorite in one of 5 slots. Click **ctrl+shift+ 1 through 5** to save it. Click **shift+1 through 5** to apply a favorite color correction to the active viewport. Favorite color corrections are saved in the settings file and will be available next time you open JefeCheck.



So to sum it up...

You use the layout controls to change how many viewports are displayed and how they are laid out on the viewports area of the Main window.

You use the viewport controls to change the way each viewport is drawn, changing parameters like the aspect ratio and zoom values and what track is displayed on each viewport.

You use the color correction controls to change the color properties of the viewport. You can save and load up to 5 favorite color corrections.

The Viewports Area

The central area of the Main Window is the Viewports Area. This is where the image sequences and other bits of information are drawn. The viewport area will be divided into up to four sections, depending on the layout that has been chosen (see Viewport Controls on page 19).

The sections are not divided by a visible divider, but you can see that each viewport also has a text overlay that shows some valuable information on what is being shown at the moment.

Viewport Behavior

Viewports show different information depending on what you are doing at the moment. Most of the time you will see images, and sometimes you will see other kinds of information, like the graphical representation of a Lookup Table.

The basic situations where you will see different things on the viewport are the following:

- **With the Load Window Open**

When the load window is open, each one of the visible viewports (remember you can change what viewports are visible by changing the layout, see p.19) shows a preview frame for each sequence that will be loaded into each track. **Each viewport will show the preview for whatever track it is assigned to.**

Additionally, when the Load Window is open, the text overlay on each viewport will show relevant information regarding the sequence that will be

SECTION 1: PLAY

loaded into its corresponding track as shown in Figure 14. The text information you will see is:

- What track the preview is for.
- The filename of the image being displayed, which is also the one from where the sequence of images will be derived
- The image's resolution in pixels (Width x Height)
- The image's format with a description (if available).
- The image's compression method (if any).
- How many channels (components) are in the image.
- How many bits per channel the image has.
- The total number of frames identified in the sequence.
- KeyCode and SMPTE Timecode information for the frame (only available in DPX images that contain such metadata)
- A Loading summary showing the total number of frames that will be loaded, followed by the start and end frame numbers according to the files numbering (The total number of frames to load depends on the In/Out track loading controls, see **Range Controls** p7), the bit depth at which the image will be loaded, and finally the scale at which the sequence will be loaded

SECTION 1: PLAY

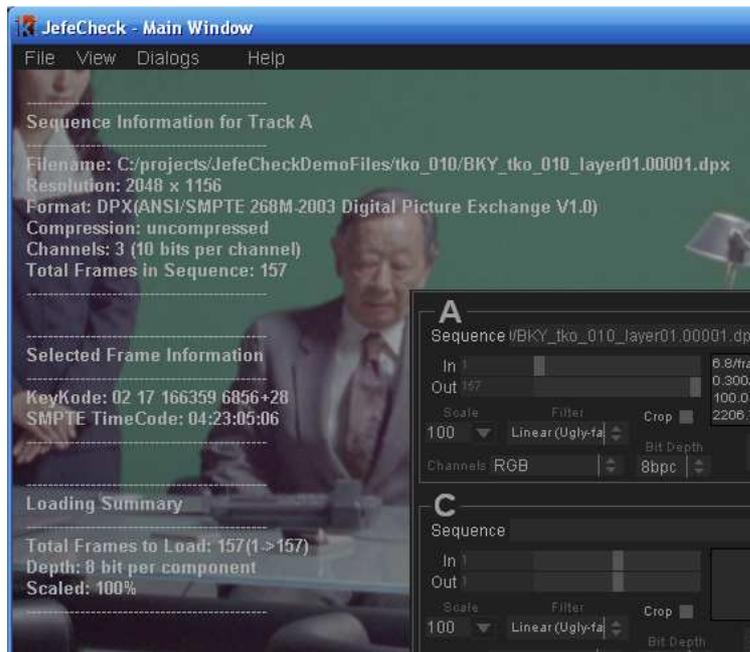


Figure 14 Preview text for track A

- **During Playback**

Once you have sequences loaded into tracks and the Load Window is closed, you will see the frames as they are loaded in the layout that has been selected.



Tracks will be displayed on whatever viewports have them assigned, and a text overlay will show relevant information. The text display can toggle between three different states by pressing the shortcut key **t**. The states are: no display, basic information, and finally basic plus metadata (if the track contains some kind of metadata (DPX, OpenEXR, EXIF)). Pressing **alt+t** will toggle the text display for all viewports.

The information that will be displayed when the text overlay is in basic mode as seen in Figure 15 is as follows:

- The current timeline frame (eg. *f:0000086*)
- The current SMPTE time relative to the first frame of the timeline (eg. *t:00:00:03:14*).
- The current playback framerate (eg. *fps:24.00*)
- The filename of the image being shown.
- The image resolution, the number of channels, and the number of bits per component of the displayed image (eg. *2048x1156x3 10bpc*)

SECTION 1: PLAY



f: 00000086 t: 00:00:03:14 fps:24.00
Z:\dgollas\JefeCheckDemo\Camila\59-1_6_s1.1746479.dpx (2048x1156x3 10bpc)

Figure 15 Playback text overlay on the viewport

When DPX metadata is present and is enabled, the information that will be displayed as seen in Figure 16 is the same as normal frames, plus the following fields:

- Timecode (from the DPX TV Header)
- KeyKode (derived from the DPX Motion Picture Header)
- Filename (from DPX File Information Header)
- Source Filename (from DPX Image Orientation Header)
- Frame ID (from DPX Motion Picture Header)
- Filename (from DPX File Information Header)
- Project (from DPX File Information Header)
- Slate (from DPX Motion Picture Header)
- Creator (from DPX File Information Header)
- Input Device (from DPX Image Orientation Header)
- Input Serial (from DPX Image Orientation Header)
- Input Device (from DPX Image Orientation Header)
- File Size in MB, KB and Bytes (from DPX File Information Header)
- Frame rate (from DPX Motion Picture Header)
- Format (from DPX Motion Picture Header)
- Creation Time (from DPX File Information Header)
- Source Creation Time (from DPX Image Orientation Header)
- Copyright (from DPX File Information Header)

SECTION 1: PLAY

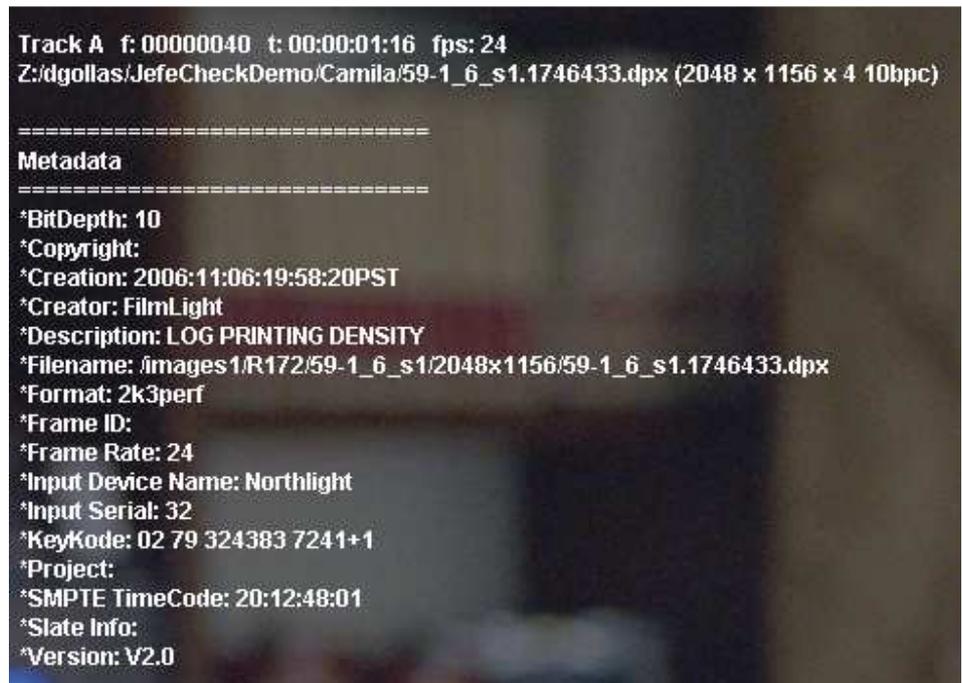


Figure 16 DPX Metadata Text Overlay

For OpenEXR images, the metadata that will be shown is:

- Focus Distance
- Exposure Time
- Lens Aperture
- ISO Speed
- KeyKode
- Timecode
- Location (Longitude, Latitude, Altitude)

For images with EXIF metadata, all available fields will be displayed

- LUT visualization

The viewports are not only useful for displaying images, they can also show a graphical representation of a 1D or 3D Lookup Table. Since we haven't

SECTION 1: PLAY

covered loading LUTs and the LUT Load Window, we will not go into greater depth, but you can get more information in Section 2: Process.

Viewport Interaction

Using the mouse and keyboard, you can interact with the viewports to modify their parameters (see p19) and obtain information about what is being displayed.

A list of what you can do with a viewport using the mouse and keyboard is presented here.



To do this...	...do this
Scrub the timeline	Click and drag to the left or to the right on the viewport while holding the Shift key.
Zoom in and out	Move the mouse wheel up or down. Pressing Shift makes the zoom act 10x slower. Optionally, Left-click and drag up and down while holding the ctrl keyboard key.
Zoom in and out of all viewports at the same time (Gang Zoom)	Zoom in and out like you would for a single viewport, but press the alt keyboard key.
Pan	Left-click and drag on the viewport.
Pan all viewports at the same time (Gang Pan)	Left-click and drag on any viewport while holding down the Alt keyboard key.
Reset the viewport transformations to its default values	Select the viewport you want to reset by clicking on it and then press Ctrl+R
Reset all the viewport's transformations to their default values	Press Ctrl+Alt+R

SECTION 1: PLAY

Adjust Gamma	Left-click and drag on any viewport while holding down the W keyboard key.
Adjust Exposure	Left-click and drag on any viewport while holding down the E keyboard key.
Adjust Brightness	Left-click and drag on any viewport while holding down the Q keyboard key.
Adjust Contrast	Left-click and drag on any viewport while holding down the D keyboard key.
Adjust Saturation	Left-click and drag on any viewport while holding down the S keyboard key.
Reset the viewport color corrections to their default values	Select the viewport you want to reset by clicking on it and then press Shift+R
Reset all the viewport's color corrections to their default values	Press Shift+Alt+R
Show the color value of a pixel on screen (color picker)	Right-Click on the viewport while pressing the ctrl keyboard key.



So to sum it up...

The Viewport Area is where you will see the Viewports displaying all the information contained in the tracks. You can interact with the Viewports by using the mouse and keyboard and display text information on what you are looking at, including DPX, OpenEXR and EXIF metadata when present.

The Menu Bar

At the top of the Main Window is where the Menu Bar lives, and from it you can access some settings that modify the way the program behaves and the viewports are displayed. From there you can also access other windows (including the Load

SECTION 1: PLAY

Window which you already know). Most menu items have a keyboard shortcut that is displayed next to it.



Figure 17 The Menu Bar

We will now break the menus down into their individual sub-items



File Menu

- **Load Manager** (Ctrl+L): Opens the **Load Manager Window**.
- **Playlist Manager** (Ctrl+P): Opens the **Playlist Window**
- **Save Session** (Ctrl+S): Allows you to save a complete JefeCheck session including loaded tracks, load time parameters and viewport configurations into a single file which can later be recalled.
- **Open Session** (Ctrl+O): Opens a complete JefeCheck session. Any sequence settings and loaded tracks will be unloaded and replaced by the information stored in the session file.
- **Save Chat Log**: Saves the complete chat log after a remote session has taken place.
- **Preferences** (Ctrl+P): Opens the Preferences Window. See Section 4: Other Stuff for information about JefeCheck's Preferences.
- **Quit** (Ctrl+Q): Quits JefeCheck.
- **Recent Sessions**: Shows the last few loaded sessions.

View Menu

- **Viewport Layout**: Allows you to change the layout of the viewports (see p19).
- **Zoom Filtering**: When you zoom in and out of a viewport, JefeCheck applies a filter to the displayed image in order to avoid pixilation. You can choose if you want the filter to be a Point Filter (You can see every pixel when you zoom in), or a Bilinear Filter that smoothes out the image.
- **Aspect bar opacity**: Select the opacity for the aspect bars overlaid on the image when the aspect ratio is modified.
- **Fit to viewport** (F): Find the best fit for the current frame in the viewport. Smaller images will not be enlarged, but big images will be zoomed out.
- **Fit all to viewport** (alt+F): Fit frames in all viewports.

SECTION 1: PLAY

- **Flip** (ctrl+8): Flip viewport vertically.
- **Flop** (ctrl+9): Flop viewport horizontally.
- **Flip All**(alt+ctrl+8): Flip all viewports vertically.
- **Flop All**(alt+ctrl+9): Flop all viewports horizontally.
- **Flop** (ctrl+9): Flop viewport horizontally.
- **Reset Current View** (Ctrl+R): Resets the current viewport's transformations.
- **Reset All Views** (Ctrl+Alt+R): Resets all the viewports' transformations.
- **Reset Current Color correction**(shift+R): Resets the current viewport's color corrections.
- **Reset All Color Corrections** (Shift+Alt+R): Resets all the viewports' color corrections.
- **Show Histogram** (Ctrl+H): Shows an RGB histogram for the current viewport. You can drag and resize the histogram within the viewport.
- **Fullscreen** (Ctrl+F): Toggles full screen mode on or off. If you have two monitors, toggling full screen mode do it in the monitor where the JefeCheck main window is located at the moment.
- **Hide Controls** (Ctrl+Alt+F): Hides the control and menu bars. Most keyboard shortcuts will work when in this mode. To return to normal mode, hit **ctr+alt+F**

Dialogs Menu

- **Load Manager** (Ctrl+L): Opens the **Load Manager Window**.
- **Playlist Manager** (Ctrl+P): Opens the **Playlist Window**
- **FX Stack Manager** (F2): Opens the **FX Stack Manager Window**. See Section 2: Processing for more information.
- **FX Manager** (F3): Opens **the FX Manager Window**. See Section 2: Processing for more information.
- **LUT Manager** (F4): Opens the **LUT Manager Window**. See Section 2: Processing for more information.

SECTION 1: PLAY

- **Remote Session Manager (F5):** Opens the **Remote Session Manager Window**. See Section 3: Share for more information.
- **Render Manager (F6):** Opens the **Render Manager Window**. See Section 4: Other Stuff for information about rendering from JefeCheck.

Help Menu

- **Help (F1):** Displays this user manual.
- **Quick Start Guide:** Shows the Quick Start Guide.
- **Online Support:** Takes you to the JefeCheck support page.
- **Video Tutorials:** Takes you to the JefeCheck video tutorials page.
- **Toggle On screen help (H):** Displays/Hides the quick help shortcuts on the viewport area.
- **Check System Requirements:** Displays a window that tells you what JefeCheck features are available on your hardware.
- **About JefeCheck:** Shows the About JefeCheck Window that contains version and contact information and development and testing credits.

Playlists

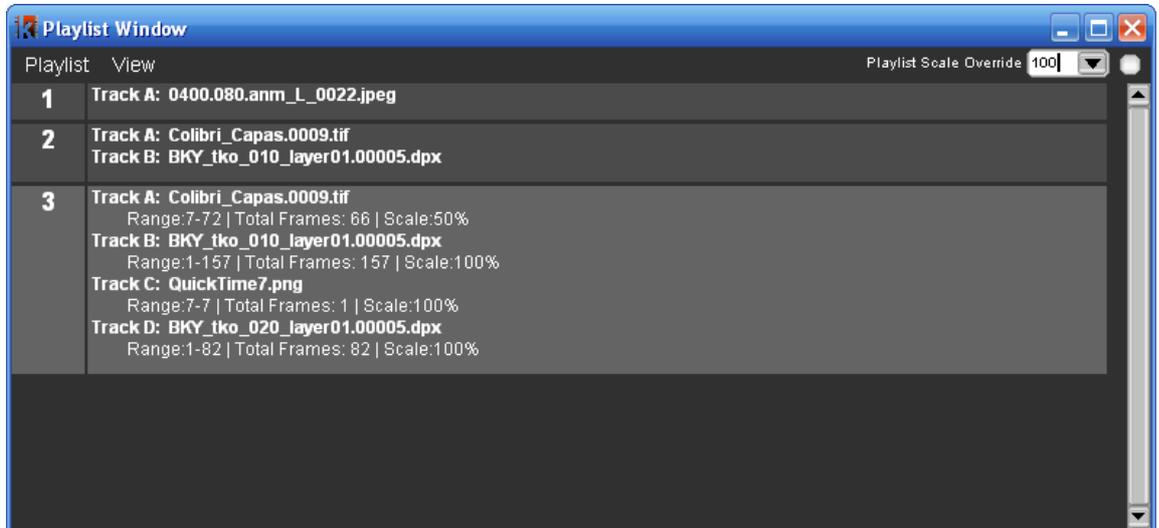
Playlists allow you to save a series of “Playlist Items” and then recall them at will. A playlist item is NOT a single sequence, but instead it holds much more information, including what sequences are loaded in what tracks and with which parameters, the viewport layout, playback settings and any FX stacks applied in your viewport (FXs and FX stacks are explained in the next section). This way, jumping from one setup to another is a snap. This is very different from a MP3 playlist for example, where each item in the playlist represents a single song.

For example, you might be reviewing a series of shots for a show. Say for shot 1 you load the original scans in track A and the completed shot in track B. You apply a blend FX between them and show it in a side by side layout. You then add this to a playlist. All the settings will be recalled whenever you double click on that item, not just what was loaded into track A.

You can do this for all the shots you will be reviewing with a client, adding each setup to the playlist. When the time for review comes, you load the playlist and select what item you want to load.

SECTION 1: PLAY

You can think of a playlist item as a snapshot of the current state of JefeCheck.



To open the playlist window, use the **ctrl+P** shortcut, or use the Dialog Menu.

You add items to a playlist in three different ways.

1. From the Load Window's add to Playlist Button,



2. From the Main Window's Add to Playlist Button.
3. By dragging a single file from the sequence into the Playlist window. If you drag more than one file, each one will be considered a sequence and be loaded into different tracks.

Whatever way you choose to add an item to the playlist, remember that the state of the application (viewport layout, FXs, playback settings) will be saved with the item as well.

You can append tracks to a playlist item by dragging a single file in the sequence on to the playlist item in the Playlist Window. It will be added to the first available track in the item.

To load a playlist item, you double click on it or select it and press the Enter key.

You can also move items by selecting them and pressing **shift+up** or **shift+down** with the keyboard.

SECTION 1: PLAY

Whatever item is selected will be highlighted and extra information will also be shown for it. You can change this behavior using the View Menu in the Playlist window.

Keep in mind that only items loaded from the playlist window will be loaded in remote sessions (See section 3: Share for more information on remote sessions).

Playlists can be saved for later use using the Playlist Menu in the Playlist Window. Just click Save Playlist and choose a path and filename for the Playlist, it will be saved with a jpl extension. To load a playlist, use the Load Playlist menu or drag it into the Playlist Window or the Main Window's viewport area. You can load more than one playlist at a time, and their items will be appended.

Finally, you can clear a playlist by using the Clear Playlist menu. This will remove all items in it.



SECTION 2: PROCESS

In this section you will how to load and use LUTs and JefeCheck FXs.

One of JefeCheck's most powerful features is the ability to process the image sequences you are watching with a variety of Lookup Tables and other effects. JefeCheck uses a simple plug-in system to control how the images are processed.

Real Time Modifications vs Load Time Modifications

As mentioned in Section 1, there are two ways to process an image in JefeCheck, at Load Time and in Real Time. Continuing the film analogy from section 1, tracks are like film reels, and viewports are a combination of projectors and projection screens. You can **think of Load Time Processing as making an different print of the film reel**, maybe you crop the image, maybe you print it at a different size ... the point is, no matter how much you adjust the projector lenses or filters, the images on that different print will be changed, and what you see up on the screen won't go back until you reload the original print on the projector again.

In contrast, **Real Time Modifications are equivalent to simply loading the original film reel on the projector, and then putting different filters or lenses on it.** You don't have to make a different print of the reel in order to see a different effect, you simply swap one filter for another, but the reel remains the same.

SECTION 2: PROCESS

In JefeCheck, Load Time Modifications are done in the Load Window, where you set how you to load the sequence. Loading a sequence and modifying it at load time is a slower process than simply loading the original sequence, the same way it takes time to make a different print of a film reel.

Real Time modifications on the other hand, are applied to the image on the fly, using the power of your computer's graphics card.

JefeCheck can load a variety of Lookup Tables formats but the easiest way to get a 3D LUT is to make your own. It is really easy to emulate any color look by using a custom made JefeCheck LUT. If you can render a single frame through your color pipeline, than you can use the color output of that pipeline in JefeCheck! All you do is process the provided calibration image (JefeCheckLUTSourceImage.tga) through your pipeline, the resulting image (should be the same size as the original) is loaded into JefeCheck and converted to a 3D Lookup table that will emulate whatever color alterations your pipeline made to the original image. See Section 4: Other Stuff to find out how to make your own 3D LUT for JefeCheck.

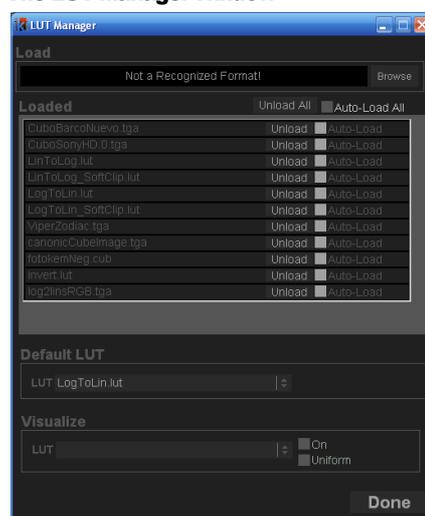
You can also load and use 1D LUTs. The file format for a 1D LUT is a very simple and straight forward text file. To learn how to write your own 1D LUT see Section 4: Other Stuff.

JefeCheck also understands Truelight .cub 3D cube files and has been tested with cubes of size 16 and 32 but should work with any size.

CMS cubes rendered from The Foundry's Nuke are similar to JefeCheck's 3D LUT images and are also supported for cubes of size 16 (these are images with a resolution of 448x448 pixels). You should render out the image as a TGA file with linear color space conversion.

To load a LUT, be it 3D or 1D, you use the LUT Manager.

The LUT Manager Window



From the LUT Manager Window you can load and unload LUTs. To open the LUT Manager Window go to the Menu Bar and from the Image Submenu choose LUT Manager (the notation for that kind of actions will be **Menu>Dialogs>LUT Manager** from now on), or press the **F4 Key**.

The first time you open the LUT manager, it will be empty, so you need to load whatever LUTs you intend to use. To load a LUT you click on the Browse button, this will open a file browser where you can select any of the supported LUT formats that JefeCheck can

SECTION 2: PROCESS

load. You can select as many LUTs as you want at the same time by dragging your mouse or by ctrl-clicking on each one. For more information on the formats that JefeCheck supports, see Section 4: Other Stuff. The LUT browser navigates by default to the FX folder in your JefeCheck installation, and that is where your LUTs should be stored for easy access.

Once you have selected one or more LUT files, click OK in the browser and JefeCheck will start loading it. If the LUT was correctly loaded, the progress bar under the Browse button will turn green and notify you of the success.

The LUT will appear in the Loaded LUTs section of the LUT Manager Window. You don't have to load a specific LUT every time you need it, you can tell JefeCheck to auto-load it for you whenever you open up the program. Simply check the Auto-Load checkbox next to the loaded LUT, and JefeCheck will automatically load it next time.

You can load as many LUTs in as different formats as you want. You unload them by pressing the Delete button next to each one. This only unloads them and does not delete them from disk.

Default LUT

You can have a default LUT that is applied to all viewports whenever you open JefeCheck. You can override this setting by setting the environment variable JEFECHECK_DEFAULT_LUT with the name of the LUT (e.g. LogToLin.lut). Make sure that you use the name of a LUT that has previously been loaded.

Visualizing LUTs

JefeCheck can show you a visual representation of the LUT you have loaded, be them 1D or 3D. To visualize a LUT you click the On checkbox in the Visualization Section of the LUT Manager Window and select the LUT you want to visualize from the **LUT option box**. You will see the LUT drawn on a Viewport in the Main Window. You use the same controls you normally use to pan and zoom a viewport to handle the LUT visualization (see Viewport Interaction p27).

SECTION 2: PROCESS

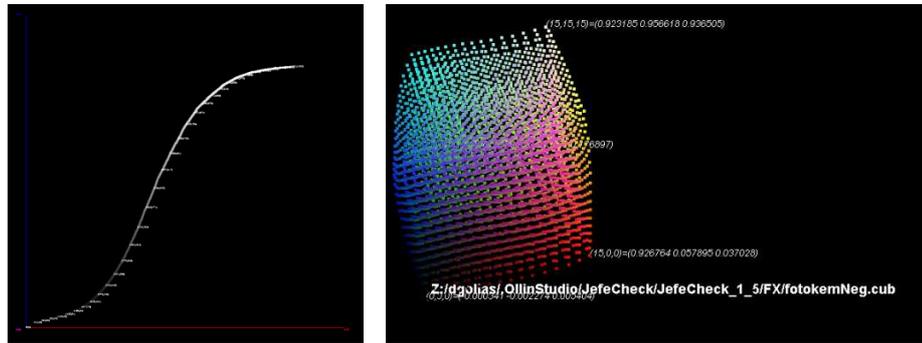


Figure 19 Examples of 1D and 3D LUT visualization

1D LUTs are plotted on an x,y plane, with the input value on the X axis, and the output value on the Y axis. A grayscale value of the output is used to color each point in the plot. A text label of a sub sample of the values is also drawn showing the input and output values.

3D LUTs are displayed as color cubes, placing a colored point at each sample in 3D space. A canonic 3D LUT would generate a perfect color cube, while a modified cube would display other characteristics. Each point in the LUT is positioned in a point in space corresponding to its output value in RGB, and is colored with that same information. This will deform the cube showing distinctive shapes for each LUT in a way that an experienced colorist will be able to predict the kind of color transformation that each cube will yield. If you click the Uniform checkbox, the samples in the cube will be spaced uniformly and form a perfect cube, but the points will still be colored with their output values. This way you can compare two cubes without being distracted by the geometric deformations and only focus on the color changes.

Using LUTs

Of course, the reason you load a LUT is not only to visualize it, but to actually use it to transform the colors in an image.

If your system has a decent video card (decent means that it has enough pixel shading power), you can use Real Time processing to apply LUTs and other effects to the image.

Real Time modifications are called FXs within JefeCheck. Each FX comes in the form of a plug-in that must be loaded in order to be used. JefeCheck comes with two FXs for LUT application, one for 1D LUTs and one for 3D LUTs. See SECTION 4: OTHER STUFF for an explanation of the included FXs.

To load an FX you use the FX Manager in a similar way to how you use the LUT Manager to load LUTs.

SECTION 2: PROCESS

The FX Manager Window



From the FX Manager Window you can load and unload FXs. To open the FX Manager Window go to the Menu Bar and from the Image Submenu choose FX Manager (the notation for that kind of actions will be **Menu>Dialogs>FX Manager** from now on), or press the **F3 Key**.

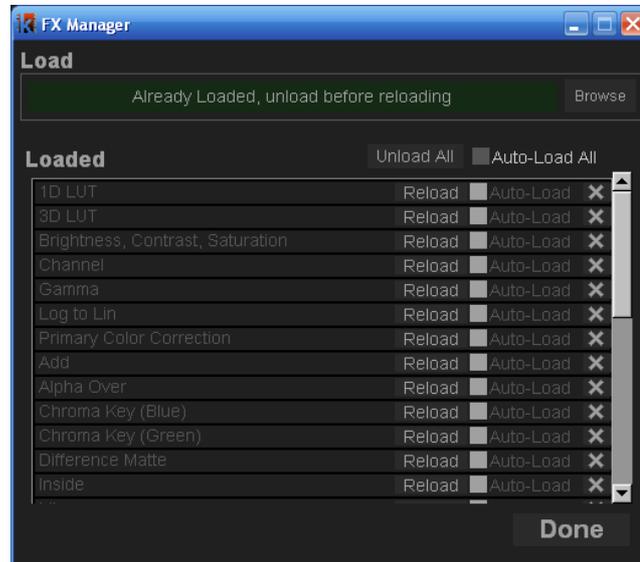


Figure 20 The FX Manager Window

When you first open the FX Manager Window, it will be empty. To load an FX, click the Browse Button. This will open a file browser window in the default FX folder within your JefeCheck installation. You can navigate to other folders if your FXs are stored somewhere else. Select one of the .jfx files and click ok to start loading the FX (you can select more

than one FX by dragging your mouse or ctrl-clicking on each one). If everything goes OK you will see a confirmation message in the progress bar under the Browse Button, the FX will now be loaded and displayed in the Loaded section of the FX Manager. Load as many FXs as you want. **You don't have to load the FXs by hand each time you open JefeCheck, to have the FXs load automatically check the Auto-Load Checkbox next to each one to have them load next time you use JefeCheck**, as shown in Figure 20.. To unload an FX click on its unload button. Once an FX is loaded it can be used on a Viewport to modify the image.



The FX Control Window



You use FXs by applying them to viewports, a viewport can have no FX applied or as many as you graphics card performance will allow you while maintaining a desired playback frame rate. **To apply an FX you use the FX Stack Manager Window**. To bring it up, click on **Menu>Dialogs>FX Stack Manager** or press **F2**. You can also click on the FX Button on the Main Window's Control Bar.

Remember that FXs are applied to each viewport independently, this way you can have one viewport playing back the unprocessed sequence, while on the viewport next to it, you can have the sequence playing back with a LUT or Primary Color Correction applied.

The FX Control window has its own menu bar from which you can access the FX Manager and apply FXs among other things (see FX Control Window Menu p40).

SECTION 2: PROCESS

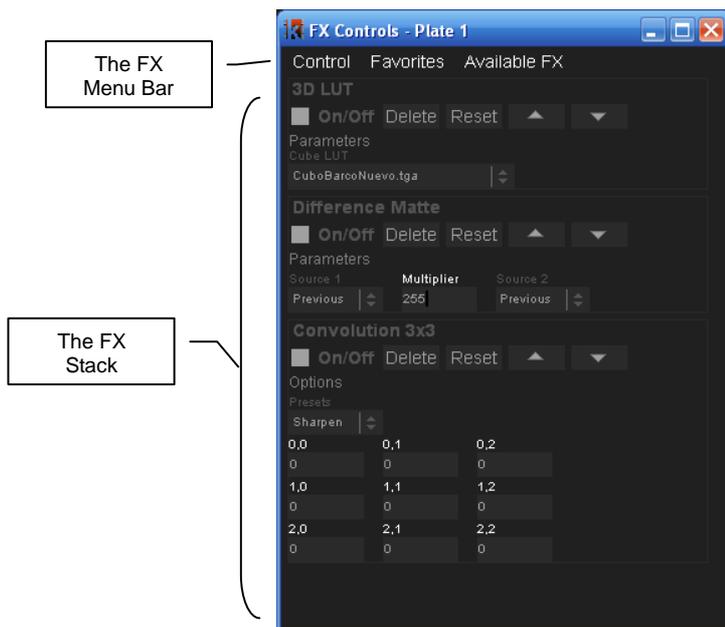


Figure 21 FX Stack Manager Window with the 3D LUT, Difference Matte and Convolution FXs applied

To apply an FX to a Viewport, open up the FX Stack Manager (F2) and click on the Viewport you wish to use, the FX Control Window title will change to show what viewport is selected (e.g. *FX Controls Plate 1* means you are working on Viewport 1), if not already there, an **Available FX** submenu will appear on the window's menu bar from which you can select what FX you wish to apply. To switch viewports in order to add, remove or modify FXs on them, simply click on the viewport which you wish to become active. The FX Control Window will change the title to show what Viewport you are working on.

Once you click on an FX from the Available FX menu, it will appear on the FX Stack as shown in Figure 21. The FXs are applied to the image in the order in which they were added to the stack, so the outcome of an FX will be passed to the one under it.

Common FX Controls

All FXs have a set of common controls:

- **On/Off:**  Turns the FX on or off. Once this checkbox is enabled, you will see the effect of the FX on the viewport. This way you can see the effect of an FX without having to delete it and re-apply it.
- **Delete:**  Removes the FX from the FX stack.
- **Reset:**  Returns all the FXs parameters to their default value.
- **Move Up/Down:**  Moves the FXs up the stack, meaning it will be applied before the one under it.



Most FXs will have parameters that you can set to customize their behavior. For example, the **Brightness, Contrast, Saturation** FX has controls for brightness, contrast, saturation and average luminance. Every FX's controls can be different, but they share some common characteristics. As with almost all other numeric input fields in JefeCheck, an FX's numeric parameters can be modified in two ways: by typing in a number, or by clicking and dragging with a mouse button on the field. The value will change according to the mouse movement at a rate dependant on what mouse button you click. If you click and drag with the left mouse button, the value will vary at the smallest decimal precision allowed by the field (if the field is something like Brightness then your mouse movements will increment or decrement the value by 0.001), if you

SECTION 2: PROCESS

middle click, the value will move at a 10x rate, and if you right click, it will move at a 100x rate.

Performance

A Viewport can have many FXs stacked one on top of the other, (although not all need to be on at the same time). You can apply an arbitrary number of FXs to a stack, but the performance and frame rate of the real time playback with many FXs applied will vary depending on your graphics hardware. When shopping for graphics cards, the main feature you should keep an eye for is the amount of Pixel Shaders or Pixel units (called Stream Processors in the newest graphics cards), the more of these that a graphics card has, the better the performance of an FX stack will be. At the time of writing, video cards that comply with the minimum hardware requirements have anywhere between 12 and 128 pixel shaders. The FX performance difference from these cards is not noticeable when a single simple FX is applied in a low resolution image, but as the number of FXs grows, or the image resolution increases, the performance at the lower end will suffer.



The amount of work that an FX can demand from a graphics card varies depending on the complexity of the FX, but tests have shown very good performance on mid range gaming level video cards with an excess of 5 FXs applied on a 2K resolution sequence.

FX Control Window Menu

The FX Control Window has its own menu bar which consists of three menus, the Available FX menu used to add FXs to a stack (when FXs are available), and a Control Menu with the following items:

- **FX Manager:** Opens up the FX Manager Window if you need to load an FX.
- **Clear All:** Removes all FXs from the stack (equivalent to pressing the delete button on each FX).
- **Save Stack:** You can save a stack with all the FXs and each one with their own particular settings into a file. You can later apply this same stack to another viewport or a different sequence. When you click on this, a browser window will appear, allowing you to choose a location and type in a name for the stack. JefeCheck FX stacks have a .fxs extension.
- **Load Stack:** If you have previously loaded an FX Stack, you can load it by selecting this item. Select the .fxs file from wherever it was stored and JefeCheck will place the appropriate FXs with the saved parameters on the stack. The FXs need to be loaded for the Load Stack function to work; otherwise you will get a message stating what FX is missing. Also note that when loading a Stack, the FXs contained there will be added to the current stack. If you want to replace the current stack, just clear all the applied FXs prior to loading.

SECTION 2: PROCESS

The Favorites menu allows you to save an FX stack and recall it later with a single keystroke. There are 5 favorite slots available, and each slot can contain a whole FX stack complete with parameters.

To save the current FX stack to a favorite, click on the Favorites menu, then Save Stack To... and finally to the stack number you wish to save it to. Or you can use the shortcuts **ctrl+shift+F1** through **ctrl+shift+F5** to save to favorites 1 through 5.

To recall an FX Favorite, you click on the Favorites menu and then Load on the stack number you want to load. You can also click on Append and it will add you favorite to the current stack. Or you can use the shortcuts **shift+F1** through **shift+F5** to load favorites 1 through 5, or **ctrl+ F1** through **ctrl+ F5** to append favorites 1 through 5.



SECTION 3: SHARE

In this section you will learn how to start and participate in a JefeCheck Remote Session.

In the previous sections you learned how to use JefeCheck to play sequences and process them using FXs, but the last big feature of JefeCheck is the ability to do all that in collaboration with people that might be miles away.

Collaborating with other people over a network connection or the Internet is called a Remote Session in JefeCheck.

Remote Sessions

When two or more people have a JefeCheck Remote Session, each participant has a copy of JefeCheck running on their computer, along with a local copy of the sequences that will be loaded for playback. JefeCheck does not send any image information over the network, this is for two reasons:

1. Sending image information requires great amounts of bandwidth that might not be available where a Remote Session participant might be.
2. Even if everybody had the bandwidth to send image data in real time, sensitive content such as frames from an unreleased movie are usually copied and sent

SECTION 3: SHARE

on a need to have basis through specialized, controlled and secured channels, managed by a data wrangler.

A JefeCheck Session requires at least two participants, one of who will need to act as a Session Server. If you have ever played a networked video game then you already know what this means, but if you haven't, the explanation is very simple and can be summed in three steps:

1. Start a Server



One participant, usually the one with the fastest connection and best computer starts a server from JefeCheck. **This participant will need to have an IP Address that other players can access from their network, normally called a Public Address if the session is being held over the Internet.** If you don't have a public IP address you will not be able to host a remote session over the Internet: this is simply how the Internet works, if other computers don't know where to find you (through a Public IP address) then you cannot exchange messages of any kind. This of course is not an issue when starting a server to hold a remote session over a local network or a VPN.

2. Other Participants Connect to the Server

Once the server is started, the server communicates to all other participants that the server is up and running, along with the IP Address that they must use to connect to it. This can be done over the phone, email, instant messaging, yelling across the hall or whatever method you prefer.

3. Play!

When everybody has connected to the server, they will all be in the same session, meaning that anything someone does on their copy JefeCheck will be mirrored on everybody else's copy, including the server.

Everything involved in setting up or joining a Remote Session is done from the Remote Session Manager Window.



The Remote Session Manager

To start or join a remote session, you use the Remote Session Manager Window. To bring it up you go to **Menu>Dialogs>Remote Session Manager** or press F5 as seen in Figure 22.

Connecting to a Server

The first section in the Session Manager Window is Connect to Server. In order to connect to a JefeCheck remote server that's already online you do the following:

1. **Type the IP address in the IP text box** (or select a recent one from the drop down menu next to it).

SECTION 3: SHARE

2. **Type in the port in the Port text box** for the server we are connecting to (ports 32000 and above recommended, but anything over 1025 should be fine).
3. **Type in the password in the Password box** (if the server requires one).
4. **Type in a Nickname in the Nickname box**, other participants in the session will identify your actions with this nickname.
5. **Select your pointer color** from 7 available colors.
6. **Click the Connect Button.**

The status box will indicate if the connection to the server was successful. If you are online with the server, the status box will turn green and say **Online**.

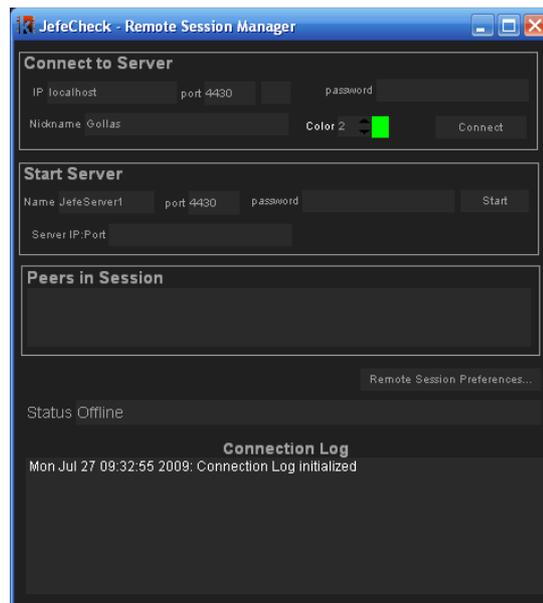


Figure 22 The Remote Session Manager Window

Starting a Server

The second Section in the Remote session manager is the **Start Server** section. You use this to start a server in order to host a remote session that other users can log into. When you are a server you are also a participant in the session as if you had connected to a server yourself. To start a server you do the following:

1. **Type the Name of the server in the Name box**, the server's name will be your nickname.
2. **Type in the port in the Port text box**, other users will use this port to connect to you (ports 32000 and above recommended).

SECTION 3: SHARE

3. **Type in the password in the Password box** (if you wish to require a password to connect to your server, recommended).
4. **Click the Start Button.**

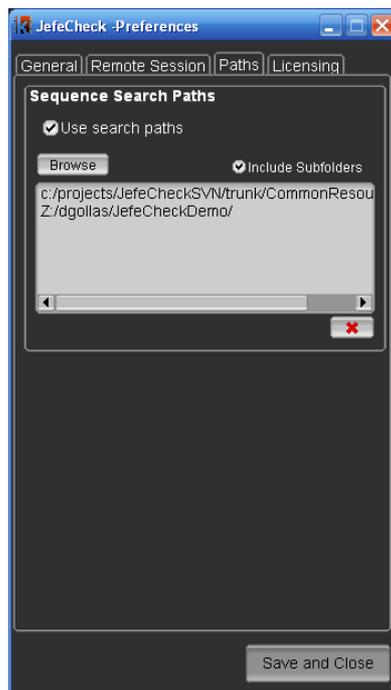
If starting the server is successful, the status bar will turn green and say **Online**. Also, the Server IP text box will show the IP address that other participants should use when connecting to you, along with the port that you are using (e.g. 172.16.28.105:32000, IP address 172.16.28.105 on port 32000), you should communicate this information to others attempting to connect to you.

Participating in a Remote Session

Once a server is set up and at least one more user is connected to the server, you can start using JefeCheck the way you normally would.

Loading Sequences in a Remote Session

Loading sequences in a remote session is done through the Playlist Window. **Only sequences loaded through the Playlist window will be automatically loaded in the other participant's computers.**



Remote session users may have different file system organizations; maybe one participant is loading the images from a folder on a CD, another one from a shared folder on a network, and a third one is loading from a local folder on a hard drive. This means that if user 1 loads a playlist item with a path like `c:/MyDocuments/images/` a remote user without that exact same path on their file system or a Linux or Mac computer will not be able to load the file.

To resolve this, you need to setup Search Paths in the JefeCheck preferences window. When a remote playlist item is loaded, JefeCheck will try to find it as it was received, if it can't be found, it will then try to find the sequence in the search paths and load the first match. Be careful with the search paths you pick and don't pick the root or `c:` folder, since that could slow down searching. Usually, you setup the search paths to point to include your project's render

folder or copy the files you will review remotely in a special folder on your desktop where you can later point the Search Paths to. Remember you can also setup more than one search paths.

To add a search path to your JefeCheck preferences, open the Preferences Window and go the Paths tab and click on the Browse button, find the folder where you want

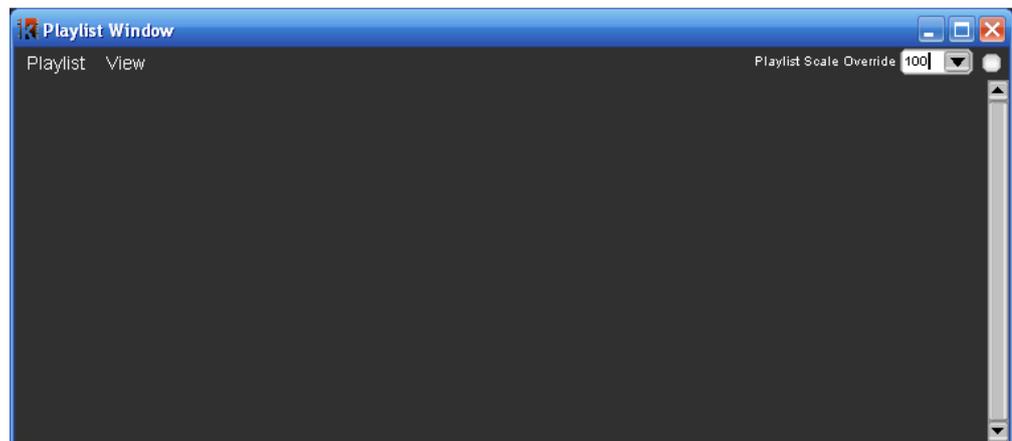
SECTION 3: SHARE

JefeCheck to search for sequences and click Ok. You can add as many search paths as you wish. Setting the Include Subfolders checkbox to true will also search in any subfolder within the paths.



One important thing to note is that although all users MUST have the same image sequences loaded, they don't have to load them at the same scale. That is, a user with a laptop can load the sequence at 50% or 25%, and a user on a bigger workstation can load the sequence at full resolution (see **Load Time Modification Controls:Scale** p9).

Even if a remote user loads a playlist item that contains sequences loaded at 100% size, you can override this from your own Playlist Window by specifying the Playlist Scale Override size in the top right corner of the window and checking the checkbox.



If you know how to use JefeCheck then you know how to participate in a Remote Session, you can do everything you do in JefeCheck normally and all other participants will see the changes reflected on their own JefeCheck. So you can pan, zoom, change layouts, play, scrub the timeline, advance or rewind frames, change In and Out points of the timeline, change the frame rate, change the playback mode, add and modify FXs (apply LUTs, Primary Color correction, mix two sequences etc) flip, flop... you get the idea.

Any playlist item you load during a remote session, will also be loaded in the remote participants sessions. Whenever a new user joins a session, it's playlist items will be merged with the ones already in the server.

There are only a few new commands involved in using JefeCheck in a Remote Session: Chat and Remote Pointers.

SECTION 3: SHARE

Chat

In JefeCheck you can text chat with all other participants, meaning you can send text messages that all participants will see on their screen while you interact with the rest of JefeCheck.

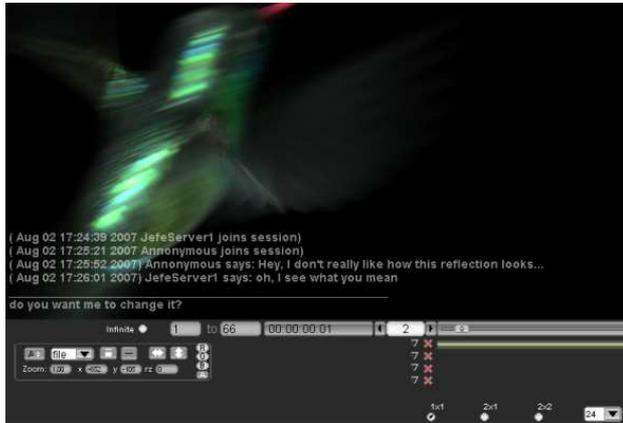


Figure 23 Chat Interface

When you are connected to a JefeCheck Remote Session press **y** to bring up the Chat Interface. Press **Escape** to exit the chat interface. The Chat interface is overlaid at the bottom of the Viewport Area of the Main Window as seen in Figure 23. The Chat interface is very simply and you will understand it instantly if you have ever used an instant messenger program. There are

two parts, divided by a horizontal line, when you are in chat mode, you type messages into the lower part simply by typing it on your keyboard and then press enter to send.

Whenever you receive a message, you will see it above the line that divides the received messages and the area where you type your messages.

You will only see the last 5 messages received, if you want to go back to see older messages, make sure you are in chat mode and press the **Up** or **Down** cursor keys on your keyboard to navigate through messages.

When you want to exit the Chat mode you press the **Escape key**. Since many keyboard shortcuts are bound to keys that you use to type messages, you will have to exit chat mode to use them.

When you are not in chat mode, you will still be able to see the messages sent to you, you can hide/show them press the **ctrl+y key** on your keyboard. Otherwise, unless you are in chat mode, sent messages will fade away automatically after a few seconds.

You can also save the whole Chat in a session by going to **Menu>File>Save Chat Log**, you will see a file chooser dialog, simply navigate to the path where you want to save and type in a name for the file. This is saved as a plain text file.

Remote Pointers

Asides from text, you can also make use of JefeCheck's remote pointers. This is a way to make everybody in the session see exactly where you are pointing at on the screen with your mouse.

To use a remote pointer, you simple right-click on the Viewport Area. Everybody else in the session will see a white point with your nickname next to it. As long as you keep

SECTION 3: SHARE

any button pressed on your mouse, the others will see your remote pointer. You can move your mouse while you keep the mouse button pressed and the others will see your remote cursor moving.



Note that if both users loaded the same sequence at different scales using the Scale parameter in the Load Window, JefeCheck's remote pointers will make up for the difference in scales. So you know that when you are pointing at a specific point in your viewport, the others are seeing your remote pointer at that same place in the image. This is also true for different monitor resolutions. Remote pointers work by showing where you are pointing at on the image on screen, not at where you are pointing at on your monitor.

Using both the chat and the Remote Pointers you can get some very useful interaction between all the participants in the session. Combining that with FXs to change colors or bring up difference mattes or anything else you can think of makes JefeCheck a very powerful remote review tool.

OTHER STUFF

4

SECTION 4: OTHER STUFF

This section shows some topics that don't quite fit anywhere else.

In the following subsections, you will learn about some of the other two dialog windows: the Preferences Window and the Rendering Window. You will also learn a little about the included FXs, and if you are a Technical Director or are so inclined you can learn how to create your own custom 1D/3D LUTs and FXs.

Preferences Window

The Preferences Window is used to change settings that affect the whole application. You access the Preferences Window with the shortcut **ctr+p** or from the **Menu>File>Preferences.**

You can close the preferences setting by clicking on Done, or you can click Save first to save the settings to your preferences file. Your preferences will be saved automatically when you close the application as well.

The Preferences Window is organized by tabs and sections, which will be described now:

- **General Tab**

- **Startup Section**

- **Open Load Window on Startup:** Determines if the load window will be automatically opened every time you open the program.
- **Start In fullscreen mode:** Determines if the program window maximizes to fullscreen automatically when opened.
- **Attempt To Recover from Crash:** When checked, JefeCheck will try to detect when the program crashed and will try to restore the last state of the application next time it is opened.
- **Default Browse Path:** If you are working on a specific project, you can set the Default Browse Path to point to the folder where you store your image files. This way when you open the browse dialog in the Load Window you will automatically be taken there.

- **BG Color Section**

- **Value:** Set the grayscale value of the background for the application. Can go from 0 (black) to a medium gray (127).

- **Action Feedback:**

- **Size:** The font size for the action feedback display (the display that is overlaid on the viewport when you adjust a setting or perform an action with a shortcut).
- **Fade:** Controls how long the Action Feedback display is show in seconds. Setting it to zero turns the display off.

- **Text Display Section**

- **Font Size:** Set the font size for the text display on the viewports.
- **Color:** Color for the text display on the viewports. Can go from black (0) to white (1)
- **Opacity:** The opacity of the text display on the viewports. Can go from almost invisible (0.1) to completely opaque (1)

- **Engine Section**

- **Percentage of RAM to use:** The higher you set this, the more frames you will be able to load into RAM, but using up all the RAM in your computer will slow down other processes dramatically. Recommended settings is 85%
- **Use Inactive Memory:** If you use this on Mac OSX, you might be able to fit a few more frames into RAM, at the expense of system performance.
- **Force GFL Loading Engine:** If you have problems loading certain types of DPX files on your system, try turning this on to load with a different loading engine. It will usually be slower but might be able to read some obscure DPX formats.
- **Continue Loading sequence after load error:** If a frame in the sequence is corrupt and can't be read, JefeCheck will stop loading the rest of the sequence unless this item is checked.
- **Enable Vertical Redraw Sync:** Syncs the viewport drawing to the monitor's refresh rate. Drastically reduces image tearing. On Xinerama displays in linux, this syncs to the main monitor, so displaying on a secondary monitor could increase tearing.
- **Try hard to maintain FPS:** JefeCheck will use more (a lot more) CPU time to maintain the desired FPS. If this is not on, the FPS counter might vary by a few hundredths every few seconds of playback. Recommended setting is to keep it off.
- **Balance Reads:** If more than one sequence is being loaded, balance reads will try to read both at them at the same rate. This helps if you are loading very small sized frames (jpegs for example) and also loading big heavy EXRs at the same time because the RAM will not fill up with the JPEGs before the EXRs have time to load.

Setting this to ON also greatly improves performance when reading more than one sequence from the same physical hard drive.

- **Force Single Buffered FX:** Enable this if you are using an old or lower end video card (GeForce 6 and older) or if you are having trouble viewing images with FXs correctly. Keep in mind that some of the more complex FXs require this to be on. Changing this parameter from on to off requires JefeCheck to restart to show the changes.

- **Formats Tab**

- **OpenEXR Section**

- **Ignore Display Window:** EXR images contain both a data window and a display window. JefeCheck will only load and show the Display Window unless this setting is checked. You usually want this off.
 - **Ignore Header's Aspect Ratio:** The EXR file header contains information about the Pixel Aspect Ratio that should be used to display the image. JefeCheck will automatically change the aspect ratio of the image unless this setting is checked. You usually want this off.
 - **Float > Integer transformation :** The EXR reference implementation recommends applying a color transformation to 16bit Floating Point images to be shown in 8 bit display. If you enable this transformation it will require you to reload whatever images where already cached . Exposure, Defog, Gamma, Knee Low and Knee High are all parameters that affect this transformation.

- **Remote Session Tab**

- **Chat Option Section**

- **Font Size:** The font size for the chat display
 - **Text Background:** Set wheather to show a background behind the text to improve readability.
 - **Opacity:** Specify the opacity of the chat display
 - **History Lines:** Specify how many lines of chat history should be visible.
 - **Auto Fade:** If you are not in chat mode, but there is chat visible, it will slowly fade away.
 - **Delay:** Specify the number of seconds before the chat fades away.

- **Remote Pointer Options**

- **Font Size:** Specify the font size to display the remote pointer name.

- **Pointer Size:** Specify the size of the remote pointer.
 - **Fade:** When on, the remote pointer will fade a few seconds after the last movement was registered on that participants computer.
 - **Trail:** Show a trail behind the remote pointer, helps overall visibility.
 - **Trail Length:** Specify the length of the remote pointer trail.
- **Update Frequency (per second) section (advanced)**
 - **Transformations:** How many times per second transformation messages will be sent over the network, reduce this number to ease network load or if you notice a lot of lag while transforming.
 - **FX Parameters:** How many times per second FX Parameter messages will be sent over the network, reduce this number to ease network load or if you notice a lot of lag while modifying FX paramaters.
 - **Other Messages:** How many times per second other messages (play/pause/scrub, flip, flop, etc) will be sent over the network, reduce this number to ease network load or if you notice a lot of lag while doing such actions.
- **Licensing Tab**
 - **License File Path:** Define where your JefeCheck license is stored.
 - **License Server Section (currently disabled)**

Rendering

Although JefeCheck is not a finishing tool, you can still output high quality images of you onscreen results in a variety of formats, including jpeg, bmp, tiff, tga, png and OpenEXR. On Linux, you can even create an avi movie file if you have mencoder installed.

The render process is as follows. You load a sequence and apply FXs to the viewport you where it is being displayed (if you want to). Then you open the Render Manager (**Menu>Dialogs>Render Manager** or **F6**), set a few parameters and click the Render Button. During render, the viewport may flicker, but rest assured the render is taking place. You can cancel the render at any time by clicking the Render button again (it the buttons label changes to Cancel during render).

The Render Window

In order to render out a viewport, you need to set a few parameters in the Render Manager Window. The parameters are organized in tabs:

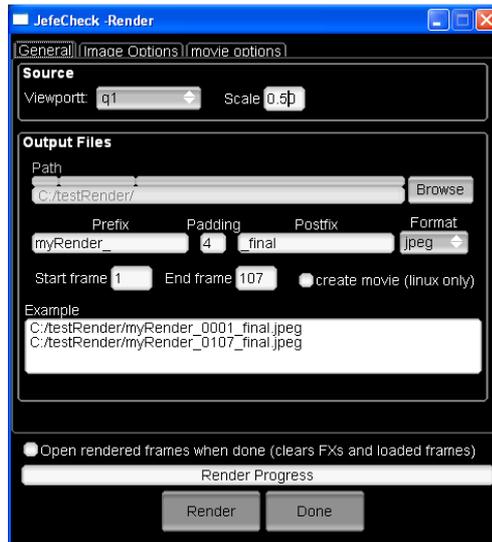


Figure 24 The Render Manager Window

- **General Tab**

- **Source Section**

- **Viewport:** Select the viewport you wish to render (q1, q2, q3, q4)
- **Scale:** The rendered image can be automatically scaled down by this factor: 1 is the original image, 0.5 is half etc.

- **Output Files Section**

- **Path:** The path where the rendered images will be written. Click the Browse to find a destination folder.
- **Prefix, padding, postfix, format:** The final image names will be composed of the prefix, a padded sequence number, a postfix, and a period followed by a format extension:

Prefix####PostFix.ext

- **Start Frame, End Frame:** Define the start and end frames to be rendered. These are in the same time domain as the timeline.

- **Create Movie:** This is a linux only feature at the moment. You need to have mencoder installed on your system. When checked, it will create a movie called PrefixPostfix.avi in the output path out of the rendered frames. The rendered frames will not be erased. Quicktime output will be implemented in future versions (Windows and OSX only).
- **Example:** The first and last output filenames will be shown here as an example of the filename generation.

- **Image Options Tab:**

- **Jpeg Section**

- **Quality:** Compression rate, higher Quality means better image fidelity but bigger files.
- **Progressive:** Stores the image as a progressive jpeg. Best for viewing on line or over a slow medium. As the image loads, the different scans stored in the jpeg complement each other to create the final image.
- **Optimized:** When not using the Progressive setting (baseline mode), optimized will yield slightly smaller file sizes.

- **PNG Section:**

- **Compression:** Set the compression rate for PNG image outputs. Larger compressions yield smaller file sizes but are more cpu intensive to decode.

- **TIFF Section:**

- **Compression:** Select the compression method for TIFF output files, currently only LZW is available.

- **OpenEXR Section:**

- **Depth:** Select the bitdepth at which the images will be written, currently only Half Floating Point is available.
- **Compression:** Select the compression method to use on OpenEXR files, you should look into the OpenEXR file format documentation to find out which one better suits your needs:

- **Lossless**

- **RLE**
- **ZIP** (per scanline or block)
- **PIZ**
- **Lossy**
 - **PXR24**
 - **B44** (vbr and fixed rate)
- **Movie options Tab**
 - **Mencoder Options (linux only) Section:**
 - **Kbits/second:** Set the desired bitrate of the output movie
 - **Codec:** Select the codec you wish to use to encode your movie. Currently only msmpeg4 is supported and recommended, since it can be watched on almost all Windows and Mac OSX machines.
 - **Open rendered frames when done:** When this option is checked, the rendered sequence will be loaded into the selected viewport's associated track. All FXs and the previous track will be unloaded to make way for the rendered sequence.

Included FXs

A list of the FXs currently installed with JefeCheck follows. More may become available online in the forums as they are developed/needed. The list is divided into categories:

- **Color**
 - **1D LUT:** Apply a 1D LUT color transformation. Select the 1D LUT you want to use from the list.
 - **3D LUT:** Apply a 3D LUT color transformation, select the 3D LUT you want to use from the list.
 - **Brightness, Contrast, Saturation:** Modify the brightness, contrast and saturation of the image, you can also change the average luminance for the contrast operation.
 - **Gamma:** Modify the gamma of the image.

- **Primary Color Correction:** Modify the brightness, contrast and saturation of the image on a per component basis.
- **Compositing**
 - **Chroma Key (Blue) and Chroma Key (Green):** Do a quick and dirty blue or green screen extraction and composite. Select the Bottom and top images, a bias parameter, show matte only, and show clean plate without the composite.
 - **Difference Matte:** Apply a difference matte operation on two images to expose the differences between the two. Select the two images to be compared and a multiplier. The operation can be viewed as $\text{absoluteValue}(A-B) * \text{multiplier}$.
 - **Inside:** Perform an inside composite.
 - **Mix:** Mix two images together according to an amount parameter.
 - **Ondita:** Proof of concept shader
 - **Over:** Perform an over composite operation. You can select the source and target image, the matte image, and the method used to extract the mask. You can also determine if the target image is premultiplied or not.
 - **Split:** Split the frame between two different images. The split is performed on a slope, and the axis of the slope can also be determined. The edge of the split can also be smoothed for a nicer looking split.
 - **Substract:** Perform substract operation on two images.
- **Special**
 - **Anaglyph Color Stereo:** Create an anaglyph color stereo image from a left/right pair. Each “eye” should be loaded in a different track. The left image is colorized to cyan, and the right one to red, you can adjust the amount of colorizing that takes place.
 - **OpenEXR:** Useful FX when loading HDR OpenEXR images in half-float format. You can adjust the gamma and exposure of the displayed image.
 - **Fields:** This is an animated effect. It alternates in blacking out the odd/even lines in the frame to achieve an interlaced look. This is not a de-interlacer, it is just an aesthetic simulation.

- **Transitions**

- **Fade:** This is an animated effect. It smoothly fades from one track to another. You can select the starting frame and the length of the fade.
- **Radial:** This is an animated effect. It transitions from one track to another using a growing radial matte. You can select the starting frame and the length of the fade. You can also set the smoothness of the matte's edge.

Creating custom 1D/3D LUTs

JefeCheck includes a few custom made 1D and 3D LUTs for you to use, but you will most probably come across the need to use a 3rd party LUT develop one yourself. This section will teach you how to create a JefeCheck compatible LUT, and how to mimic any color process through JefeCheck's very clever LUT reverse engineering.

1D LUTs

1D LUTs simply take single floating point input value, and output another floating point value. When you apply a 1D LUT to an image, each color component is passed through the LUT independently. A 1D LUT therefore can be described simply by a list input and output numbers. The JefeCheck 1D LUT format (.lut extension) is just a text file containing the output values, the input values are implicitly derived from the position in the text file. The file must contain the following information:

- Header: #JefeCheck LUT Header v1.0
- Number of entries in the LUT (usually 256 for 8 bit LUTs, 1024 for 10 bit etc)
- Input Bitdepth (deprecated but still necessary for compatibility): This complements the number of entries (8 for 8 bit, 10 for 10 bit etc)
- Output Bitdepth (deprecated but still necessary for compatibility): The range of the output values in bits (8, 10, 16)
- Entries: The actual values. These are the output values, **THEY MUST BE NORMALIZED**, meaning the output values must be in the range 0.0 to 1.0.

```
#JefeCheck LUT Header v1.0
256
8
8
1.0000
0.9961
0.9922
0.9882
...
0.0078
0.0039
0.0000
```

For an 8 bit LUT, there would be 256 entries: the first value would be the output value for 0, the second for 1, the third for 2 etc.... up to the last value, which would be the output value for 255.

A truncated LUT is shown here, for the whole version, see the included *invert.lut* 1D LUTs in your JefeCheck installation folder.

As you can see, this is an inversion 1D LUT with 256 entries: Low input values yield a high output and vice versa. This will effectively make a negative of the input image: white is black, black is white etc.

Other LUTs will obviously have more interesting entries.

3D LUTs

3D LUTs are similar to 1D LUTs in the sense that they take an input value and output another, the main difference is that the input is actually a 3 component vector, and the output is also a 3 component vector. In practical terms, this means that a 3D LUT converts one particular RGB color into another. This is more sophisticated than the single component transformation performed by 1D LUTs, because complex color relationships can be defined: e.g. we can define a 3D LUT that varies the amount of green depending on the amount of red, but that also takes into account how much blue there is.

A 3D LUT can be viewed as a cubical lattice, with each color sample placed on a vertex of the lattice. JefeCheck works with lattices of 16x16x16 samples. This effectively gives us 4096 different input and output values. When these samples are plotted on screen, the result is usually some kind of deformed “color cube”, this is why 3DLUTs are also called cubes.

JefeCheck can accept 3D LUTs in three different formats. The first one is a *Truelight* cube defined in a text file (.cub extension), the second one is a Nuke CMS color patch rendered to TGA image (up to size 16) and the third one is JefeChecks native LUT format, defined as a color patch saved in an image file.

Truelight cube support is still experimental, and may not yield exact results.

The native LUT format for JefeCheck offers a great deal of flexibility. It consists of an image 64x64 pixels in size. Each pixel represents an entry in the 3D LUT lattice. A

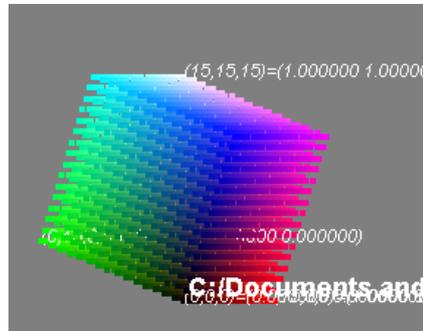


Figure 26 Perfect, Canonic or Unit Color Cube

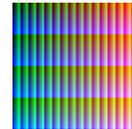
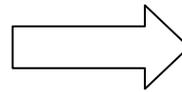


Figure 25 Unit Cube Color Patch

perfect color cube (i.e. one that does not change anything when applied) starts with perfect black (0,0,0) on one corner (coordinates 0,0,0), and progresses to perfect white (1,1,1) on the opposing corner of the cube (coordinates 15,15,15, remember this is a 16x16x16 cube). Along each edge of the cube, the primary components increase gradually from 0 to 1, so we also have a perfect red, perfect blue, and perfect green corners, and also have perfect secondary colors corners as seen in Figure 26 Perfect, Canonic or Unit Color Cube.

This perfect color cube can be encoded into an image by “unfolding” into 2D. The result of unfolding the image is the little color patch shown in Figure 25. Now for the interesting bit, if you process this little image through any color pipeline, you can “burn in” all the transformation into it. When you then load it back into JefeCheck as a 3D LUT, you effectively apply all your color processing to any image in real time.

Since the Canonic Color Patch is just an image, you can put through any image processing pipeline like Shake, Photoshop, etc. Simply open the canonicCubeImage.tga in your software, process it as you like (only color processing of course) and save back in TGA format. Then you can load it back into JefeCheck and use it as a 3D LUT!

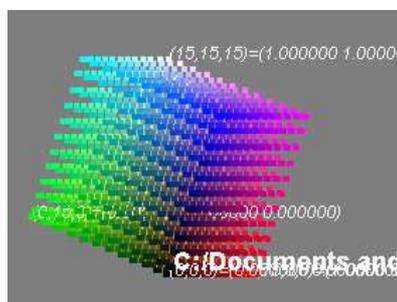


Figure 27 A modified Cube for the Viper Camera

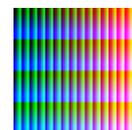
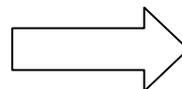


Figure 28 Viper Camera Color Patch

An example of a a processed color patch to correctly display data recorded with a Viper camera is shown in Figure 27 and Figure 28.

More extreme color processes will yield more deformed cubes. Remember that you can plot the LUT directly in JefeCheck by using the LUT Manager (see Visualizing LUTs p.36). For example, taking the patch and converting it to grayscale in an image

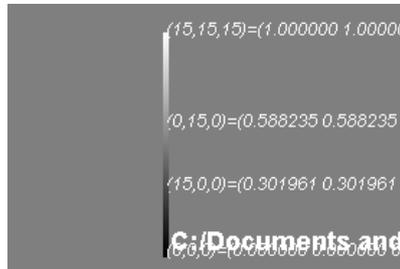


Figure 30 A grayscale 3D LUT

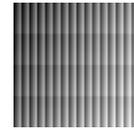
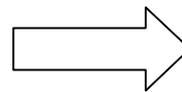


Figure 29 The grayscale color patch

processing program would yield a Cube where all the vertices are in a single line as

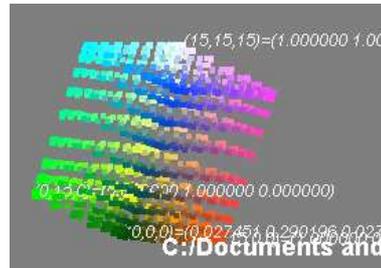


Figure 32 Heavily Processed LUT

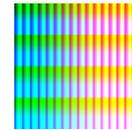
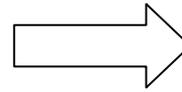


Figure 31 Heavily Processed color patch

shown in Figure 30 and Figure 29.

In the final example, we took the color patch and applied some heavy brightness and contrast operations on it, as well as a little shift towards green (Figure 32 and Figure 31).

Creating new FX plug-ins

Creating a new FX plug-in for JefeCheck is quite easy if you have some experience with a shading language, particularly the OpenGL Shading Language and have some technical knowledge on XML files.

JefeCheck's FXs consist of three files: a file.jfx, a .frag and a .vert.

The .vert and .frag files define OpenGL Shading Language vertex and fragment shaders, and the jfx file is a simple text file containing XML data. The XML document describes the FX, the controls and the variables that will affect the vertex and pixel shaders and the vertex and fragment shader names.

It is easier to show how it works by showing how the MIX FX is built, so you can jump to the example at the end of this section to see it and try to understand it and then use the following explanations as reference.

The jfx is an XML tree with a structure shown in Figure 33. Each node of the tree must contain certain XML attributes:

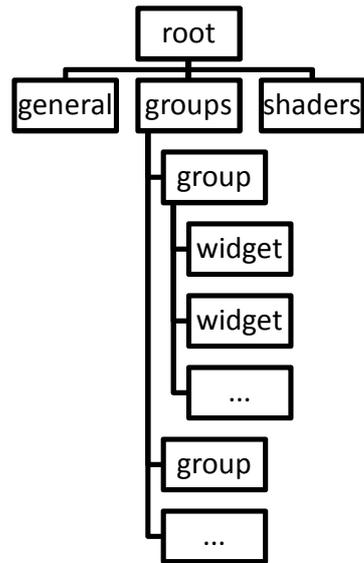


Figure 33 General XML Structure of the FX jfx file

- **root node:** The root node is the base of the XML file. Must contain a **comment** attribute with whatever text you wish.
- **general node:** The general node contains descriptive information on the FX in the following attributes
 - **Description:** A brief description of what this FX does. (e.g. "Blend two images together")
 - **menuName:** The menu structure where the FX will appear in the FX Stack Control Window (e.g. "Comp/Mix" will make the FX appear in a submenu called Comp, with the name Mix).
 - **version:** The version number of the FX (e.g. "2.0")
 - **author:** Who wrote the FX (e.g. "Daniel Gollas")
 - **name:** The name that will appear when the FX is applied in the FX Stack Manager Window and in the FX Manager Window. (e.g. "Mix")
- **groups node:** The groups node contains a series of **group** nodes, each containing a series of controls and variables for use within the shaders. Each

control group can have a name that will be displayed in the GUI. The information contained in these nodes will be used to create the GUI for the FX.

- **group node:** Each group node contains a series of widget nodes, and has a **name** attribute that names the group of widgets (e.g. "Parameters").
- **widget node:** These nodes are the real essence of the jfx file. Each one represents a GUI element for the FX. The GUI element is linked to a *uniform variable* in the shaders so you can use the widgets value inside them. There are several types of widgets available:
 1. **float** : Creates a numeric input in the FX's GUI and passes a floating point numeric value to the shaders.
 2. **bool** : Creates a checkbox widget in the FX's GUI and passes a floating point 1.0 or 0.0 to the shaders.
 3. **texture**: Creates a selection box in the FX's GUI and passes a rectangular 2D texture sampler to the shaders. The selection box is filled with 5 values, A,B,C,D and previous. A,B,C and D represent the image for that particular track, this is how you pass the image information to your shaders. Previous does not represent a particular track, but instead passes the the previous FX's result to the shader. This is what allows FXs to be stacked. If you don't specify any texture widgets, the previous one will be passed on by default with the variable name "image".
 4. **cube**: Creates a selection box in the FX's GUI and passes a 3D texture sampler to the shaders. The 3D texture contains a 3D LUT. This selection box is filled with all the loaded 3D LUTs.
 5. **lut**: Creates a selection box and passes a 1D texture sampler to the shaders. The 1D texture contains a 1D LUT. This selection box is filled with all the loaded 1D LUTs.
 6. **choice**: Creates a selection box and passes a floating point numeric number representing the index of the selected item to the shaders. A widget not of this type must have children nodes named *choice*, each of those child nodes must have an *label* attribute that will specify the text to appear in the selection box (see the OVER.jfx FX for an example).
 7. **newline**: Makes the following widgets appear in the next line.
 8. **spacer**: Creates a space between the previous widget and the next. Must have an attribute called width that specifies how much space will be added.

Each widget node must contain the following attributes to define it (*italic* means optional):

- *type*: One of the previously mentioned widget types (e.g. "texture" or "float")
 - *varName*: This is the link between the GUI and the shaders, the uniform variable with this name inside your shaders will contain this widgets value (e.g. "Amount" or "first").
 - *label*: This is the label that will appear in the GUI for this widget (e.g. "Blend Amount")
 - *minimum*: The minimum allowed value for float type widgets.
 - *maximum*: The maximum allowed value for float type widget.
 - *step*: Defines the decimal precision for float type widgets (0.1, 0.01, 1.0 etc).
 - *default*: The default value for the widget.
 - *labelColorR*, *labelColorG*, *labelColorB*: Together, they define the label color of the widget. Each one must specify a value between 0 and 255.
-
- **shaders node**: The shaders node specifies the name of the shaders that will be used in the FX. Many FXs can share the same shader files (most of the included FXs use the fixed.vert vertex shader, although each one has it's own fragment shader). The attributes that must be contained in the shaders node are the following:
 - *vertex*: specifies the filename of the vertex shader (e.g. "fixed.vert")
 - *fragment*: specifies the filename of the fragment or pixel shader (e.g. "MIX.frag").

All this sound very complicated, but in practice it is quite simple. Let's take a look at the MIX.jfx FX. It starts with a simple standard XML version 1 header and the root node with comment attribute.

Inside the root node we find the general node. Showing the descripio, manuName, version, author and name attributes.

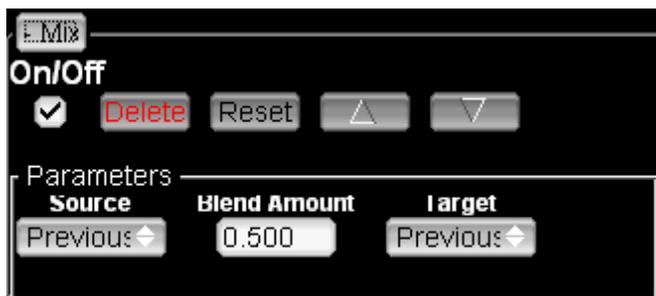
Following the general node is the groups node. This particular FX contains only one group of controls named “Parameters”.

Within that group node we have all the widgets. We start with a newline, then a texture

```
<?xml version = '1.0' encoding = 'UTF-8'?>
<root comment="this is the root node" >
  <general Description="Blend two images together" menuName="Comp/Mix" version="2.0"
  author="Daniel Gollas" name="Mix" />
  <groups>
    <group name="Parameters" >
      <widget type="newLine" varName="0" />
      <widget labelColorR="255" labelColorB="255" type="texture" default="0"
      label="Source" labelColorG="255" varName="first" />
      <widget labelColorR="255" labelColorB="255" type="float" step="0.001"
      default="0.5" label="Blend Amount" minimum="0" labelColorG="255" maximum="1"
      varName="Amount" />
      <widget labelColorR="255" labelColorB="255" type="texture" default="0"
      label="Target" labelColorG="255" varName="second" />
      <widget type="newLine" varName="0" />
    </group>
  </groups>
  <shaders vertex="fixed.vert" fragment="MIX.frag" />
</root>
```

with varName “first”, a float with varName “Amount” and a second texture with varName “second”. The MIX FX takes to textures and mixes them according to floating point amount, we want to be able to select which track will be mixed and how much, that is why we specify two textures and one float, easy! The other attributes simply specify some GUI aspects of the FX and are pretty self explanatory.

The shaders node also specifies to filenames, one for the vertex shader and one for the fragment shader.



The resulting GUI for this FX is shown here. The generic controls are added automatically, the name is at the top of the FX and then the Parameters group is shown with all the correct widgets inside.

The rest of the FX is the vertex and fragment shaders. The filenames in the .jfx file assume that the vert and frag files are stored in the same path. Most of the time you

```
void main()
{
    gl_TexCoord[0] = gl_MultiTexCoord0;
    gl_TexCoord[1] = gl_MultiTexCoord1;
    gl_TexCoord[2] = gl_MultiTexCoord2;
    gl_TexCoord[3] = gl_MultiTexCoord3;
    gl_Position = ftransform();
}
```

won't need to change the vertex shader, unless you really know what you are doing and are familiar with vertex shader (although you can do whatever you

want with it, just keep in mind that it must at least pass the texture coordinates for 4 texture units on to the fragment shader). The standard fixed.vert vertex shader is shown here.

```
uniform float Amount;
uniform sampler2DRect first;
uniform sampler2DRect second;

void main()
{
    gl_FragColor = mix(
        texture2DRect(first,gl_TexCoord[0].st),
        texture2DRect(second,gl_TexCoord[1].st),
        Amount);
}
```

The fragment shader is where most of the action happens since the FXs are essentially designed for 2D image processing at the pixel level. The shader gets its uniform variables from the application through the declared widgets and varNames in the jfx files, here is the example MIX FX with the passed parameters as uniform variables. You can see the float Amount uniform, the sampler2DRect first uniform, and the sampler2DRect second uniform. These would contain the values assigned through the GUI. The main function in the shader is the entry point to the program, and the resulting pixel goes in determine

A pixel shader as its name implies, is applied on a per pixel basis. The resulting pixel color must be assigned to the special variable gl_FragColor. In this example, we sample the color values of the two textures using the texture2DRect function, passing the *first* or *second* variables, and using the special variables gl_TexCoord[0] or gl_TexCoord[1] respectively. You should always use the 0 texture coordinates for the first declared sampler, the 1 for the second, etc (this is why it's important for the vertex shader to pass the appropriate texture coordinates). After sampling the two textures, we simply use the mix OpenGL Shading Language function to linearly interpolate the two samples by the amount indicated in the *Amount* parameter.

Aside from the uniform variables declared in your .jfx file, the application also sends a few additional variables that can help you create some interesting effects in your shader, you don't have to use them if you don't want to, but if you do you must declare them as uniform variables in your shaders. The variables are:

- texCoord0 to texCoord4: these are vec2 parameters, they hold the size of the four possible textures passed to the shader.
- currentFrame: this float value represents the current timeline value.
- timeStep: This float value represents the number of milliseconds since we last drew the frame.

- `targetFPS`: This float value represents the target FPS.
- `X_size`: The size for 3D LUT cube x. If your shader contains uses a 3D LUT variable named `myLut`, you will also receive the uniform `myLut_size` containing the size of the LUT.

You should take a look at all the other included FXs to get an idea of the type of things you can do.