Autodesk®
Creative Finishing
2012 Edition

Installation and Configuration Guide for Linux Workstations

Contents

Introduction

1

About this Guide

Overview

This guide provides information about installing and configuring the following Autodesk Creative Finishing applications:

- Autodesk® Inferno®2012

- Autodesk® Flame® Premium 2012

- Autodesk® Flame®2012

- Autodesk® Flint®2012

- Autodesk® Smoke®2012 for Linux® workstations

- Autodesk® Backdraft® Conform 2012

- Autodesk® Lustre®2012

NOTE In this document, we use the term Visual Effects and Finishing to refer to Inferno, Flame, Flame Premium, Flare, Flint, Smoke and Backdraft Conform software. This is how we identify instructions that do not apply to Lustre, which is grading software.

- For instructions on installing and configuring Autodesk® Smoke® for Mac OS® X, see *Autodesk Smoke 2012 for Mac OS X Installation and Licensing Guide*.

- For instructions on installint and configuring Autodesk® Flare™, see *Autodesk Flare 2012 Installation and Configuration Guide*.

NOTE In most cases, both hardware setup and application installation are done on delivery by an authorized technician, so you may not need to perform all of the procedures in these guides.

Intended Audience

This guide assumes you have knowledge of the Linux® operating system, preferably a distribution from Red Hat® Linux. It also assumes familiarity with Linux networking terms, tools, and procedures.

If you plan to reconfigure your workstation or your storage, knowledge of computer hardware, storage and networking in a professional video/film production environment is recommended.

Do not attempt to carry out the procedures in this guide if you are not familiar with the concepts they present. Contact Autodesk Media and Entertainment Customer Support if you require further assistance.

If you are upgrading an existing workstation without reinstalling the operating system, this document assumes you have *root* access to your system. If you do not have *root* access, contact your system administrator. The default root account password on an Autodesk workstation is *password*.

Installation Workflows

Workflow for Upgrading the Software

Follow this workflow to upgrade your software to a new version, service pack, or extension without reconfiguring your hardware or reinstalling your operating system.

To determine whether you need to install a new version of Linux® for a new software version or hardware platform, see Do I Need to Reinstall Linux? (page 33). If you do need to upgrade Linux, follow the steps in the Workflow for Installing from Scratch or Upgrading Linux (page 4) rather than this workflow.

To upgrade the software:

1   Gather the following materials and documentation for your installation:

- ■ The *Discreet Kernel Utilities* (DKU) installation package and the software installation package. For major releases, all product software is available on the product USB key. If you are installing a service pack or an extension, download the DKU *tar* file from the link provided in the Release Announcement you received from Autodesk.

- ■ The *Release Notes* for the software version or service pack you plan to install. The Release Notes contain important procedures that you might need to perform **before** upgrading your software, as well as any system requirements.

2   Make sure the hardware still meets all the system requirements for the new software version or service pack, as specified in the Release Notes for the version you plan to install.

3   Upgrade the Discreet® Kernel Utilities (DKU) to the required version.

4   Update the AJA card firmware if necessary. Firmware update instructions are available in Upgrading the AJA Firmware (page 45).

5   Install the software.

6   If you are upgrading from a previous version of a Visual Effects and Finishing application (not including Lustre), run the *copyProjects* utility to copy projects from previous versions of the application to the most recently installed version. See Upgrading Visual Effects and Finishing Projects to the Current Version (page 68).

    NOTE This step is not necessary if you are upgrading to a service pack of the same application version or to a service pack of the same extension.

7   License your software. If you are on subscription, you can choose between node-locked and network licenses. If you are not on subscription, you can only use node-locked licenses. See:

- ■ About Node-Locked Licenses (page 99)

- ■ About Network Licenses (page 105)

    NOTE Re-licensing is not necessary if you are upgrading to a service pack of the same software version or to a service pack of the same extension.

8   Read the Help to get familiar with your application. To open the Help, press **Ctrl+=** or click the Help button.

Workflow for Installing from Scratch or Upgrading Linux

Follow this workflow when you need to completely rebuild your system: from connecting the hardware peripherals and storage arrays, to installing the operating system, and installing, licensing, and configuring the software components.

Most of this workflow also applies when you need to reinstall or upgrade the operating system without changing your hardware or storage setup. A new distribution of Linux may sometimes be required for a new version or service pack of the software, or for newer hardware platforms. To determine whether you need to upgrade Linux for a new application version or hardware platform, see Do I Need to Reinstall Linux? (page 33).

To install your system from scratch or upgrade Linux:

1 Gather the following materials and documentation for your installation:
   ■ The installation disc for the Linux operating system. Your workstations ship with the custom Autodesk distribution of Red Hat Enterprise Linux on DVD. For information on the version of Linux required for your hardware platform, see Do I Need to Reinstall Linux? (page 33).

   ■ The *Discreet Kernel Utilities* (DKU) and software installation package. For major releases, the software installation packages are available on the USB key. For service packs and extensions, the software packages are distributed as *tar* files. Links to the *tar* are provided in the Release Announcement you received from Autodesk.

   ■ The *Release Notes* for the software version or service pack you plan to install. .

   ■ The Hardware Setup Guide for your workstation. This document contains detailed information on setting up your workstation.

2 Make sure the hardware meets all the system requirements. See your *Release Notes*.

3 If you are installing your hardware perform the following tasks:
   ■ Verify your hardware shipment, and make sure your facility meets the documented power and air conditioning requirements for the hardware components.

- Connect all peripherals (mouse, keyboard, Wacom® pen tablet, graphics monitor, house network) to the proper ports on the workstation.

- Connect your workstation to the Autodesk Wire® network, using either Gig-E or InfiniBand®.

- Connect a VTR and a broadcast monitor to your workstation.

- Connect and configure the audio hardware for your workstation.

- Connect your storage arrays to the workstation, but **DO NOT** power them on before having installed Linux, to prevent the Linux installer from attempting to format the arrays and use them as system drives.

4 If necessary, update and configure the BIOS of your workstation. See Configuring your System BIOS (page 28).

NOTE The BIOS settings must be properly configured **before** installing Linux.

5 Install the required distribution of Linux. See Linux Installation Workflow (page 34).

6 Perform the tasks in Linux Post-Installation Tasks (page 36) to configure your operating system.

7 Install the required version of the Discreet Kernel Utility (DKU). Refer to the latest *Release Notes* for the required version and Installing the DKU (page 44) for installation instructions.

8 Update the AJA card firmware on your workstation if necessary. See Upgrading the AJA Firmware (page 45).

9 If you are using a new Stone® Direct storage array, configure the RAID volumes, mount the storage, and format it. See Overview (page 46) for LUN creation instructions and XFS filesystem creation instructions.

10 Install the software.

11 For Visual Effects and Finishing software (not Lustre): Configure the software to use your media storage filesystem. See Configuring a Direct Attached Storage or SAN as Media Storage (page 60).

12 For Visual Effects and Finishing software (not Lustre): Configure the Wire network, background Wire and proxy generation. See Setting up the Wire Network (page 65).

13 License your software. If you are on subscription, you can choose between node-locked and network licensing. If you are not on subscription, you can only use node-locked licensing. See:
- About Node-Locked Licenses (page 99)

■ About Network Licenses (page 105)

> **NOTE** Re-licensing is not necessary if you are upgrading to a service pack of the same software version or to a service pack of the same extension.

14 Read the Help to get familiar with your application. To open the Help, press **Ctrl+=** or click the Help button.

Workflow for Connecting New Stone Direct Storage

Follow this workflow when you need to rebuild your Stone Direct storage array. For instance, this may be required when you purchase supplementary storage enclosures to expand available disk space.

> **NOTE** You do not need to reinstall or re-license the software after changing your storage configuration.

To connect new or expanded Stone Direct storage:

1 Connect your storage enclosures to the workstation. See Connecting Media Storage (page 19).

2 Configure the RAID volumes and mount the storage. See Configuring Storage.

3 For Visual Effects and Finishing software (not Lustre): Configure your application to use the new media storage filesystem. See Configuring a Direct Attached Storage or SAN as Media Storage (page 60).

4 Your new storage is now ready to use.

Preparing your Workstation

2

System Requirements

Supported Workstations

The following table lists the workstations and operating systems supported in this release, as well as the supported BIOS version for each workstation.

Workstation	Supported Software	Supported BIOS Versions	Operating System
HP® Z800	All Creative Finishing software	3.07	Red Hat® Enterprise Linux® Desktop 5.3 with Workstation Option (64-bit)
HP xw8600	All Creative Finishing software	1.35	Red Hat Enterprise Linux WS 4, Update 3 (64-bit)
HP xw9400	Not supported for Lustre	3.05	Red Hat Enterprise Linux WS 4, Update 3 (64-bit)
HP xw8400	All Creative Finishing software	2.26	Red Hat Enterprise Linux WS 4, Update 3 (64-bit)

See the latest *Hardware Setup Guide* for your workstation for BIOS update instructions.

Memory Requirements

Use the following table to determine the minimum and recommended memory requirements for your system. Use the recommended amount of memory to ensure optimal interactivity and stability.

Minimum RAM	Recommended RAM
12.0 GB for HP Z800 8.0 GB for other platforms	12.0 GB for HP Z800 16.0 GB for other plaforms (highly recom-mended)

For Visual Effects and Finishing products, you can adjust the MemoryApplication keyword in the software initialization file (*init.cfg*). For example, a system running at 8 GB with a 2K project could benefit from having the MemoryApplication keyword set to 5500. See MemoryApplication (page 127).

Setting Up Your Workstation Hardware

Overview

If you are only upgrading an existing application, you do not need to reconfigure your hardware.

This section contains wiring diagrams for HP Z800 workstation with the NVIDIA 6000 graphics card and the AJA KONA 3G. It does not contain information and diagrams for the older workstations supported in this version.

If you have an HP Z800 with an NVIDIA 5800 graphics card, consult the HP Z800 Workstation Hardware Setup Guide

Typical Configuration Overview for Visual Effects and Finishing Applications

The following illustration shows a typical configuration, including the Wacom tablet, and other optional components.
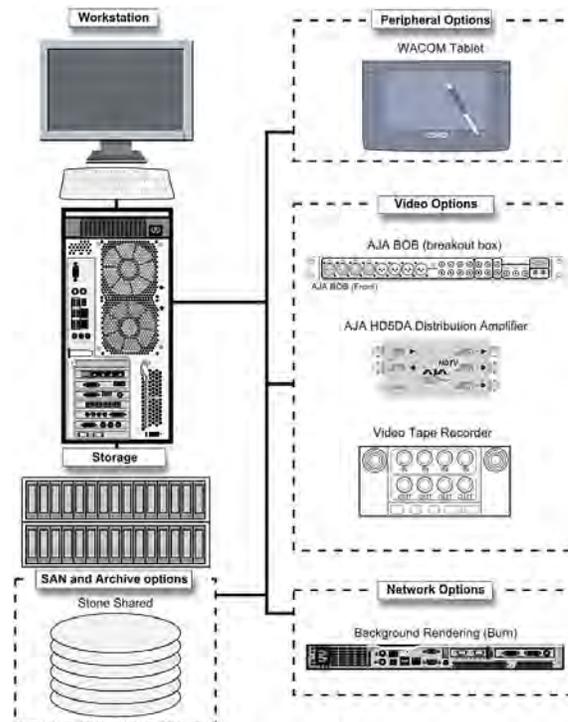
Typical Configuration Overview for Autodesk Lustre

A Lustre system consists of a number of different hardware and software components. This section describes each of the hardware components in a Lustre installation, and the software associated with each component. You may or may not have all of the components listed here in your Lustre setup. The type of workgroup, as well as the feature set you purchase for each of the workstations in that workgroup, determine the components in your installation.

Lustre Workstations

There are two kinds of Lustre workstations: the Master Station and the Lustre Station. The optional Lustre Station offers a way to improve the efficiency and cost-effectiveness of a pipeline by offloading tasks that do not require the full feature set of the Master Station.

Master Station

The Master Station is designed for GPU-accelerated sessions where the colorist works together with the cinematographer. It includes an extensive creative toolset for elaborate visual design and grading, using up to 4K resolution and 16-bit files and for completing tasks like dust busting, conforming, rotoscoping, and capture/playout. It also includes SD and HD I/O, dual link and HSDL video formats, and the film workflow features, which consist of infrared channel dust removal and support for all standard input and output resolutions and bit-depths.

The Master Station can support up to three panels of the Autodesk Control Surface and, with an additional license, a Slave Renderer.

Lustre Station

Tasks that do not require the direct intervention or supervision of the colorist can be efficiently handled by a Lustre Station. Multiple Lustre Stations can work in parallel to increase throughput and can be used for tasks such as dust-busting, preparatory work, fine-tuning creative sessions, conforming data from EDLs, updating editorial changes, and mastering to different formats

using the real-time deliverables function. The Lustre Station includes the film workflow features, and the ability to create geometries and masks.

Primary colour grading on the Lustre Station requires a separate license. The following features also require an additional license: SD and HD I/O, dual link and HSDL video formats.

The Lustre Station does not support the Slave Renderer.

Other Components

You can expand the features of your Lustre system and improve the efficiency of your workflow by adding any of the following components.

Control Surface	The Autodesk Control Surface provides improved interactivity when colour grading film and video footage. You can perform many of the same tasks you do in the Lustre user interface using the Control Surface.

You connect the Autodesk Control Surface to a Master Station and configure the Control Surface on the workstation to which it is connected.

Video I/O Card and Breakout Box	On the HP Z800 workstation, video I/O is provided by the AJA card. The video I/O card consists of an HD/SD card and a breakout box. This configuration provides real-time SDI input and output of uncompressed 8- or 10-bit HD or SD video in both YUV (4:2:2) and RGB formats (4:4:4 or 4:2:2). For a list of supported video formats, see the *Autodesk Lustre User Guide*.

The drivers for the video card and breakout box are installed by the Discreet® Kernel Utility (DKU).

Slave Renderer	The Slave Renderer is a rack-mounted server that frees system resources by automatically off-loading render tasks, thus ensuring real-time interaction on the Lustre system.

Connect the Slave Renderer to a switch on your house network. Note that only one Lustre Master station can use the Slave Renderer.

You install the Slave Renderer software package on the Slave Renderer.

Background Renderer	Background rendering frees up Lustre workstations for colour grading. You can use up to eight background rendering machines to process your final frames.

You install background rendering software on each background render node.

The following illustration shows a typical Lustre workgroup configuration built around a Master Station. It includes a Lustre Station, and other optional components.
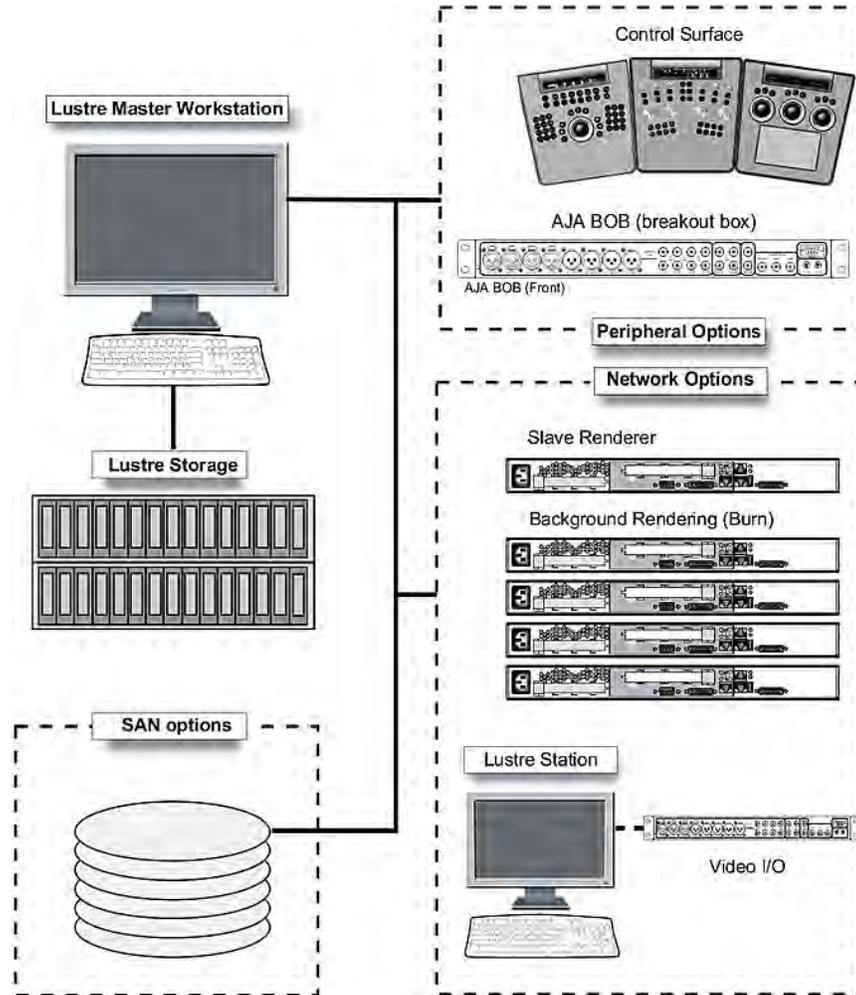
Overview of Peripheral Connections

The following diagram provides an overview of the adapter card positions, and illustrates how to connect peripherals, such as the graphics monitor, keyboard, mouse, Wacom® Intuos® pen tablet, and house network, to the HP Z800 workstation.

It is recommended you connect all hardware peripherals, as instructed in this chapter, before booting the workstation for the first time.

HP Z800 Workstation

To Lucid Converter
(for remote control)

To keyboard,
mouse, tablet

To Lustre control surface

To facility network

Power

To SAN, private network,
or Incinerator node

To Direct-Attached Storage

To display port
of the Monitor

To InfiniBand or
10-GigE Switch
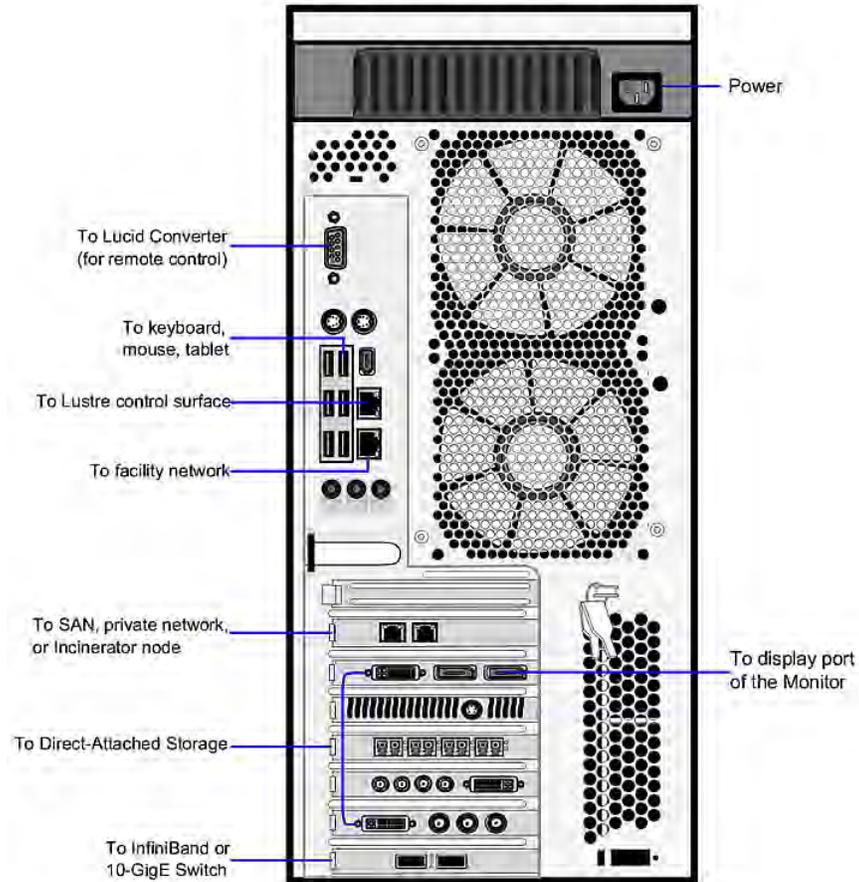
NOTE This diagram depicts the HP Z800 workstation with a 2-port GigE adapter in slot 1 (top to bottom), and a Mellanox® QDR InfiniBand / 10-GigE adapter in slot 7. Optionally, your workstation can be set up with a second ATTO Fibre Channel adapter in slot 1.

Connecting Video Components

The following diagram illustrates how to connect the video hardware components included in your shipment. The only video hardware you must

provide are a sync generator, VTR, HD/SDI-ready broadcast monitor and patch panel (if desired).
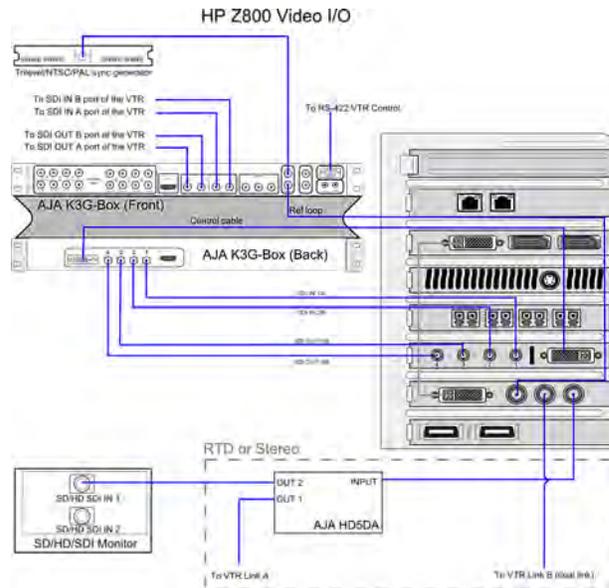
1. Connect the output of the sync generator to the top **Ref Loop** port of the AJA K3G-Box.

2. Connect the **Input** port of the NVIDIA SDI card (the one next to the DVI port) to the bottom **Ref Loop** port of the AJA K3G-Box.

3. Connect the **Fill** (outer) port of the NVIDIA SDI card to the **Input** port of the AJA HD5DA distribution amplifier.

4. Connect one of the **OUT** ports of the distribution amplifier to an **SDI IN** port of the broadcast monitor.

5. For RTD or Stereo, connect another **OUT** port of the amplifier to the **Link A** input port of the VTR.

6. For Stereo or RTD, connect the **Key** (middle) SDI port of the NVIDIA SDI card to the **Link B** input port of the RTD VTR.

7. Connect the **RS-422** port of the AJA K3G-Box to the VTR control port.

8. If not using RTD, connect **SDI OUT A**, **SDI OUT B**, **SDI IN A** and **SDI IN B** ports of the VTR to the **SDI1**, **SDI2**, **SDI3** and **SDI4** ports on the front of the AJA K3G-Box.

9   Connect the **SDI IN 1/A**, **SDI IN 2/B**, **SDI OUT 1/A** and **SDI OUT 2/B** plugs of the AJA cable to the **SDI1**, **SDI2**, **SDI3** and **SDI4** ports of the AJA Kona 3G card.

10   Connect the **SDI IN 1/A**, **SDI IN 2/B**, **SDI OUT 1/A** and **SDI OUT 2/B** plugs on the other end of the AJA cable to the **SDI1**, **SDI2**, **SDI3** and **SDI4** ports on the back of the AJA K3G-Box, as illustrated.

11   Connect the AJA control cable to the AJA Kona card and to the corresponding port on the back of the AJA K3G-Box.

Connecting Audio Components

Overview of Audio Components

Your application uses the Discreet Native Audio subsystem. Discreet Native Audio uses the following hardware components, shipped with your system.

Lucid ADA 88192 Audio Converter	Converts signals between the workstation and all digital or analog audio I/O devices.
AJA K3G-Box Balanced Audio breakout box and AJA Kona 3G adapter card	The Balanced Audio breakout box is the audio component of the AJA breakout box. It provides connections for audio I/O. This breakout box connects to the AJA Kona 3G adapter on your workstation. The Kona 3G adapter provides real-time input and output of uncompressed SD and HD video signals as well as audio data at 24-bit resolution. The Kona 3G adapter handles balanced AES/EBU audio signals from the Balanced Audio breakout box.

Wiring Audio Components

The following diagram illustrates how to connect the Discreet Native Audio hardware components to the AJA breakout box.

AJA K3G-Box (front)

Digital
outputs

Male 75 Ohm BNC
to male 110 Ohm XLR
Adapters

To workstation

serial port

Lucid ADA 88192
Converter

Digital
outputs

Output to
analog destination

VTR

---- Required for remote
control of converter
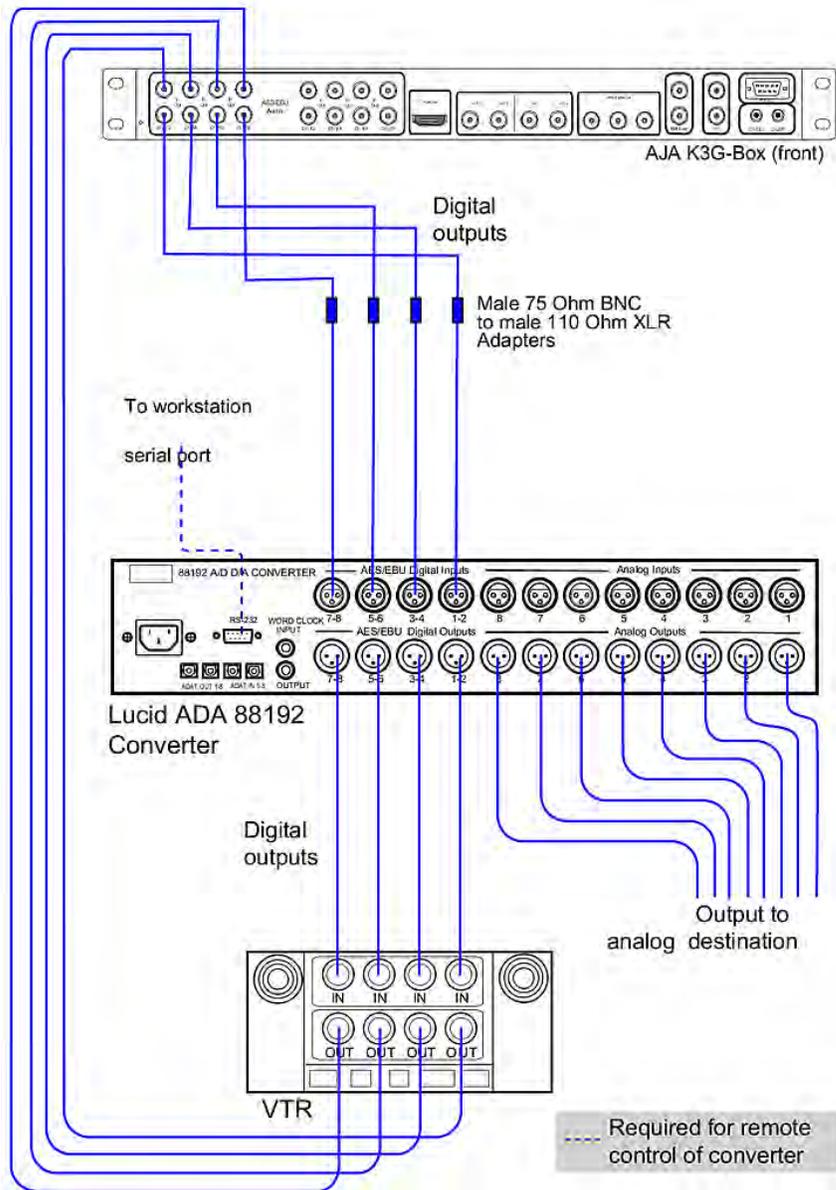
Connecting Media Storage

The workstation can be connected to two types of media storage.

■ **Stone Direct XR-series disk arrays** Autodesk's high-performance direct-attached storage (DAS) solution designed to address the different real-time playback requirements of various workflows. Direct attached storage devices provide storage to individual workstations.

■ **A SAN** A storage infrastructure that allows multiple workstations to share simultaneous access to a central storage enclosure.

Connecting the Fibre Channel Adapter

On HP Z800 workstations, connections to media storage are made using the ATTO Celerity FC84-EN fibre channel adapter, located in slot five (top to bottom).

The fibre channel adapter is equipped with four ports, called *loops*. You can connect your storage enclosure to the fibre channel cards using either 2 loops or 4 loops. A 2-loop device can be connected to either the inner pair of ports or the outer pair of ports. A 4-loop device requires both the inner pair and the outer pair.
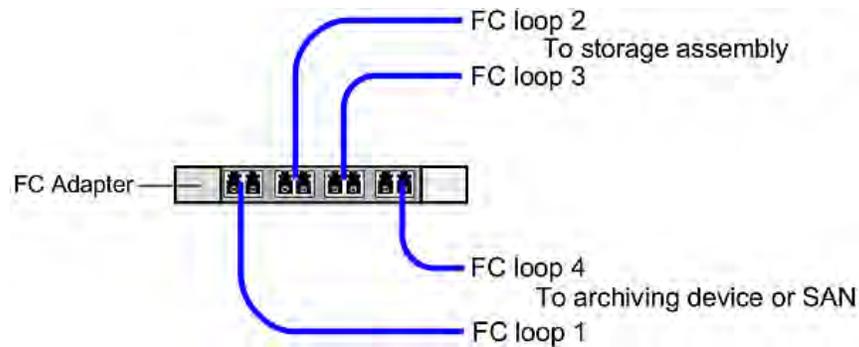
NOTE If you are not using all the ports of a fibre channel adapter, terminate the two unused ports with the FC loopback couplers that shipped with your system. Terminating these ports significantly decreases the time required to boot the system.

Connecting Stone Direct Storage

The diagrams in this section illustrate how to connect the fibre channel loops to the Stone Direct storage enclosures supported with your workstation. For technical details on each storage model, see the Storage chapter in the hardware setup guide.

Connecting XR 6500 Storage

The following diagrams illustrate 2-loop and 4-loops connections for XR 6500 series storage assemblies. Cable your storage exactly as illustrated to ensure proper functionality.

A XR 6500 RAID enclosure supports a maximum of seven XE expansion enclosures. Configurations with two XR RAID enclosures are not supported.

WARNING Do not power on your storage before installing Linux®, otherwise the Red Hat® installer might attempt to format the storage array and use it as the system disk.

Single XR 6500 RAID enclosure, 2 loops

To FC adapter on workstation / LMS (port 1)

To FC adapter (port 2)
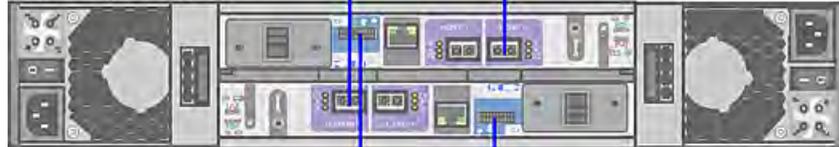
XR RAID Enclosure

XE (EBOD) Expansion Enclosure

XE (EBOD) Expansion Enclosure

XE (EBOD) Expansion Enclosure

XE (EBOD) Expansion Enclosure

– – – – – Cabling for additional enclosures

Single XR 6500 RAID enclosure, 4 loops

To FC adapter on workstation / LMS (port 0)

To FC adapter (port 1)

To FC adapter (port 2)

To FC adapter (port 3)

XR RAID Enclosure

XE (EBOD) Expansion Enclosure

XE (EBOD) Expansion Enclosure

XE (EBOD) Expansion Enclosure

– – – – – Cabling for additional enclosures
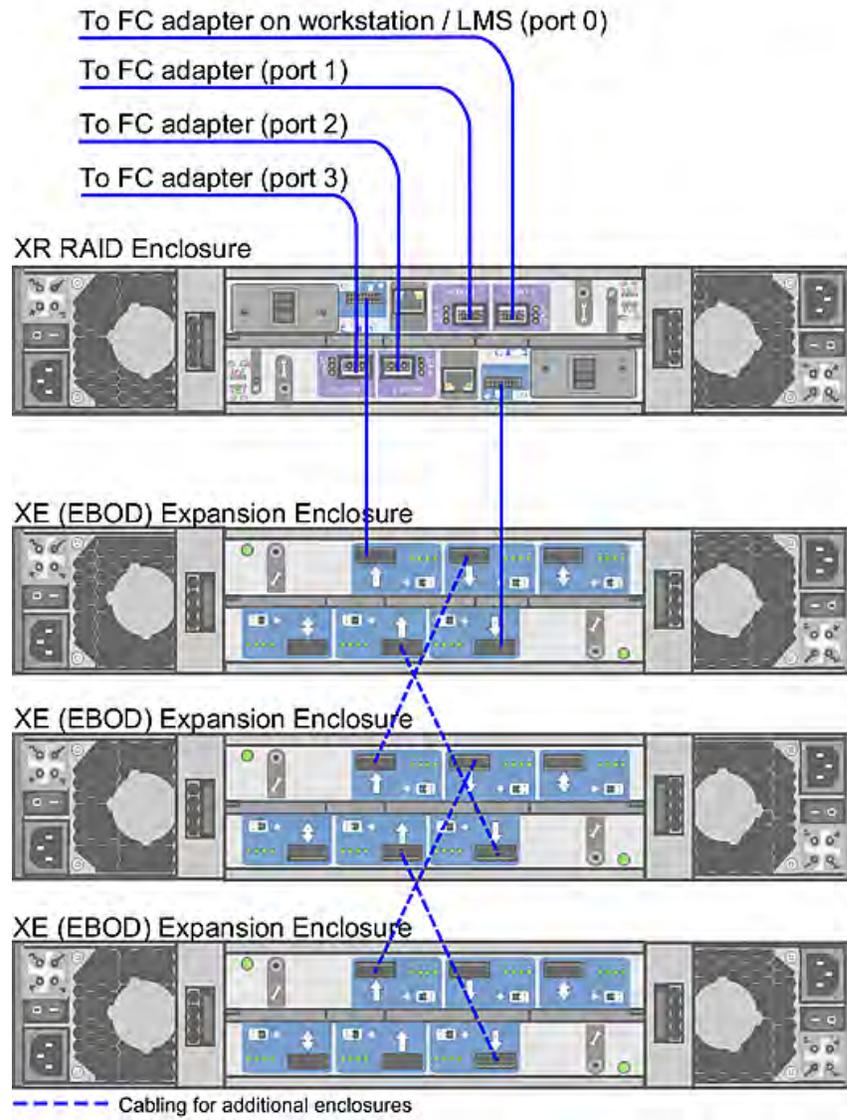
NOTES:

■ In a 4-loop configuration, you need a minimum of one XE expansion enclosure attached to the XR 6500 RAID enclosure.

■ The total number of enlcosures must be an even number.

Connecting XR 6412 Storage

The following diagrams illustrate 2-loop and 4-loop connections for XR 6412 series storage assemblies.

A XR 6412 RAID enclosure supports a maximum of seven XE expansion enclosures. Configurations with two XR RAID enclosures are not supported.

WARNING Do not power on your storage before installing Linux, otherwise the Red Hat installer might attempt to format the storage array and use it as the system disk.

Single XR 6412 RAID enclosure, 2 loops

To FC adapter on workstation / LMS (port 1)
To FC adapter (port 2)

XR RAID Enclosure

XE (EBOD) Expansion Enclosure

XE (EBOD) Expansion Enclosure

XE (EBOD) Expansion Enclosure

XE (EBOD) Expansion Enclosure

- - - Cabling for additional enclosures

Single XR 6412 RAID enclosure, 4 loops

To FC adapter on workstation / LMS (port 0)
To FC adapter (port 3)
To FC adapter (port 1)
To FC adapter (port 2)

XR RAID Enclosure

XE (EBOD) Expansion Enclosure

XE (EBOD) Expansion Enclosure

XE (EBOD) Expansion Enclosure

XE (EBOD) Expansion Enclosure

--- Cabling for additional enclosures

NOTE In a 4-loop configuration with a XR 6412 RAID enclosure, you need a minimum of one XE expansion enclosure attached to the XR RAID enclosure.

Connecting XR 5402 and XR 5412 Storage

The following digrams illustrate 2-loop and 4-loop connections for XR 5402 and XR 5412 series storage assemblies.

XR 5402 and XR 5412 series storage assemblies support 2-loop configurations with one XR RAID enclosure, and 4-loop configurations with two XR RAID enclosures.

WARNING Do not power on your storage before installing Linux, otherwise the Red Hat installer might attempt to format the storage array and use it as the system disk.

Single XR 5402 / 5412 RAID enclosure, 2 loops

To FC adapter on workstation / LMS (port 1)
To FC adapter (port 2)

XR RAID Enclosure

XE (EBOD) Expansion Enclosure

XE (EBOD) Expansion Enclosure

XE (EBOD) Expansion Enclosure

XE (EBOD) Expansion Enclosure

- - - Cabling for additional enclosures
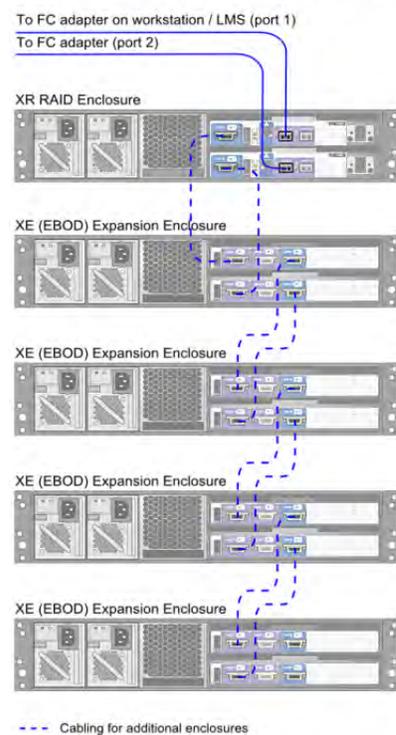
NOTE The XR 5402 and XR 5412 RAID enclosures support a maximum of four XE expansion enclosures.

Two XR 5402 / 5412 RAID enclosures, 4 loops
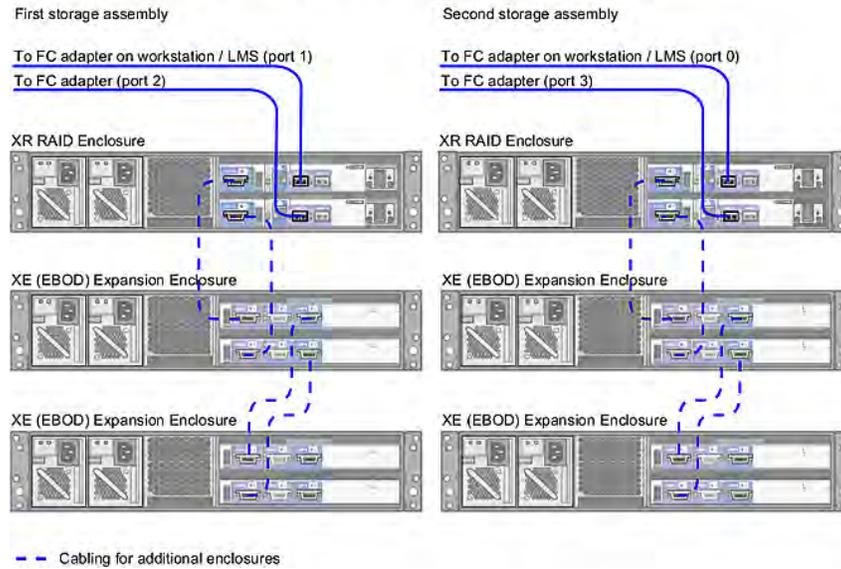
NOTE In a configuration with two XR RAID enclosures, the number of XE extension enclosures per XR RAID enclosure must be the same. The total number of enclosures in the storage assembly must be an even number.

Connecting Archiving Storage to Visual Effects and Finishing Workstations

NOTE This section does not apply to Autodesk Lustre.

Your Visual Effects and Finishing software supports archiving to USB 2.0, FireWire® (IEEE 1394) and fibre channel devices. This includes filesystems, VTRs and tape drives. This section discusses the requirements for filesystem and tape drive devices. For information on connecting a VTR, see Connecting Video Components (page 15).

Filesystem Devices

USB 2.0 and FireWire (IEEE 1394) attached devices must be formatted with one of the following filesystems: XFS, ext2, ext3, or HFS (Mac®). NTFS is not supported.

NOTE Care must be taken to ensure that the filesystems used to store archives are reliable, stable and properly backed up so as to protect the archive integrity. This functionality is not provided by the application.

SCSI Tape Drive Devices

SCSI tape drives can be connected to a fibre channel port of the workstation by way of a fibre-channel-to-SCSI bridge, such as the Atto FibreBridge 2390D. The tape drive device must meet two criteria. First, the device's driver must accept standard UNIX tape device calls. Second, the device must be certified for use with the currently supported version of the workstation operating system and kernel.

While Autodesk Media and Entertainment makes no certification statement about any tape drive device, you can use the "Tape Drive Test" plug-in in Autodesk SystemCentral to test SCSI tape drives for compatibility. For more information, see the *Autodesk SystemCentral User Guide*.

Configuring your System BIOS

This section contains information on the certified BIOS settings for the HP Z800 workstation, as well as instructions on updating your BIOS and restoring it to the default factory settings. For information on updating your BIOS version, or reverting the BIOS to factory defaults, consult the Hardware Setup Guide.

NOTE The BIOS of your workstation is correctly configured prior to shipping, so you normally do not need to perform the procedures in this section.

Checking the BIOS Settings

After wiring the workstation and **before** installing Linux, turn the workstation on and check the BIOS version and BIOS settings.

WARNING If BIOS settings are not set up correctly before installing Linux, you will need to reinstall the OS.

The certified BIOS version for your workstation is **3.07**.

Make sure the BIOS version number that appears on screen while booting the workstation matches the certified version. If the versions do not match, update the BIOS to the certified version, after configuring BIOS settings. BIOS update instructions are at the end of this chapter.

To adjust system BIOS settings, press F10 while booting the workstation.

The following table lists the Autodesk certified BIOS settings. Items not listed are set to their default factory settings.

BIOS Menu	Submenu	Item	Value
Storage	Storage Options	SATA Emulation	RAID+AHCI
	Boot Order	Optical Drive	
		USB Device	
		Hard Drive	<hard drive model>
			Integrated SATA
Power	OS Power Management	Runtime Power Management	Disable
		Idle Power Saving	Normal
		MWAIT Aware OS	Disable
		ACPI S3 Hard Disk Reset	Disable

BIOS Menu	Submenu	Item	Value
	Hardware Power Management	SATA Power Management	Disable
		Intel Turbo Boost Technology	Disable
Advanced	Processors	Hyper-Threading	Enable
	Chipset/Memory	Memory Node Interleave	Enable
		NUMA Split Mode	Disable
	Device Options	S5 Wake on LAN	Disable
		Internal Speaker	Disable
		NIC PXE Option ROM Download	Disable
	Slot 1 PCIe2 x8(4)	Option ROM Download	Disable
	Slot 3 PCIe2 x8(4)	Option ROM Download	Disable
	Slot 4 PCIe2 x16(8)	Option ROM Download	Disable
	Slot 6 PCI32	Option ROM Download	Disable

Connecting the Autodesk Control Surface to a Lustre Workstation

The Autodesk Control Surface consists of three panels connected to the workstation's on-board GigE port 1 by way of the supplied network switch. The central panel requires an additional connection to the remote unit of the USB extender kit.

This section provides information on making the physical connections needed to operate the Autodesk Control Surface. For information on using the Control Surface, see the *Autodesk Control Surface User Guide*.
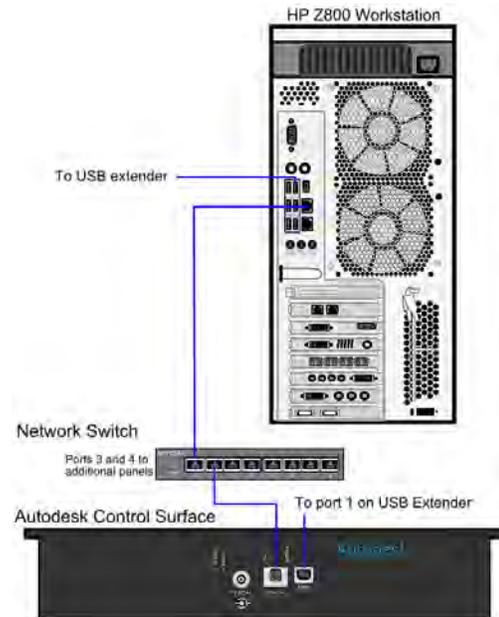
NOTE

The illustration shows the central panel only (the Colour Grading panel). This panel is easily identified, since it has both an Ethernet network port and a USB connector. It is also the panel with the built-in tablet.

To connect the Autodesk Control Surface:

1 Connect each panel to a power supply using the AC power adapter cables.

2 Use network cables to connect each panel to the available ports in the network switch.

3 Connect the Colour Grading panel to the remote unit of the USB extender kit.

4 Use a network cable to connect an available port on the Netgear network switch to the workstation, as illustrated.

Assigning an IP Address to the Autodesk Control Surface

You must assign an IP address and subnet mask to the Autodesk Control Surface.

Assign the network port connected to the control surface with a static IP address that does not interfere with any of the IP addresses on your network.

To configure the Autodesk Control Surface:

1 Type:

   nedit /etc/sysconfig/network-scripts/ifcfg-eth<port#>

2 Modify the IPADDR and NETMASK values. For example:

```
IPADDR=192.168.125.10
NETMASK=255.255.255.0
```

Configuring Lustre to Connect to the Autodesk Control Surface

Before you can configure Lustre to connect to the Autodesk Control Surface, Lustre must be installed, either as a standalone install (page 56) or as part of a Flame Premium install (page 53).

To configure Lustre to use the Autodesk control surface:

1 Turn the power on to each of the modules and look at the top display panel on the module. It should display the panel name and ID.

2 After you install Lustre, open the init.config file located in /usr/autodesk/<lustre_version> with a text editor.

3 In the ControlSurface section of the init.config file of your project, enter the panel IDs for each panel.

```
ControlSurface
AutodeskPanels state="On"
PanelIDs function="<panel_ID>" grading="<panel_ID>"
navigation="<panel_ID>"
```

4 Start Lustre. The following message should appear in each Console:

```
Panel #<panel_ID> is detected
```

This confirms that the Autodesk Control Surface is enabled.

Installing Red Hat Enterprise Linux or CentOS and the DKU

Do I Need to Reinstall Linux?

Your workstations ship with the correct Autodesk distribution of Red Hat Enterprise Linux already installed and configured. Thus, you should only need to install Linux in the following situations:

■ You plan to upgrade your software on an older supported hardware platform running a version of Red Hat Enterprise Linux no longer supported by the current version of the software. In this case, you must perform a fresh install of the required Red Hat Enterprise Linux version.

■ Your system experienced an unrecoverable hard disk failure and you had to replace the system disk.

NOTE If you must replace the hard disk of your workstation, you need to obtain a new license. Contact Customer Support for assistance.

Use the following table to determine the correct version of Red Hat Enterprise Linux required for your hardware platform.

Hardware Platform	Linux Version
HP Z800	Custom Autodesk distribution of Red Hat Enterprise Linux Desktop 5.3 with Workstation Option
HP xw8400, HP xw9400, HP xw8600	Custom Autodesk distribution of Red Hat Enterprise Linux WS 4, Update 3

To determine the version of Red Hat Enterprise Linux your system is currently running, open a terminal and type:

cat /etc/redhat-release

The Linux version appears. For example:

```
Red Hat Enterprise Linux Client release 5.3 (Tikanga)
```

If your system is not currently running the correct version of Red Hat Enterprise Linux, install the required version of the operating system.

Linux Installation Workflow

Perform the following procedure to prepare your system and perform a fresh install of the customized distribution of Red Hat Enterprise Linux.

To install Linux:

1 Locate the installation media.

The DVD containing the customized Autodesk distribution of Red Hat Enterprise Linux is included with your shipment. The customized Autodesk distribution installs certain Linux packages that are required by Autodesk applications but are not installed by the commercial distribution of Red Hat Enterprise Linux.

WARNING Even though your shipment may also include the disc set for the commercial distribution of Red Hat Enterprise Linux, **DO NOT** install the commercial distribution. Your software only works on the custom Autodesk distribution of Red Hat Enterprise Linux.

2 Power off all storage enclosures connected to the system, to prevent the Red Hat installer from attempting to reformat them as part of the installation process.

WARNING Do not skip this step. Failure to power off or disconnect your storage before installing Linux may result in data loss.

NOTE Turn off the XR RAID controllers first, then the XE expansion enclosures.

3 If you are reinstalling Linux on an existing system, back up all user settings, project settings, and media from the system to another system or to a removable drive. The Linux installation erases and formats the system disk, so this information is lost if it is not backed up. In addition, back up the following directories:

■ **/usr/discreet** (for software setup and configuration files)

- **/etc** (for Linux configuration files, networking, etc)
- **/usr/lib/DPS** (for fonts)
- **/usr/local/flexlm/licenses** (for the license file)

4 Make sure your system is in the following state:

- Mouse, keyboard and graphics monitor are connected, and the graphics monitor is powered on.

- Other peripherals, such as the tablet and video breakout box are connected and powered on.

- If you are using a KVM switch, it is switched to the system on which you want to install Linux.

- The storage array is powered off.

- If a DVI-Ramp is part of your configuration, the DVI-Ramp is connected and powered on.

5 Insert the DVD of the custom Autodesk distribution of Red Hat Enterprise Linux in the DVD-ROM drive, and power off the system.

WARNING If your system is equipped with internal media storage, after turning the system off, open the side panel and disconnect the main connector for the internal storage drives. Make sure you do not disconnect the system drive!

6 Restart the system.

7 When the system starts to boot, press F10 to enter the BIOS and make sure the BIOS settings reflect the recommended values. The BIOS must be correctly configured **before** installing Linux.

NOTE If the keyboard is connected to the system through a KVM switch, you may have to press F10 repeatedly or hold it down to enter the BIOS.

8 Save the correct BIOS settings and reboot the system.

The system should boot to the Linux installation disc. If it does not, review your BIOS settings to make sure the DVD / CDROM drive is set as the primary boot device.

9 When the Red Hat Linux installation screen appears, press F8 to display the Autodesk Red Hat Linux installation menu.

10 At the boot prompt in the Red Hat Linux installation menu, type **ifffsbc** and then press Enter to launch the Linux installation.

WARNING You must use this exact command to start the Linux installation. If you use the default options presented by the Linux installation prompt, the installation proceeds, but some important packages required by Autodesk software are not installed.

The command launches the Linux installation. The system spends several minutes loading drivers. The installer guides you through the rest of the process.

11 The installation process may ask you to initialize the system disk. Follow the prompts to initialize the disk, if necessary.

12 The remainder of the installation process is automated.

When the installation completes, you are prompted with "Congratulations, the installation is complete".

13 Eject the disc and click Reboot to reboot the system. After the system reboots, perform the post-installation tasks in the following section.

NOTE After the system reboots, you may be prompted by the Kudzu hardware setup utility to set up new hardware detected for your system. You can ignore these prompts and allow the utility's count-down to expire since the hardware for the system is unchanged.

Linux Post-Installation Tasks

After booting into your new Linux installation, perform the following post-installation tasks:

1 Change the default root password to secure the system.
- Log into your system as *root*, using the default password *password*.
- Open a terminal and type:

  passwd
- Enter your current password when prompted, and then enter the new password.

2 Configure the time zone for your geographic location. The automated Autodesk installation sets the time zone to North American Eastern Standard Time (EST) by default.
- Log into the Linux Desktop as root.
- Right-click the clock, and select Configure Clock from the context menu.

- Click the Timezones tab, and select the city closest to you from the list of locations.

- Click Apply to save the new settings, and then click OK to close the window.

3 Configure the network settings for your system to match the ones used in your facility. The automated Linux installation sets a default IP address and hostname for your system, which may not be suitable for your facility network. See Configuring Basic Network Settings (page 37).

4 Install the Discreet Kernel Utilities (DKU), and upgrade the firmware of your AJA card.

Configuring Basic Network Settings

This section provides instructions on configuring your system network settings to match those of your facility.

Before configuring your network settings, gather the following information from your network administrator:

- An available IP address on your network, to assign to your system. The IP must not change over time, and must not be used by any other system on your network.

- A unique host name for your system.

- The IP address of the gateway system on your network.

- The subnet mask of your network.

- The IP addresses of one or more DNS servers.

The following files contain the network settings of your Linux system.

Networking Configuration File	Description
/etc/sysconfig/network	Contains global networking settings, such as the system host name, Gateway IP address, and NIS domain name, if applicable.
/etc/resolv.conf	Contains DNS server information.
/etc/hosts	Contains IP/host name pairs for host name resolution.

Networking Configuration File	Description
/etc/sysconfig/network-scripts/ifcfg-eth\<x\>, where \<x\> specifies the ethernet port number, usually 0 for the first on-board port.	Contains interface-specific configuration.

The following procedures provide instructions on setting up each configuration file. Log is as root before editing these files. When you are done changing the network settings, restart your system for the changes to take effect.

To configure the system hostname and default Gateway:

1 Open the *the /etc/sysconfig/network* file in a text editor.

2 Make sure the NETWORKING keyword is set to *yes*.

3 Configure the HOSTNAME keyword with the hostname you want to use for your system. For example:

HOSTNAME=workstation1

4 Configure the GATEWAY keyword with the IP address of the gateway system on your network. For example:

GATEWAY=192.168.0.1

This is the global default gateway that will be used by the system if no gateway is defined in a network port's individual configuration file.

5 Save and close the file.

To configure the system IP address and netmask:

1 Open the *the /etc/sysconfig/network-scripts/ifcfg-eth*\<x\> file in a text editor, where \<x\> is the number of the network port your system uses to connect to the facility network, typically 0.

NOTE In most cases, the network port *eth0* is the first on-board port on the back of your workstation. Note however that, if your workstation has an add-on Ethernet card installed, the Linux operating system may assign ports *eth0* through *eth3* to the add-on card. In such a case, the on-board Ethernet ports become *eth4* and *eth5*.

2 Make sure the BOOTPROTO keyword is set to *static*. Your software does not support DHCP network configurations.

BOOTPROTO=static

3 Set the `IPADDR` keyword to the IP address you want to use. Make sure this IP is not used by any other system on your network. For example:

```
IPADDR=192.168.0.100
```

> **NOTE** It is not recommended to change this IP address once you have installed your Autodesk software.

4 Set the `NETMASK` keyword to the netmask used by your network. For example:

```
NETMASK=255.255.255.0
```

5 Make sure the `ONBOOT` keyword is set to yes.

```
ONBOOT=yes
```

6 Optional: If you have not specified a global default gateway for your system, or if you want to use a different gateway for this particular network port, add a `GATEWAY` keyword in this file, and set it to the IP address of the gateway. For example:

```
GATEWAY=192.168.0.1
```

7 Save and close the configuration file.

To configure host name resolution:

1 Open the */etc/hosts* file in a text editor.

2 Locate the loopback setting in the *hosts* file. By default, the loopback setting is identified by the IP address 127.0.0.1, for example:

```
127.0.0.1 vxfhost.localhost.localdomain localhost
```

3 Remove the host name preceding `localhost.localdomain` from the loopback setting.

For example, if you remove the host name `vxfhost` in the example above, the loopback setting should appear as follows:

```
127.0.0.1 localhost.localdomain localhost
```

4 On a new line, type the IP address of the system, press `Tab`, and then enter the explicit host name of the system.

The new line defines the network IP address and host name of the system. For example, if the IP address is 192.168.0.100 and the host name is *workstation1*, your *hosts* file should be as follows:

```
127.0.0.1 localhost.localdomain localhost
192.168.0.100 workstation1
```

5 Optional: Add hostname / IP address pairs for other workstations on your network. For example:

```
192.168.0.150 flame1
```

6 Save and close the file.

To configure DNS server information:

1 Open the */etc/resolv.conf* file in a text editor.

2 Add a line for each DNS server IP address you want to use. The syntax is:

```
nameserver <DNS_server_IP>
```

For example:

```
nameserver 22.10.50.101
nameserver 22.10.50.102
```

NOTE Be careful when editing the *resolv.conf* file. The syntax is very strict, and any extra spaces or carriage returns can render the file unusable by the system.

Configuring an InfiniBand Card

If your workstation is equipped with a Mellanox QDR InfiniBand / 10 GigE adapter card, or a QLogic 9000 DDR InfiniBand adapter card, perform the procedures in this section to configure the IP address of the Infini|Band port.

NOTE Install the DKU before performing the procedures in this section. If the Mellanox or QLogic InfiniBand adapter card is present in the workstation when the DKU is installed, the drivers are automatically installed and configured by the DKU. If you add an InfiniBand card to your workstation after having installed the DKU, run the DKU installer again to configure the required drivers.

To configure the IP address for a Mellanox QDR InfiniBand card:

1 If the InfiniBand card was not installed into your workstation when you last upgraded the DKU, re-run the DKU installation script to setup the drivers for the InfiniBand card.

2 Open a terminal and log in as root.

3 Navigate to the OFED Distribution directory:

cd /usr/discreet/DKU/InfiniBand/ofed*

4 Run the OFED Software Installation script:

./install.pl

5 Choose option **4** in the main menu.

6 Type **Y** when asked if you want to configure ib0.

The current configuration is displayed. For example:

```
DEVICE=ib0
BOOTPROTO=static
IPADDR=10.10.11.3
NETMASK=255.255.255.0
NETWORK=10.10.11.0
BROADCAST=10.10.11.255
ONBOOT=yes
```

7 Type **Y** to change the configuraton.

8 Enter the new IP address, netmask, broadcast address and base network address for the InfiniBand port.

The settings are saved in the *etc/sysconfig/network-scripts/ifcfg-ib0* file, and you are returned to the main menu.

9 Type **Q** to exit the configuration utility.

10 Reboot your workstation.

To configure the IP address for a QLogic DDR InfiniBand card:

1 If the InfiniBand card was not installed into your workstation when you last upgraded the DKU, re-run the DKU installation script to setup the drivers for the InfiniBand card.

2 Open a terminal and log in as root.

3 Run the Silverstorm InfiniBand configuration utility:

/sbin/iba_config

4 Choose option **2** in the main menu.

5 Follow the on-screen instructions to configure the IP address of the InfiniBand card.

The settings are saved in the *etc/sysconfig/network-scripts/ifcfg-ib1* file, and you are returned to the main menu.

6 Type **X** to exit the configuration utility.

7 Reboot your workstation.

Installing the DKU and the AJA OEM-2K Firmware

Mounting the Product USB Key

For major releases, all your product software is distributed on a USB key.

You must mount the USB key before you can install the product software.

NOTE If Red Hat is newly installed (or re-installed), automount is enabled. When the USB key is connected to the system, the USB device is mounted automatically. However, you cannot install the DKU because the Red Hat installation does not allow for the execution of commands from a USB device. Before you install the software, you must unmount the USB key and then follow the instructions below for mounting this USB device. To unmount the USB key, type **eject /<mountpoint> or eject /dev/<device id>** .

To mount the product USB key:

1 Plug in the USB key to your workstation.

2 Log in to your workstation as root and open a terminal.

3 List the disk devices attached to your system by typing:

   fdisk -l | grep "/dev/sd"

   In the output of the command, identify the SCSI device name that was assigned to the USB device. It should look similar to */dev/sds* , */dev/sde* , etc.

   For example:

   ```
   Disk /dev/sda: 250.0 GB, 250059350016 bytes
   /dev/sda1 * 63 208844 104391 83 Linux
   /dev/sda2 208845 4401809 2096482+ 82 Linux swap / Solaris
   /dev/sda3 4401810 488392064 241995127+ 83 Linux
   ...
   ...
   Disk /dev/sdf: 8011 MB, 8011120640 bytes
   /dev/sdf1 62 12128687 6064313 c W95 FAT32 (LBA)
   ```

4 Alternatively, you can use the *dmesg* command right after you have connected the USB drive. This will list the most recent USB device connected. For example:

```
usb-storage: device found at 30
usb-storage: waiting for device to settle before scanning
Vendor: TOSHIBA Model: TOSHIBA USB DRV Rev: PMAP
Type: Direct-Access ANSI SCSI revision: 00
SCSI device sdf: 31277056 512-byte hdwr sectors (16014
MB)
sdf: Write Protect is on
sdf: Mode Sense: 03 41 80 00
sdf: assuming drive cache: write through
SCSI device sdf: 31277056 512-byte hdwr sectors (16014
MB)
sdf: Write Protect is on
sdf: Mode Sense: 03 41 80 00
sdf: assuming drive cache: write through
sdf: sdf1
```

5   In the example above, the sdf device was the last device to be connected.

6   Create a mount point directory for the disk. Type:

   mkdir /mnt/usbdisk

7   Mount the disk on the new directory:

   mount -o shortname=winnt /dev/<device>1 /mnt/usbdisk

   where <device>1 is the usb device name. For example:

   mount -o shortname=winnt /dev/sdf1 /mnt/usbdisk

NOTE You can also mount the disk using the GUI. Double-click the Home icon on the desktop, go to the root folder, expand the 'dev' folder, right-click the USB device and select 'Mount' from the context menu.

Extracting an Installer from a TAR Package

At extensions and service packs, your product software packages are distributed as TAR files. Links to the TAR files for your product are supplied in the Release Announcement.

You must extract an installer from the TAR package before you can install your software.

1   Log in to your workstation as root and open a terminal.

2 Download the *tar* file from Autodesk.

3 After downloading the *tar* file, verify its integrity using the Linux md5 checksum. In the directory where you saved the *tar* file and the checksum file, type **md5sum** `<filename.tar.gz>`, and make sure the checksum displayed matches the Linux md5sum listed in the checksum file.

4 Unpack the downloaded application *tar* file to a temporary directory:

tar -zxvf`<filename.tar.gz>`

The file is uncompressed and an installation directory is created on your system.

Installing the DKU

Before installing your software, you must install the required version of the DKU. See the Release Notes for the required DKU version for this version.

For major releases, the DKU is available on optical media or as a download from Autodesk. For extensions and service packs, the DKU is only available for download. The download link is provided in the release announcement you received from Autodesk.

To install the DKU:

1 Open a terminal and log in as root.

2 If you are upgrading an existing application, check the currently installed DKU version by typing:

head -n1 /etc/DKUversion

If the DKU version output by the command does not match the version required for the new application version, perform the remaining steps in this procedure.

3 Mount the USB key (page 42) or extract the tar file (page 43) to access the DKU installation directory.

4 Go to the DKU installation directory (*/mnt/usbdisk/DKU-<version>*, if you are installing from the application USB key) and launch the DKU installation script:

./INSTALL_DKU

WARNING If you attempt to install the DKU on an unsupported platform, the installation script only installs a generic configuration. Such a configuration can render your operating system unusable. Never attempt to install the current version of the DKU on unsupported platforms, including older Creative Finishing systems that are no longer supported in the current version.

When the DKU installation script completes, a warning to update the AJA card or DVI-Ramp firmware may appear and you are returned to the command prompt. See the following section for information on updating your AJA card firmware.

5   If you installed from a disc, return to the root directory, and eject the disc by typing:

eject

6   Before rebooting the system, make sure that the Wacom tablet is connected. If storage devices (external disk arrays or internal storage) were previously disconnected or turned off as part of an operating system installation, reconnect them and power them up.

NOTE Power on the XE expansion enclosures first and the XR RAID controller units last. This ensures the RAID controllers detect the other units in the Stone Direct storage.

Wait for all the disks in the storage devices to fully power up before rebooting the system. The green light for each disk stops flashing once it is fully powered up.

7   Reboot the system. Type:

reboot

NOTE After the workstation reboots and until you install the application, the Wacom tablet might not behave correctly. Installing the application, then logging out of the KDE desktop and logging back in should eliminate this problem.

Upgrading the AJA Firmware

The AJA firmware update utility is distributed with the DKU. It checks the firmware version and upgrades the firmware if necessary.

NOTE Before upgrading the AJA firmware, make sure you have upgraded the DKU, and that you have restarted the system.

To verify and upgrade the AJA firmware:

1 Open a terminal, log in as root.

2 Go to the */usr/discreet/DKU/current/Utils/AJA_firmwareUpdate* directory.

3 Run the *AJAfw_update* utility to scan the AJA current firmware and, if required, update to the latest firmware version. Type:

   ./AJAfw_update

4 If you have an AJA OEM-2K card, you must start the firmware update by typing Y and then pressing Enter.

   If you have a AJA Kona 3G, the upgrade begins automatically.

   While the AJA firmware and drivers are being updated, your workstation appears to be frozen and your mouse and keyboard do not work. This is normal and indicates that the firmware is being updated. Once the firmware update is complete, you are returned to the terminal.

5 Shut down your workstation by typing:

   shutdown -g0

   If your workstation does not prompt you to power down, press the power button for 10 seconds to force a power down.

6 Disconnect the power cord.

7 Wait 10 seconds, reconnect the power cord, then restart your workstation.

Configuring Storage

Overview

This chapter describes how to configure a direct-attached storage array connected to the workstation.

If you are also configuring a Storage Area Network (SAN), it is recommended that you configure the SAN prior to configuring a direct-attached storage.

Before you can configure the storage, the workstation must be up and running the custom Autodesk distribution of Red Hat Enterprise Linux, as well as the version of the DKU required for your application version.

WARNING When configuring a storage array, it is strongly recommended to disconnect any other storage device (DAS or SAN) connected to the workstationso that you do not reconfigure the wrong device.

Do I Need to Perform these Procedures?

If you are reinstalling your system from scratch, or adding new storage devices, you must perform these procedures.

If you are only upgrading an existing application installation without adding new storage, skip to the next chapter for instructions on installing the software.

Powering Up Your Storage

Powering your system and storage up or down should be done in a proper sequence. This ensures that the system functions properly.

WARNING An incorrect power up sequence can mean your system does not recognize all drives.

NOTE If you are configuring a storage array with two sets of enclosures, connect only one set of enclosures at this point, and then perform the filesystem creation procedures in the next section. Then disconnect the storage, and repeat the entire process for the second set of enclosures.

To power up your system:

1 Make sure your workstation is shut down.
2 Power up the XE expansion enclosures.
3 Power up the XR RAID controller enclosures.
4 Wait about 90 seconds for all the drives to spin up. Their lights are solid green when they are spun up.
5 Power up your workstation.

To power down your system, shut down your workstation first, then the XR RAID controller enclosures, and finally the XE expansion enclosures.

Partitioning the Storage and Creating the XFS Filesystem

Before you can use a new storage array, you need to create logical units (LUNs) on it, partition them, group the partitions into a logical volume, and then create the XFS filesystem on the logical volume.

All these operations can be easily performed using version 1.5 of the *XR Configuration Utility* from Autodesk. This version of the *XR Configuration Utility* is available in versions 4.5.1 and newer of the *Discreet Kernel Utilities* (DKU).

If your workstation is using an older version of the DKU that does not have the latest *XR Configuration Utility*, or if you prefer to manually partition your storage array and configure the filesystem, refer to Overview (page 184).

Perform one of the following procedures to configure your storage using the *XR Configuration Utility*. The procedures are different for storage arrays with a single set of enclosure, and for arrays with two sets of enclosures.

Note that the *XR Configuration Utility* exits without configuring your storage if any of the following issues are detected:

■ The total number of disks is not a multiple of 12.

■ One of more of the enclosures do not have the correct firmware.

■ When using two sets of enclosures, the number of expansion chassis attached to each XR RAID enclosure is not the same.

■ An odd number of enclosures in a 4-loop configuration. Only even numbers of enclosures are supported.

To configure arrays with a single set of enclosure:

1 Open a terminal and log in as root.

2 Go to the */usr/discreet/DKU/current/Utils/Storage/XR/* directory, and run the *XR Configuration Utility* by typing:

   ./XR_config.pl

   The script displays information on the storage, such as: enclosure type, RAID controller firmware version, number of enclosures, number of loops.

3 If a LUN configuration already exists on the storage, you are prompted for confirmation to overwrite that configuration.

   WARNING LUN configuration is destructive. Make sure you want to overwrite an existing configuration before you confirm.

4   After the utility detects the number of enclosures and drives, it presents you with a list of options.

5   Type **2** and press `Enter`.

This option instructs the configuration utility to create LUNs with a sector size of 512 bytes. This is the optimal sector size for XFS DAS (direct-attached) storage of Creative Finishing applications.

6   When asked if you have a 2-loop or a 4-loop configuration, select the option that applies to your storage. Note that 4-loop configurations are only supported for XR 6412 and XR 6500 storage.

The utility creates LUNs on your storage. This process might take a few minutes.

When the LUN creation process completes, the configuration utility asks you if you want to create the XFS filesystem on the storage array.

7   Type **Y** (the default answer).

A list of detected devices appears.

8   Type **Yes** to confirm.

The configuration utility partitions the storage array, groups the partitions into a logical volume, creates the XFS filesystem on the volume, and mounts the volume to */mnt/StorageMedia* on HP Z800 workstations, or */mnt/stoneMedia* on older workstations.

NOTE If the storage driver does not reinitialize quickly enough after LUN creation, the filesystem creation step may fail, and the *XR Configuration Utility* gives you an error message similar to the following:

`Can't open /dev/sdb1 exclusively. Mounted filesystem?`

The filesystem will appear mounted. Type **pvdisplay -v** in the terminal to verify that the physical volume, volume group and logical volume were created.

If the physical volume, volume group and logical volume were not created, reboot your workstation, and perform the steps in .

9   Type **x** to exit the configuration utility.

NOTE Even though your storage array is usable at this point, the drives are still initializing following the LUN creation. Your storage operates at reduced performance until the initialization process completes. This might take a few hours. To check the progress of the initialization type:

./XR_config.pl --arrayStatus

10 Once the initialization process completes, create a back-up of the storage configuration by typing:

./XR_config.pl --saveConfig

A back-up of the storage configuration is saved.

To configure arrays with two XR enclosures:

1 Open a terminal and log in as root.

2 Make sure only one set of enclosures is connected to the workstation.

3 Go to the */usr/discreet/DKU/current/Utils/Storage/current/* directory, and run the *XR Configuration Utility* by typing:

./XR_config.pl

The script displays information on the storage, such as: enclosure type, RAID controller firmware version, number of enclosures, number of loops.

4 If a LUN configuration already exists on the storage, you are prompted for confirmation to overwrite that configuration.

WARNING LUN configuration is destructive. Make sure you want to overwrite an existing configuration before you confirm.

5 After the utility detects the number of enclosures and drives, it presents you with a list of options.

6 Type **2** and press `Enter`.

This option instructs the configuration utility to create LUNs with a sector size of 512 bytes. This is the optimal sector size for XFS DAS (direct-attached) storage of Creative Finishing applications.

7 When asked if you have a 2-loop or a 4-loop configuration, select the option that applies to your storage. Note that 4-loop configurations are only supported for XR 6412 and XR 6500 series storage.

The utility creates LUNs on your storage. This process might take a few minutes.

When the LUN creation process completes, the configuration utility asks you if you want to create the XFS filesystem on the storage array.

8 Type **N**.

NOTE If you answered *Y* by mistake (the dealt answer), you get another chance to answer *no* after the system rescans the disks. It is very important that you do not create the filesystem at this point.

9 Type **x** to exit the configuration utility.

10 Once the initialization process completes, create a back-up of the configuration for this set of enclosures by typing:

./XR_config.pl --saveConfig

A back-up of the configuration is saved. Note that you can only save the configuration for one set of enclosures at a time.

11 Power off the workstation, disconnect the storage enclosures, and connect the second set of enclosures.

12 Power the workstation back on.

13 Perform steps 3 through 10 in this procedure on the second set of enclosures.

14 Power off the workstation again, connect both sets of enclosures to the workstation.

15 Power the workstation back on and perform the tasks in the following procedure.

Use the following procedure if you need to perform just the partitioning and filesystem creation operations on a storage array where LUNs have already been created.

To perform only the filesystem creation operations:

1 Open a terminal and log in as root.

2 Go to */usr/discreet/DKU/current/Utils/Storage/current/*, and type:

./XR_config.pl --createFS

The configuration utility partitions the storage array, groups the partitions into a logical volume, creates the XFS filesystem on the volume, and mounts the volume.

NOTE Using this parameter has the same effect as running the *XR Configuration Utility* with no parameter, and choosing option *4* from the main menu.

3 Type **Yes** to confirm.

4 Create a back-up of the storage configuration by typing:

./XR_config.pl --saveConfig

NOTE You can only back-up your storage configuration after the drives have finished initializing. This process might take a few hours. To check the progress of the initialization type:

./XR_config.pl --arrayStatus

To get a complete list of the command-line parameters available for the *XR_config.pl* script, type:

./XR_config.pl --help

Installing your Software

3

Installing Flame Premium or Visual Effects and Finishing Standalone Software

Use the following procedure to install or upgrade your software.

> **NOTE** To ensure optimal performance, it is recommended that only Autodesk Creative Finishing applications and required Red Hat Enterprise Linux packages be installed on the workstations.

To install your software:

1 Log in to your workstation as root and open a terminal.

2 Archive existing media on the framestore (highly recommended). Refer to the "Archiving" topic in the application help.

3 If you need to change your system date or time, do it **before** installing the application.

4 Go to the application installation directory. You must mount the USB key (page 42) or extract the tar file (page 43) to access the install directory.

5 Start the application installation script by typing:

./INSTALL_<application_name>

where <application_name> is in upper case.

For example, if you are installing Flame Premium, type:

./INSTALL_FLAMEPREMIUM

Or, if you are installing Smoke, type:

./INSTALL_SMOKE

NOTE If you are installing an educational version, start the initialization script by typing **./INSTALL**_<application_name> --edu, for example **./INSTALL_SMOKE --edu**

The installation script verifies the installation packages and checks your system for previously installed versions of the application. This process may take a few minutes.

6 If you are upgrading from a previous version of the application, earlier versions are detected and the installer asks you to select one of those versions. Do one of the following:

■ Click None if you would like a clean installation of the application.

■ Click one of the previous versions if you want to copy its custom resource and setup directories to the new version.

7 The installation script asks you if you want to create an application user:

■ Answer Yes if you want to create a Linux user account for the application, or No if you want to run the application with an existing user account.

■ If you answered No, a list of all the existing users in the system is displayed. Select the users accounts you plan to use the application with.

TIP You can use CTRL+click to select multiple entries.

The application icon is placed on the Linux Desktop of the selected users.

8 When prompted to configure the software initialisation file (*init.cfg*), click OK.

The software initialisation file contains general settings the application reads on startup.

■ If you are installing the application for the first time, the *init.cfg* file appears in a text editor.

■ If you are upgrading the application, the old (previous) and new *init.cfg* files appear beside each other in an *xxdiff* window. The left panel displays the old *init.cfg* file. The right panel displays the new *init.cfg* that is installed with your application. The vertical bar on the right side indicates where differences exist between the two panels.

9 Scroll through the *init.cfg* file, verify the keywords, and modify their values as necessary. It is recommended to verify at least the following keywords to make sure they reflect your hardware configuration.

Keyword	Setting
Video	The video card used by your workstation. It can take the following values: AJA_KONA3G AJA_OEM2K See Video (page 129) for details about the keyword.
DVIRamp	The DVIRamp keyword is applicable only if your workstation is equipped with a Miranda DVI-Ramp. See DviRamp (page 130) for details about the keyword.
VideoPreviewDevice	The VideoPreviewDevice keyword is applicable only if your workstation is equipped with a Miranda DVI-Ramp or an NVIDIA® SDI card. You can use the VideoPreviewDevice keyword to enable alternate SD or HD broadcast monitors. See VideoPreviewDevice (page 131) for details about the keyword.
TabletDriver	`wacom`® See TabletDriver (page 133) for details about the keyword.
VTR	Enable the VTR keywords that have settings corresponding to the VTRs you intend to use. See Vtr (page 134) for details about the keyword.
Audiodevice	`AJA` See Audiodevice (page 138) for details about the keyword.

10 To modify the *init.cfg* file, do one of the following:
- If the file opened in a text editor, edit the values as needed, then save and exit the *init.cfg* file.

- If you are in *xxdiff*, each difference between the old and the new *init.cfg* files is highlighted. Click the correct value for each keyword, regardless of whether it is in the left or right panel. Make sure you select a value for each highlighted difference. Then open the File menu and choose Save as Right. After the file is saved, close *xxdiff*.

11 If any changes are detected in the *etc/X11/xorg.conf* file, you are prompted to configure this file:

■ To keep the old configuration settings, when the file appears in the *xxdiff* editor, in the Global menu, choose Select Left, and then, in the File menu, choose Save as Right. If prompted to overwrite, click OK.

■ To use the new configuration file without adding the settings from the old configuration file, just close the program.

WARNING The *etc/X11/xorg.conf* file contains specific parameters for the graphics display, tablet, and other external devices. It is customized prior to shipping to reflect your configuration. It is strongly recommended that you never manually edit this file, as incorrect entries may result in peripherals that do not function properly, or an inability to launch the user interface. Use *xxdiff* to move entries from one version of the file to another.

12 If you modified the *xorg.conf* file, you receive a message to restart the X server. When the installation script completes, log out of the Linux desktop and then log in again to restart the X server.

The application, as well as additional components (WiretapCentral™, Wiretap® Gateway, Backburner™ Server, Backburner Manager, Backburner Media I/O Adapter) are installed on your workstation.

NOTE By default, Backburner™ Server and Backburner Manager are set to automatically run on the workstation. The manager for the local Backburner Server is set to *localhost*. You can reconfigure these services after the application is installed. See Configuring Backburner Services (page 70) for instructions.

13 Prior to licensing and starting the application for the first time, perform the post-installation procedures necessary for your system. See Installing Flame Premium or Visual Effects and Finishing Standalone Software (page 53).

Installing Lustre Software

Lustre is installed during the installation of Flame Premium. If you have Flame Premium, see Installing Flame Premium (page 53).

To install Lustre software:

1 Open a terminal, and log in as root.

2 If you need to change your system date or time, do it **before** installing the application.

3 Go to the application installation directory. You must mount the USB key (page 42) or extract the tar file (page 43) to access the install directory.

4 Start the application installation script by typing:

./INSTALL_LUSTRE

5 If prompted with a message asking if you would like to automatically run Backburner™ Server on this machine, click Yes.

The Lustre application, as well as supporting software (Backburner, Wiretap, Wiretap® Gateway, the *BrowseD* server) are installed. A new user account is created for the version of Lustre you just installed. Read the following chapter for information on licensing and starting your application for the first time.

NOTE If you are upgrading to a service pack of the same release or extension, you do not need to re-license your software.

6 Prior to licensing and starting the application for the first time, perform the post-installation procedures necessary for your system.

Uninstalling your Software

The *Autodesk Software Remover* utility enables you to easily uninstall Autodesk Creative Finishing applications from your workstation.

To uninstall your application:

1 If you are logged in as the application user in KDE, log out and log back into KDE as root.

2 From the KDE menu, select Autodesk > Remove Software.

The user interface appears.
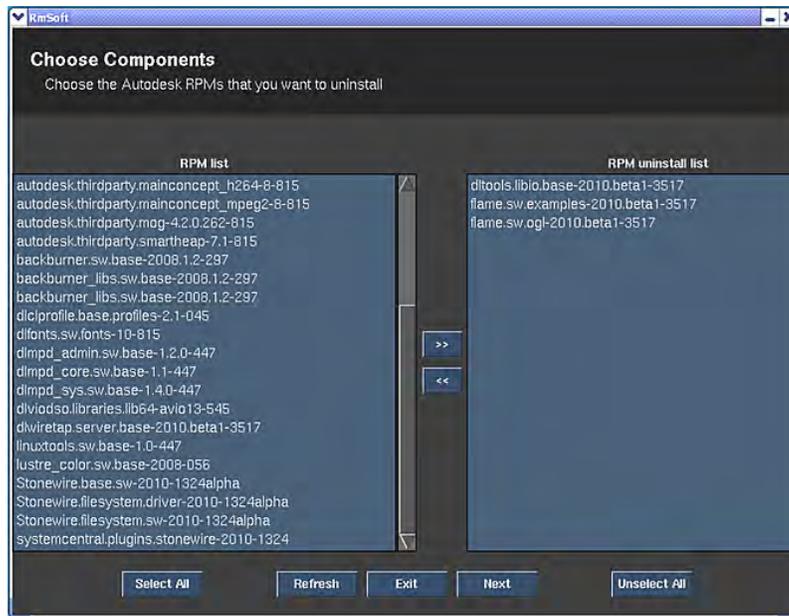
3 Select the packages you want to uninstall in the RPM list on the left

(click Select All to select all the packages), then click [>>] to move them to the RPM uninstall list on the right.

4 Click Next.

The Choose folders window appears.

5 Select the application directories you want to remove from the */usr/discreet* directory, and click Next.

A confirmation dialog appears.

6 Click Uninstall & Remove to confirm the removal of the selected packages and directories.

The uninstallation starts and displays details on the process.

7 When the operation completes, click Exit to close the Autodesk Software Remover utility.

8 Optional: You can also delete the log files associated with a given application version in the */usr/discreet/log* directory.

Configuring Your Software

4

Configuring Visual Effects and Finishing Software

Do I Need to Perform These Tasks?

NOTE The tasks in this chapter do not apply to Lustre.

If you installed your application from scratch, perform the procedures in this chapter to configure your application to access the media storage, and to use the Wire network.

If you are upgrading an existing application, you have already performed most of the procedures in this chapter during the original installation. You do not need to repeat these procedures, unless you are adding new media storage devices to your workstation.

The only task you might need to perform when upgrading from a previous version is to transfer your projects and clip libraries to the new version. See Upgrading Visual Effects and Finishing Projects to the Current Version (page 68) and Transferring Clip Libraries to 2012 Projects (page 69).

Configuring Media Storage

You can use your application with the following storage configurations:

- A standard UNIX-compatible filesystem on a centralised SAN in your facility. See Configuring a Direct Attached Storage or SAN as Media Storage (page 60).

- A standard UNIX-compatible filesystem on a direct attached storage (DAS) device. See Configuring a Direct Attached Storage or SAN as Media Storage (page 60).
- A standard UNIX-compatible filesystem on a Network Attached Storage (NAS) based on the Network File System (NFS) protocol.

  NOTE The fact that all I/O operations occur over the network makes it impossible to reliably predict steady performance for NAS filesystems. Note also that the NFS protocol features built-in cache and time-out mechanisms, which may cause several workstations working on the same media source to appear de-synchronized.

Configuring a Direct Attached Storage or SAN as Media Storage

This section provides an overview of the steps involved in defining a standard UNIX®-compatible filesystem to be used as media storage by your application.

Media can be stored on any of the following hardware, *as long as it is not the system drive*:

- A direct attached storage (DAS) device, such as an Autodesk Stone Direct XR-series disk array
- A storage area network (SAN)

  NOTE When attached to a CXFS SAN declared as a standard filesystem partition to Stone and Wire, Visual Effects and Finishing workstations running the current release have shown optimal (real-time) performance with version 4.02 of the CXFS client, and the following mount options for the CXFS volume:

  ```
  rw,noatime,filestream,inode64
  ```

Autodesk recommends using a properly configured and maintained XFS filesystem installed on Stone Direct XR-series disk arrays.

Concepts and Terminology

If this is the first time you are configuring an Autodesk Visual Effects and Finishing filesystem, familiarize yourself with the following key concepts.

If you are already familiar with these concepts, continue with Configuring your Application to Use the Media Storage Filesystem (page 62).

Stone and Wire	The software package that encompasses local management of media and the transferring of media between Visual Effects and Finishing workstations. This package is installed automatically with the application.
Partition	In Stone and Wire, a partition (also referred to as "volume") is defined as a volume of media storage. When creating a project, the Visual Effects and Finishing applications permanently associate it to one of the available partitions. This association means that the project inherits the rules and media preferences of the partition. You can define up to eight partitions.
Managed Media	Media is said to be managed when the media assets, typically DPX files, are managed or "owned" by the Visual Effects and Finishing application. The application deletes managed media it no longer needs. All managed media is stored in a Managed Media Cache.
Unmanaged Media	Unmanaged media refers to media assets that are used by, but not exclusively owned by, the Visual Effects and Finishing application. Soft-imported clips are an example of unmanaged media. This is relevant in shared access workflows, where numerous applications are using the same media. Unmanaged media is used in projects and clip libraries in the same way as managed media. However, the application does not delete it when it is no longer needed.
Managed Media Cache	A directory residing on a standard FS volume, to which all managed media is written. This includes intermediates, imported media, captured video, audio, and proxies. Although the Managed Media Cache can be accessed by standard operating system tools such as defragmentation and backup tools, it is not meant to be accessed by any other application or user than the Visual Effects and Finishing and Wiretap applications.
Media Database	The standard filesystem media database plays a central role in the management of media residing on standard filesystems. It is responsible for mapping frame IDs to the location of the media on disk. Database entries are created for all managed media and soft-imported media. There is one standard filesystem media database file per volume.
Clip library	A clip library is a catalog maintained locally by its Visual Effects and Finishing application. It is used, internally by the

application, to store clip metadata, including frame IDs. The artist acts upon portions of it indirectly, via the application User Interface.

Configuring your Application to Use the Media Storage Filesystem

To set up a standard filesystem as media storage:

1 Make sure the disk array or SAN storage is connected to your workstation and powered up, and that a UNIX-compatible filesystem exists on the storage. See Partitioning the Storage and Creating the XFS Filesystem (page 48) for details on formatting your storage array to the recommended XFS filesystem.

2 Open a terminal and log in as root.

3 Stop Stone and Wire with the command:

/etc/init.d/stone+wire stop

Messages appear in the terminal indicating that Stone and Wire is stopped.

4 Perform the following steps to create the Managed Media Cache directory:

■ If a mount point for your storage does not exist, create a directory that will serve as the mount point. For example:

mkdir -p /mnt/StorageMedia

or

mkdir -p /mnt/SAN1

NOTE If you are configuring a direct-attached storage array, the DKU installer should have already created a mount point for it under */mnt/StorageMedia* (for HP Z800 workstations) or */mnt/stoneMedia* (for older workstations).

WARNING Do not use the word "stonefs" as the name for your mount point directory. "Stonefs" is a reserved word, and can create issues if used as the mount point directory name.

■ Mount the filesystem to the newly created directory. For example:

mount -av -t xfs -o rw,noatime,inode64 /dev/vg00/lvol1 /mnt/StorageMedia

Type **man mount** in a terminal for more information on mounting filesystems.

NOTE To set the filesystem to automatically mount at boot, create an entry for it in the */etc/fstab* file. Type **man fstab** in a terminal for details.

■ Create the Managed Media Cache directory on the mounted filesystem, in a directory named after the workstation hostname. The Managed Media Cache directory should be named after the partition name (by default, partition 7, or *p7*).

For example, if the filesystem mount point is */mnt/StorageMedia*, your workstation hostname is *smoke1*, and the partition name is *p7*, type:

mkdir -p /mnt/StorageMedia/smoke1/p7

■ Set the ownership for the directory created in the previous step to the root user. For example:

chown -R root:users /mnt/StorageMedia/smoke1/p7

■ Set permissions for the directory to be readable, writable, and searchable by the owner and its group, and readable and searchable by all. For example:

chmod -R 775 /mnt/StorageMedia/smoke1/p7

5 Optional: If you want to use several media storage volumes (for example, additional direct-attached storage arrays, or SAN volumes), repeat step 3 to create separate mount points and Managed Media Cache directories for each additional volume. The DKU only creates the mount point and */etc/fstab* entry for the initial volume (*/dev/vg00/lvol1*).

6 To make your application aware of the standard filesystem volumes, define the Managed Media Cache of each volume as a partition in the Stone and Wire configuration file, and set its preferences:

■ Open the file */usr/discreet/sw/cfg/stone+wire.cfg* in a text editor.

The settings for each partition are defined in a `[Partition<partition_number>]` section of the file. For example `[Partition7]`.

NOTE By default, the *stone+wire.cfg* file contains only one `[Partition]` section. If you are using multiple volumes, create copies of the `[Partition]` section for each volume, and configure them accordingly.

■ Uncomment the `Name` keyword and enter a meaningful name for the volume. For example:

`Name=XFS_partition`

This name identifies your storage volume in the application.

■ Uncomment the `Path` keyword and specify the path to the Managed Media Cache directory on the mounted volume. For example:

```
Path=/mnt/StorageMedia/smoke1/p7
```

■ Optional: Flag the partition as shared by uncommenting the `Shared` keyword and setting it to `True`.

   NOTE Always set the `Shared` keyword to `True` for a SAN volume.

■ Optional: In the `[DefaultFileFormats]` section, specify the preferred file format for each bit depth. Uncomment the desired bit depth and set the file format to one of the supported formats. The frames for each bit depth will be saved to the media storage using the file formats specified here (for example *DPX* or *JPG*).

   NOTE The settings in the `[DefaultFileFormats]` section apply to all volumes. If you want to customize file format settings for certain volumes, copy the file format lines to the `[Partition]` section for the respective volumes. Settings defined in a `[Partition]` section overwrite the settings in the `[DefaultFileFormats]` section.

■ Optional: If using the *JPG* file format, uncomment the `JpegCompressionFactor` keyword and set JPEG compression to a value between `0` and `100`.

■ Optional: To prevent the use of symbolic links across filesystems, uncomment the `SymlinkAcrossFilesystems=False` keyword.

■ Save and close the *stone+wire.cfg* file.

7 If this is the first filesystem you are configuring for this workstation, perform the following additional steps:

■ Type the following command to display the contents of the */usr/discreet/sw/cfg/sw_framestore_map* file:

   cat /usr/discreet/sw/cfg/sw_framestore_map

■ Locate the ID value in the `FRAMESTORE` line of the command output. For example:

   ```
   FRAMESTORE=smoke1 HADDR=192.168.1.152 ID=152
   ```

■ Write down the ID number.

■ Open */usr/discreet/sw/cfg/sw_storage.cfg* in a text editor (create it if it does not exist) and enter the ID value you wrote down in the previous step. For example, if the ID value is 152, the *sw_storage.cfg* file should now look like this:

   ```
   [Framestore]
   ID=152
   ```

■ Save and close *sw_storage.cfg*.

8 Optional: To prevent other Visual Effects and Finishing tools and applications (such as background Wire, Wiretap™, or remote Visual Effects and Finishing applications) from impacting the performance of your local application, configure bandwidth reservation for your partition in the *usr/discreet/sw/cfg/sw_bwmgr.cfg* file. For details and examples, see Configuring Bandwidth Reservation (page 203) and the comments inside the */usr/discreet/sw/cfg/sw_bwmgr.cfg.sample* file.

9 Restart Stone and Wire by typing:

/etc/init.d/stone+wire restart

A message appears indicating Stone and Wire has restarted.

10 Make sure the filesystem is mounted. Type:

/usr/discreet/sw/sw_df

A report appears providing information about the total, free, and used disk space in the partition.

Any project created in your application after this point has the new standard filesystem partition as its primary partition. Any pre-existing projects are unaffected by this setting and remain associated to their respective partitions.

Setting up the Wire Network

This section provides instructions for setting up the Wire network and associated components, such as Wiretap.

Wire is a component that enables the high-speed transfer of uncompressed video, film, and audio between Autodesk workstations, over industry-standard TCP/IP and InfiniBand networks.

Wiretap is a networking protocol that allows products that do not run Stone and Wire to access the filesystem of a Visual Effects and Finishing product.

Perform the following sequence of procedures to configure your Wire and Wiretap settings. You must stop all Stone and Wire daemons before editing any of the following configuration files.

To stop Stone and Wire:

1 Open a terminal and log in as root.

2 Stop Stone and Wire services by typing:

/etc/init.d/stone+wire stop

To specify the network interfaces to be used by Wire:

1 Open the */usr/discreet/sw/cfg/sw_framestore_map* file in a text editor.

2 Specify the network interfaces that you will use for Wire networking in the `[INTERFACES]` section. This task must be performed on every host that is a part of the Wire network.

   NOTE The interfaces are listed in the order of preference. Wire will try to connect to other hosts using the first interface in the list first.

   For example:

   ```
   [INTERFACES]
   FRAMESTORE=kamloops
   PROT=IB_SDP IADDR=192.168.0.1 DEV=1
   PROT=TCP IADDR=10.0.0.5 DEV=1
   ```

   ■ The `FRAMESTORE` keyword sets the name that identifies the framestore when remote hosts attempt to connect to it.

   ■ The `PROT` keyword defines the network protocol used to communicate to the framestore: `TCP` for TCP/IP or `IB_SDP` for InfiniBand.

   ■ The `IADDR` keyword specifies the IP address of the network card.

      NOTE If you are using an InfiniBand card, make sure the IADDR is different from the host address (HADDR) of the card, otherwise, the workstation sends house network traffic through the InfiniBand network and impairs Wire transfers.

To configure Wire self-discovery:

1 Open the */usr/discreet/sw/cfg/sw_probed_cfg* file in a text editor.

2 Make sure the value of the `Port` keyword matches the value used by the other workstations in your network.

3 Make sure the `SelfDiscovery` parameter is set to "yes".

   If these values are set properly, Stone and Wire uses its self-discovery mechanism to locate all other Wire hosts on your network and make their framestores accessible to your application.

To enable remote access to clip libraries, projects, users and setups:

1 Open the */etc/exports* file in a text editor.

2 specify all directories you want remote hosts to be able to access.

You can export the entire */usr/discreet* directory to enable remote access to all clip libraries, projects, user preferences and setups. Or you can export only certain subdirectories, such as */usr/discreet/clip* or */usr/discreet/project*.

For example, to export the */usr/discreet/clip* directory, add the following line to the end of the *exports* file:

```
/usr/discreet/clip *(rw, sync, no_root_squash)
```

3 After saving the updated exports file, type the following command in the terminal to apply the changes:

exportfs -a

To configure the Wiretap server:

1 Open the */usr/discreet/wiretap/cfg/wiretapd.cfg* file in a text editor.

2 Locate the `[Server]` section of the file.

That section contains the IP addresses of all the network interfaces that will be published for self-discovery. The house network IP address should already be present in the file, as it is added automatically during installation.

3 Specify any additional IP addresses Wiretap should broadcast for self-discovery, besides the house network IP. For example:

```
IP1=10.10.0.1
IP2=10.10.11.200
```

4 Optional: If your network comprises Windows® or Mac OS® X systems that will use Wiretap to access media from your workstation, configure the Wiretap Path Translation service so that media paths are accurately translated between operating systems. See About Wiretap Path Translation (page 193).

To restart and test Wire:

1 To restart Stone and Wire, type:

/etc/init.d/stone+wire restart

2 To test that Wire is working, type:

/usr/discreet/sw/tools/sw_framestore_dump

This command should show all Wire hosts discovered on your network.

Upgrading Visual Effects and Finishing Projects to the Current Version

Use the *copyProjects* command-line tool to upgrade your projects to the current version.

NOTE You do not need to perform this task if you are upgrading to a service pack of the same version or to a service pack of the same extension.

Using this tool instead of manually re-creating your projects is the recommended approach, as it reduces the time needed to migrate multiple projects, and prevents human error.

NOTE The *copyProjects* tool does not copy any media. It only re-creates projects and copies their settings and setups to the current version. You still have to transfer your media using the network library.

To copy projects using the *copyProjects* tool:

1 Open a terminal and log in as root.

2 Run the *copyProjects* script:

/usr/discreet/<product_home>**/bin/copyProjects**

NOTE The *copyProjects* tool requires no command-line arguments.

The tool detects all projects from previous versions, and returns the following message:

```
Projects will be copied to version 2012

Do you wish to continue?
```

If the version returned by the script is the newly installed version, answer Yes and continue this procedure. Otherwise, perform the steps in the next procedure to change the application version to which projects are copied.

3 Confirm each project you want to copy to the new version, or type "**a**" to copy all projects.

The tool creates a copy of each project for the current version. The string *_copy* is appended to the end of the new project name.

4 Use the network library to transfer media into the copied projects.

To change the application version projects are copied to:

1 Open a terminal and log in as root.

2 Type the following command:

/usr/discreet/<product_home>/bin/selectWiretapVersion.py

3 Select the version you want to copy projects to.

4 When you are finished, run the *copyProjects* script.

See the following section for instructions on transferring clip libraries from previous versions.

Transferring Clip Libraries to 2012 Projects

To access clip libraries from projects created with any Visual Effects and Finishing software between versions 2007 to 2011, you must transfer the clip libraries to a project created in version 2012.

You can install version 2012 on the system and use Wire to transfer the clip libraries to a version 2012 project. See your application help.

For a Linux system where you do not wish to install 2012, or for IRIX® systems, you can also use the following procedure to transfer media to 2012.

To transfer clip libraries to a 2012 project:

1 Use Backdraft® Conform 2007 or any 2008, 2009, 2010, or 2011 application to create a project and a clip library on the 2012 system. See your application help. The project and library is formatted for the software version it was created with.

2 Use the 2007, 2008, 2009, 2010, or 2011 application to move clips over the network into the previously created project on the 2012 system. You can use Background Wire to help free up the resources of the workstation running the older version of the application. See your application help.

3 Create a new project on the version 2012 system. If you have many projects, use the *copyProjects* script.

4 Use the 2012 application to open the library created in step 1 (the library is read-only).

5 Use Dual View to drag the clips into the 2012 project and library.

6 Use the older application version to remove the project created in step 1. 2012 applications cannot remove projects created with previous versions because they are read-only.

TIP If you have 2007, 2008, 2009, 2010 or 2011 installed on your 2012 system, you can access the 2007, 2008, 2009, 2010, or 2011 system remotely to transfer the media using Wire without interrupting a session on that station. See your application help.

Configuring Backburner Services

Background Wire transfers and proxy generation rely on Backburner. Backburner can also be used to manage background processing jobs sent to Burn® render nodes, and distributed transcoding jobs to import certain media formats, such as REDCODE™ RAW or OpenEXR.

Backburner Server and Backburner Manager are installed with the Autodesk Visual Effects and Finishing application, and are set to run automatically when the workstation starts. By default, the Backburner Server installed on the workstation communicates with the locally installed Backburner Manager.

Use the following procedure to configure Backburner services on your workstation.

To configure Backburner services:

1 Make sure Backburner Server is set to run automatically on your workstation. Type:

chkconfig –-list | grep -i backburner_server

The command should output the following information:

```
backburner_server 0:off 1:off 2:on3:on4:on5:on6:off
```

2 If backburner_server is not set to "on" for run levels 3, 4, and 5, type the following commands to enable it:

chkconfig backburner_server on

/etc/init.d/backburner_server restart

3 Decide if you want to use the locally installed Backburner Manager, or a centralized Manager installed on another system in your network. It is recommended to use a centralized Manager if several workstations on your network use the same farm of Burn render nodes.

Consult the *Autodesk Backburner Installation Guide* at www.autodesk.com/backburner-documentation for detailed guidelines.

4 If you do not want to use the local Backburner Manager installed on your workstation, type the following commands to disable it:

chkconfig backburner_manager off

/etc/init.d/backburner_manager stop

5 If you stopped the local Manager, open the */usr/discreet/backburner/cfg/manager.host* file on your workstation in a text editor, delete "localhost", and enter the hostname of the system where Backburner Manager is installed.

6 Make sure the following Backburner keywords in the */usr/discreet/*`<product_home>`*/cfg/init.cfg* file are configured properly.

Keyword	Setting
BackburnerManagerHostname `<hostname>`	Set `<hostname>` to the hostname or IP address of the workstation running Backburner Manager. This hostname or IP address must match the one in the */usr/discreet/backburner/cfg/manager.host* file. If it does not, background Wire transfers will fail.
BackburnerManagerPriority `<priority>`	Set `<priority>` to a value from 0 to 100. The default is 50.
BackburnerManagerGroup `<groupname>`	Set `<groupname>` to the name of a group of computers on a Burn® rendering network. For example, if the name of the group is "renderfarm1", you would set this keyword to BackburnerManager-Group renderfarm1.

Configuring Lustre Slave Rendering

Workflow for Configuring Slave Rendering

The Slave Renderer renders modified frames when the artist moves to the next shot on the timeline from the Master Station. With slave rendering, playback is enabled without compromising the interactivity of the Master Station during creative sessions.

Connect the Slave Renderer to a switch on your GigE or 10-GigE house network. You can configure it to access media through NFS mount points, or using the *BrowseD* service. Using *BrowseD* is the recommended approach, as it is much faster than NFS.

See the following table for a summary of the steps necessary to configure slave rendering.

Step:	Refer to:
1. Install Linux and the Discreet Kernel Utility (DKU) on the Slave Renderer.	Do I Need to Reinstall Linux? (page 33) and Installing the DKU (page 44).
2. Connect the Slave Renderer to a switch on your house network.	The Slave Renderer section in Setting Up Your Workstation Hardware, or in the *Hardware Setup Guide* for your workstation.
3. Configure the network address that the Slave Renderer uses to communicate to the workstation.	Configuring the Network Addresses (page 73).
4. Install the slave rendering software and license.	Installing the Slave Renderer Software and License (page 73).
5. If you are not using *BrowseD*, configure read/write access to storage attached to the Master Station from the Slave Renderer.	Configuring Read/Write Access to Storage from the Slave Renderer (page 74).
6. If you are not using *BrowseD*, mount the storage attached to the Master on the Slave Renderer.	Mounting the Storage on the Slave Renderer (page 76).
7. Set up the configuration file so that Lustre projects can use slave rendering.	Setting Up Lustre Projects to Use Slave Rendering (page 77).
8. Render shots as you work.	*Autodesk Lustre User Guide*.

Configuring the Slave Renderer Network Address

After installing Linux on the Slave Render, you must configure the IP of the ethernet port that connects the Slave Render to your house network.

NOTE The address you choose must not conflict with any of the other IP addresses on the network.

In a Lustre installation in which you have not altered any of the IP addresses used by default, you can use 193.1.1.2 as the Slave Renderer IP address.

To configure the network address:

1 Start the Slave Renderer and log in as *root*.

2 Assign an IP address to the ethernet port that connects the Slave Renderer to the network switch. The IP address must be in the same IP class as your network.

   TIP Typically, your Linux workstation's TCP/IP settings are configured in the file */etc/sysconfig/network-scripts/ifcfg-eth*`<port#>` , where `<port#>` is the port used to connect the Slave Renderer to the network, or directly to the Master Station. For example, if the connection is on port 0 of the onboard network card, the file you would edit would be */etc/sysconfig/network-scripts/ifcfg-eth0*.

Installing the Slave Renderer Software and License

Use the following procedures to install the Slave Renderer software and license.

To install the Slave Renderer software:

1 Start the Slave Renderer and log in as *root*.

2 If you are installing from a USB key, insert the key navigate to the Lustre install directory

3 If you are installing from a downloaded *tar* file, unpack the tar file by typing:

   tar zxvf `<file_name>`**.tar.gz**

The installation package is unpacked into a new directory. Navigate to the new directory.

4 Start the installation script by typing:

./INSTALL_LUSTRE_SLAVE

The Slave Renderer is installed.

To install the Slave Renderer license:

1 If you have not already done so, obtain a license for the Slave Renderer. You will need the *dlhostid* of the Slave Renderer. See Requesting License Codes (page 100).

2 Log in to the Slave Renderer as root.

3 Create the directory for the license file. Type:

mkdir -p /usr/local/flexlm/licenses/

4 Create a file called *DL_license.dat* using a text editor such as *nedit*, and save it to the new directory. Type:

nedit /usr/local/flexlm/licenses/DL_license.dat

NOTE The license directory and filename are case-sensitive and must be created and named exactly as specified.

5 Copy the license code you received for the Slave Renderer from Autodesk to the *DL_license.dat* file.

Configuring Read/Write Access to Storage from the Slave Renderer

The storage must be exported from the Master Station in order to allow read/write access. This, in turn, makes it possible for the Slave Renderer to remotely mount the storage system.

NOTE You do not need to perform the tasks in this section if you are using the *BrowseD* service with the Slave Renderer. See Using BrowseD with the Slave Renderer (page 97).

To export the storage attached to the Master Station:

1 Log in to the Master Station as *root*. The storage filesystem is mounted as */mnt/StorageMedia* (for HP Z800 stations), or */mnt/md0* (for older systems).

2 Open the */etc/exports* file in *vi* by typing:

vi /etc/exports

The contents of the *exports* file appears, listing the available network shares (filesystems mountable over the network). Next, add a new line for the mount point of the storage filesystem.

3 Scroll to the end of the *exports* file using the arrow keys.

4 Press Insert to enter insert mode.

5 Add a new line for the mount point. For example:

/mnt/StorageMedia *(rw,no_subtree_check,async)

or

/mnt/md0 *(rw,no_subtree_check,async)

This line creates a network share for the mount point for the storage filesystem. Other users on the network will access the storage devices by mounting its network share.

6 Press Esc to enter command mode.

7 Save and close the file by pressing Esc to enter command mode, then Shift + zz.

The file is saved and you are returned to the command prompt.

8 Activate the new network share for the storage device. Type:

exportfs -a

The list of network shares is refreshed from the *exports* file.

9 Confirm the storage devices are available to be mounted by typing:

exportfs

The network shares of the Lustre Frame server appears in a list similar to the following:

```
/mnt/StorageMedia <world>

...
```

Mounting the Storage on the Slave Renderer

After you configure read and write access on the storage filesystem, you create a mount point on the Slave Renderer and configure the Slave Renderer to automatically mount that filesystem.

NOTE You do not need to perform the tasks in this section if you are using the *BrowseD* service with the Slave Renderer. See Using BrowseD with the Slave Renderer (page 97).

To create a mount point on the Slave Renderer:

1 On the Slave Renderer, log in as *root*.

2 To go to the top level directory, in a terminal, type:

   cd /

3 Create a directory for the mount point. For example, type:

   mkdir /mnt/StorageMedia

4 Change the permissions on this directory to allow read/write access. For example, type:

   chmod 666 /mnt/StorageMedia

5 Mount the storage exported from the Master Station. For example, if the storage on the Master Station is mounted under */mnt/StorageMedia* type:

   mount <Master Station>:/mnt/StorageMedia/ /mnt/StorageMedia

To configure the Slave Renderer to automatically mount the storage filesystem:

1 In a terminal, type:

   vi /etc/fstab

2 Press Insert on the keyboard, and then add a line for the mount point you just created:

   <IP address>:/<exported filesystem> /<mount point> nfs rw,bg,hard,intr 0 0

   For example, type:

   172.16.60.226:/mnt/StorageMedia /mnt/StorageMedia nfs rw,bg,hard,intr 0 0

   NOTE Both examples are single lines.

3 Save and close the file by pressing `Esc` to enter command mode, then `Shift + zz`.

The file is saved and you are returned to the command prompt.

4 Restart the Slave Renderer machine.

When you restart your system, the storage filesystem mounts automatically.

Setting Up Lustre Projects to Use Slave Rendering

After you have configured the Master Station and the slave rendering machine to communicate, you must:

■ Configure the "HostName string=" keyword with the slave rendering machine IP address in the Slave Renderer section of the *init.config* file. See NO LABEL .

■ Enable slave rendering for the project.

Configuring Lustre Background Rendering

About Background Rendering

During background rendering, a shot on the timeline is rendered by a background rendering network. This is different from the Slave Renderer, which renders shots on a shot-by-shot basis as they are colour graded to enable improved playback performance.

Background rendering in Lustre is performed using Burn™ for Lustre, also known as the Lustre Background Renderer. This application is specific to Lustre and provides asynchronous background processing of Lustre render jobs. By off-loading rendering activities to remote Linux® servers, Lustre stations are freed up for interactive colour grading, while background rendering is sped up by splitting the task amongst multiple hosts.

Background Rendering Components

The components of the basic background rendering package include Lustre, a background management and monitoring application (such as Backburner™ Web Monitor, or the Backburner Monitor in Autodesk WiretapCentral™), and several render nodes running on Linux servers.The Lustre system and all background rendering nodes are connected over a dedicated background TCP/IP network. Render nodes can access media through NFS mount points, or by using the faster *BrowseD* service. Using *BrowseD* is the recommended approach. See About BrowseD (page 93).

The background rendering components are illustrated as follows.
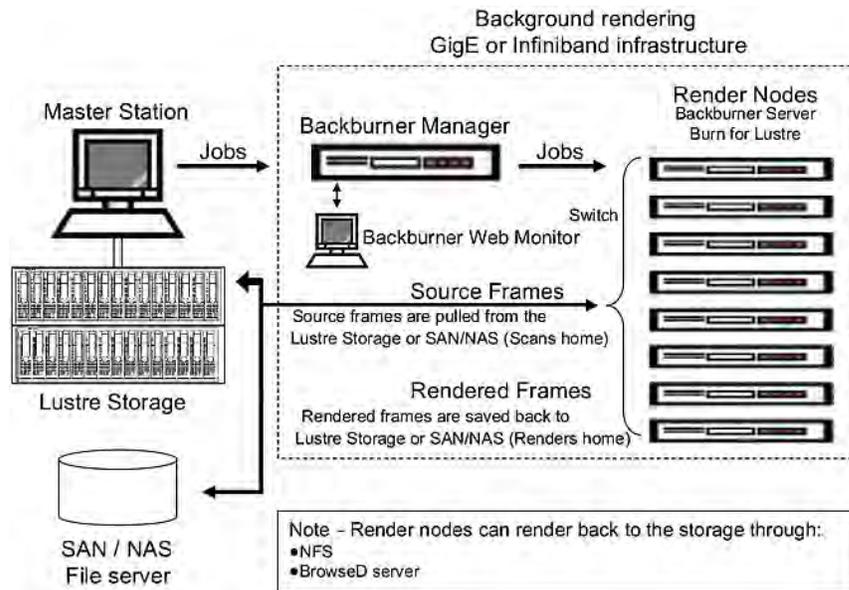
NOTE You can have up to eight render nodes on the background rendering network.

The other background rendering components are described as follows.

Lustre application	This is the client application. Running on a Linux workstation, Lustre rendering jobs are submitted for background rendering through the Render | Backburner menu.

Backburner Manager	This is the hub of the background rendering workgroup. Backburner Manager can reside on a Master Station or Lustre station, or can be installed on a dedicated machine in your network. When jobs are submitted from Lustre to Backburner Manager, Backburner Manager breaks each submitted job into tasks and distributes the tasks to the rendering servers on the network. To view the progress of the tasks, use Backburner Monitor.
Backburner Web Monitor	This is the user interface for the Backburner rendering network. It is automatically installed with Backburner Manager, and can be accessed through a Web browser from any workstation on the network. It allows you to view and control jobs currently being processed. You can stop, restart, reorder or remove jobs completely using the Monitor. You also use Backburner Web Monitor to identify any render nodes that are not working and check the overall status of the rendering network.
Backburner Server	This is a server that runs on render nodes, and is automatically installed with Burn for Lustre. Backburner Server accepts commands from Backburner Manager to start and stop rendering tasks. Backburner Server communicates through a plug-in to execute rendering tasks.

NOTE Consult the latest *Autodesk Backburner Installation Guide* and *Autodesk Backburner User Guide* for detailed information on Backburner. Use the guides to obtain Backburner information that does not relate directly to Lustre background rendering, such as details on setting up Web monitoring and troubleshooting tips. You can find these guides on the *Backburner documentation webpage*.

Burn for Lustre	This is the Linux rendering engine that renders one or more frames from Lustre render jobs.

NOTE Burn for Lustre is a different software than Burn for Visual Effects and Finishing software. While they can co-exist on the same node, the installation and configuration procedures are different. For information about how to set up Burn for Visual Effects and Finishing software, consult the *Autodesk Burn Installation and User Guide*.

Shared storage mount point	This is the mount point on each Linux server that allows Burn for Lustre to transfer rendered frames/files to the Lustre storage system.

NOTE The storage does not have to be mounted on the render nodes if you are using *BrowseD* for background rendering. See

Workflow for Setting Up Background Rendering

The following table outlines the general workflow for installing and configuring background rendering.

Step:	Refer to:
1. If you are not using *BrowseD*, enable sharing on the system storing the Lustre project and render files.	Sharing the Storage for Read/Write Access from Background Render Nodes (page 80).
2. Install and configure Backburner Manager and Backburner Web Monitor.	Installing Backburner Manager and Backburner Web Monitor (page 81).
3. Configure the Lustre workstation to connect to Backburner Manager.	Configuring Lustre to Detect Backburner Manager (page 91).
4. Set up the background render nodes.	Setting Up Render Nodes (page 83).
5. Specify the mount point of the background render nodes in the Lustre application.	Specifying the Background Rendering Path in Lustre (page 92).

Sharing the Storage for Read/Write Access from Background Render Nodes

In order to allow read and write access, the storage must be exported from the system to which it is connected. This makes it possible for the background rendering components on the network to remotely mount the storage system.

NOTE Skip this section if you are using the *BrowseD* service.

To make the storage system visible over the network:

1   Log in to the Lustre system (or the central storage system) as *root*. The storage filesystem is mounted as */mnt/StorageMedia*(on HP Z800 workstations) or */mnt/md0*(on older workstations).

2   Open the */etc/exports* file in a text editor, such as *vi*.

   The contents of the *exports* file appears, listing the available network shares (filesystems mountable over the network).

3   At the end of the file, add a new line for the mount point of the storage filesystem:

   /mnt/StorageMedia *(rw,no_subtree_check,async)

   or

   /mnt/md0 *(rw,no_subtree_check,async)

   This line creates a network share for the mount point of the storage filesystem. Other users on the network will access the storage devices by mounting its network share.

4   Save and close the file.

5   Activate the new network share for the storage device. Type:

   exportfs -a

   The list of network shares is refreshed from the *exports* file.

6   Verify the storage devices are available to be mounted by typing:

   exportfs

   Make sure the storage filesystem mount points are listed in the command output.

Installing Backburner Manager and Backburner Web Monitor

On Lustre stations, Backburner components are automatically installed with the Lustre software. The installer asks you if you want to run Backburner components on the station.

To avoid performance degradation, it is recommended to avoid running Backburner components on your Lustre stations, and to relocate Backburner Manager on a dedicated Linux machine in your network. Render jobs handled by Backburner can be viewed and managed in Backburner Web Monitor from any workstation on the network.

To install Backburner Manager and Backburner Web Monitor on a Linux system:

1   Open a terminal and log in as root.

2   Access the Autodesk Lustre software installation package (from a downloaded *tar* file or on USB key).

3   Install Backburner by typing:

   rpm -Uvh backburner*

   The Backburner components (Server, Manager and Web Monitor) are installed on the system.

4   If the system on which you are installing Backburner runs Linux in graphical mode, you are asked if you want to automatically run Backburner Manager and Backburner Server on the machine. Answer Yes to the question about the Manager, and No to the question about the Server.

5   Run the Backburner Web Monitor configuration script:

   /usr/discreet/backburner/backburnerConfigWeb

   The script automatically configures Backburner Web Monitor.

To verify that Backburner components are running properly:

1   On the system on which you installed Backburner Manager, as root, type:

   ps -ef | grep -i backburner

   The output of the command should include a line containing the path to the Backburner Manager application, similar to the following:

   root   14989   10 Nov29 ?   00:02:36 /usr/discreet/backburner/backburnerManager

   NOTE If Backburner Manager is not running, the Backburner log file */usr/discreet/backburner/Network/backburner.log* may be a useful source of information for troubleshooting.

2   To test that Backburner Web Monitor is installed and configured properly, open a Web browser on the Manager machine and point it to *http://loc- alhost/backburner*.

   Backburner Web Monitor opens. If you are asked for a user name and password, use *backburner* for both.

Refer to the *Autodesk Backburner Installation Guide* for detailed information on configuring Backburner.

Setting Up Render Nodes

You can set up as many as eight render nodes for background rendering with Lustre. On each system intended for background rendering, you must do the following.

Step:	Refer to:
1. Verify that the render nodes meet the minimum system requirements.	Render Node Hardware and Software Requirements (page 83).
2. Install the appropriate Red Hat operating system.	Installing Linux on Render Nodes (page 84).
3. If you are not using *BrowseD*, configure each node to mount the storage that contains the project render files.	Mounting the Storage on the Render Nodes (page 85).
4. Install Burn for Lustre.	Installing Burn for Lustre on Render Nodes (page 87).
5. Connect the render nodes to Backburner Manager.	Configuring Backburner Server to Detect Backburner Manager (page 90).
6. Start the Backburner Server on each render node.	Starting Backburner Server (page 91).

Render Node Hardware and Software Requirements

Render nodes purchased from Autodesk come with all the necessary hardware and software components preinstalled.

To use Burn for Lustre on nodes that were not purchased from Autodesk, the nodes must meet the minimum hardware and OS requirements listed in the following table.

Processor:	1 or 2 Xeon® Intel® Processor DP
Memory:	2 GB

Hard drive:	20 GB
Ethernet:	100/1000 Mbps
OS:	Red Hat Enterprise Linux 5.3 with Workstation option, customized with the Autodesk kickstart file

Installing Linux on Render Nodes

Autodesk Burn for Lustre runs under the custom 64-bit Autodesk distribution of Red Hat Enterprise Linux.

Render nodes purchased from Autodesk ship with the correct Autodesk distribution of Red Hat Enterprise Linux on DVD.

If you did not purchase your render node from Autodesk, you must get your own 64-bit distribution of Red Hat Enterprise Linux Desktop 5.3 with Workstation option, and customize it using the Autodesk kickstart file.

The kickstart is used to automatically install the packages required for Burn, some of which are not installed as part of a general installation.

This file is availabe in the Lustre installation directory. You must copy it to the DVD of your Linux distribution.

TIP The kickstart file can be used to automate the Linux installation process for multiple render nodes.

To copy the Autodesk kickstart file to the DVD of your Linux distribution:

1 On a computer running Linux and with a DVD burner, log in as root.

2 Insert the DVD of your Linux distribution into the drive. You do not need to mount it at this time.

3 In a terminal, extract an ISO image of the disc by typing:

dd if=/dev/`<optical_disc_device>`**of=/**`<destination path for the extracted ISO image>`

For example:

dd if=/dev/cdrom of=/tmp/RHEL5.3.iso

Depending on the speed of your disc drive, this command may take several minutes to complete.

4 Eject the disc.

5 Access the *dist* subdirectory of the Lustre installation package.

The directory contains a kickstart file, *ks_centos5_rh5.cfg*, as well as a script that adds the kickstart file to an ISO image.

6 Run the *build_kickstart_cd* script to add the kickstart file to the ISO image of your Linux distribution DVD:

./build_kickstart_cd ks_centos5_rh5.cfg`<original ISO image name> <new ISO image name>`

For example, if the ISO image you created is called */tmp/RHEL5.3.iso* and you want the new ISO image to be called */tmp/RHEL5.3_KS.iso*, type:

./build_kickstart_cd ks_centos5_rh5.cfg /tmp/RHEL5.3.iso /tmp/RHEL5.3_KS.iso

7 Once the new ISO image of the Linux distribution DVD is created, burn it to a blank disc using a tool such as **cdrecord**.

NOTE Type **man cdrecord** for information about this utility.

The new disc that you burn contains the Autodesk kickstart file and replaces the DVD in the Linux distribution.

You are now ready to install Linux on the render node. See Linux Installation Workflow (page 34).

Mounting the Storage on the Render Nodes

After Red Hat Linux is installed, you must mount the media storage on each render node using an NFS mount point.

NOTE The *BrowseD* service can also be used to allow background render nodes to access the workstation's storage. *BrowseD* allows for fast access. If you are using *BrowseD* for background rendering, skip this section. See About BrowseD (page 93).

Mounting the storage involves identifying the IP address of the system that manages the storage and setting a mount point on each render node that points to the correct path on the storage. Also, for this mount point to be available when you restart the system, you must add an entry in the *fstab* file.

If the system that stores the render files is different from the system that stores the Project Home directories, you need two (or more) mount points. For example, if, in the Project Setup menu, the Project Home path is

/mnt/md0/SCENES and the Renders Full Home path is */sanserver:/RENDERS*, located on a central storage system, you must define a mount point for each path.

To mount the storage on a render node:

1 Determine the IP address of the computer that manages the storage.

   If you are using direct attached storage only, this is the IP address of the Lustre Master Station. You must also determine the IP address of your centralized file server if you are also using a server area network (SAN) or network attached storage (NAS).

2 On the render node, log in as *root*.

3 Go to the root directory. In a terminal, type:

   cd /

4 Create a directory for the mount point. For example, type:

   mkdir /mnt/md0

   NOTE The directory name must be the same for the mount point on each node. Also, the directory should not be created in the */root* directory, but in the */mnt* directory.

5 Change the permissions on this directory to allow read/write access. For example, type:

   chmod 666 /mnt/md0

6 Set up the mount point to access the Lustre system storage or the central system storage. You must use the mount point as defined in the *Project_home* setting in the Lustre Project Setup menu. For example, if the Project Home is */mnt/md0/SCENES*, and the Linux mount point directory is named */mnt/md0*, the mount command will be:

   mount<hostname>**:/mnt/md0/SCENES /mnt/md0**

To add an entry in the *fstab* file:

1 Open the file */etc/fstab* in a text editor and then add a line for your new mount point:

   <IP address>:/<exported filesystem> /<mount point> nfs rw,bg,hard,intr,nfsvers=3 0 0

   For example, type:

   172.16.60.226:/mnt/md0 /mnt/md0 nfs rw,bg,hard,intr,nfsvers=3 0 0

NOTE The example is a single line.

2 Save and close the file.

The file is saved and you are returned to the command prompt.

3 Restart the render node.

When you restart your system, this remote location will mount automatically.

Installing and Configuring Burn for Lustre on Render Nodes

Install and configure Burn for Lustre on each render node. To install and configure Burn for Lustre, you must:

■ Install Burn for Lustre.

■ Add the IP address of the machine where Backburner Manager is installed to the *manager.host* file on each render node.

■ Start the Backburner Server on each render node.

■ License Burn for Lustre.

Installing Burn for Lustre on Render Nodes

Perform the following procedure to install the Burn for Lustre software on each node.

NOTE When you install Burn for Lustre, the necessary Backburner components are also installed on the render node.

To install Burn for Lustre on a render node:

1 Open a terminal to the render node, and log in as *root*.

2 If you are installing from a USB key, insert the key navigate to the Lustre install directory.

3 If you are installing from a downloaded *tar* file, unpack the tar file by typing:

tar zxvf <file_name>**.tar.gz**

The installation package is unpacked into a new directory. Navigate to the new directory.

4 Start the Burn installation script by typing:

./INSTALL_LUSTRE_BURN

The Burn for Lustre package is installed. For instructions on obtaining and installing a Burn license, see Licensing Burn for Lustre (page 88).

5 Repeat this procedure on all render nodes.

Licensing Burn for Lustre

You need a license for your render nodes. Burn uses a floating license scheme, which means that licenses are centralized on a license server.

The license server automatically provides a license to all registered machines.

The license server can be any of the render nodes on the rendering network.

Perform the following tasks to obtain a license code for background rendering, install the license server, and configure the license server to distribute licenses to burn nodes.

To license a Burn for Lustre network:

1 Open a terminal on the license server and log in as root.

2 Run the *dlhostid* utility to obtain the unique host ID of the machine. Type:

dlhostid

A message appears that includes a line indicating the *dlhostid* of the machine. For example:

```
The Discreet host ID of this machine is
"DLHOST01=25231AEF83AD9D5E9B2FA270DF4F20B1"
```

3 Write down the *dlhostid* (including the "DLHOST01=" part).

4 Request license codes for background rendering from the Autodesk Media and Entertainment Licensing Department. See Requesting License Codes (page 100).

5 If you are installing from a USB key, insert the key navigate to the Lustre install directory.

6 If you are installing from a downloaded *tar* file, unpack the tar file by typing:

tar zxvf <file_name>**.tar.gz**

The installation package is unpacked into a new directory. Navigate to the new directory.

7 Run the license server installation script by typing:

./INSTALL_LICSERV

The license server is installed.

8 Once you have received the license codes, go to the */usr/discreet/licserv/licenses/* directory, and open the *DL_license.dat* file in a text editor.

   NOTE If the file does not exist, create it by typing:

   touch /usr/discreet/licserv/licenses/DL_license.dat

9 In this file, enter all the information submitted to you by the Licensing Department upon your registration.

10 Save and close the file.

11 Start the license server by running the following command:

   /etc/init.d/license_server start

   NOTE The license server starts/stops automatically when the machine is booted/shut down. You can stop and restart the server manually by running one of the following commands:

   /etc/init.d/license_server stop

   /etc/init.d/license_server start

Now that the license server is set up, configure each node to contact the server and retrieve its license. Perform the following procedure on each node.

To enable render nodes to contact the license server:

1 Log in to each render node as root.

2 Navigate to the license directory. Type:

   cd /usr/local/flexlm/licenses/

3 Open the *DL_license.dat* file in a text editor.

4 In this file, copy the first three lines of the information submitted to you by the Autodesk Licensing Department upon your registration. For example:

   SERVER burn-01
   DLHOST01=25231AEF83AD9D5E9B2FA270DF4F20B1
   VENDOR lustre
   USE_SERVER

5 Save and close the file.

NOTE Each render node must be able to contact the license server. Use *ping* from each node to the license server to confirm that the machines can communicate.

Configuring Backburner Server to Detect Backburner Manager

Backburner Server needs to be able to detect the location of Backburner Manager in order to provide status information concerning the render jobs. Set the IP address of the Backburner Manager machine in the */usr/discreet/backburner/cfg/manager.host* configuration file on each render node.

NOTE You can also use the hostname of the Backburner Manager machine, if it has been properly defined in the */etc/hosts* file.

To configure Backburner Server to detect Backburner Manager:

1   On the Backburner Manager system, open a terminal and log in as root.

2   Determine which IP address the Backburner Manager workstation uses to connect to the network. Type:

   ifconfig

   Information about all the network adapters is displayed. The second line of the output for each adapter contains the IP address (listed as "inet addr"). For example:

   ```
   inet addr:172.16.129.152 Bcast:172.16.135.255
   Mask:255.255.248.0
   ```

   Write down the IP address of the network adapter. If the machine has several network adapters, make sure you write down the address of the one connected to your render network.

3   On each render node, open a terminal and log in as *root*.

4   Type:

   cd /usr/discreet/backburner/cfg

5   Use a text editor to edit the *manager.host* file.

6   Enter the IP address of the Backburner Manager machine. For example:

   10.10.10.1

7   Save and close the file.

   The file is saved and you are returned to the command prompt.

Starting Backburner Server

You must start the Backburner Server daemon on each Linux system for the first time. Once it is started the first time, the daemon will start automatically.

To start Backburner Server:

1   Log in to your Linux system as *root*.

2   In a terminal, type:

    service backburner_server start

3   You can verify that Backburner Server is properly configured by checking the log file *usr/discreet/backburner/Network/backburnerServer.log*. Type:

    tail -f /usr/discreet/backburner/Network/backburnerServer.log

    The file should contain lines similar to the following:

    ```
    <INF>: Loading plugin: Lustre renderer plugin
    ```

4   To stop viewing the file, press Ctrl+C.

5   Repeat these steps to start Backburner Server on each node in the background rendering network.

    If you cannot start the Backburner Server, contact Autodesk Customer Support.

NOTE If you need to stop or restart Backburner Server, use one of these commands:

/etc/init.d/backburner_server stop

/etc/init.d/backburner_server restart

Configuring Lustre to Detect Backburner Manager

For Backburner Manager to receive render jobs, you must set up the Lustre application to connect to the system on which Backburner Manager is running.

To configure new projects to use background rendering, you must set the IP address of the Backburner Manager workstation in the *init.config* file.

To configure the Lustre application to detect Backburner Manager:

1   Locate the Backburner keyword.

2 In the `HostName` line, set the `string` parameter to the hostname or IP address of the system where Backburner Manager is installed. For example:

```
<HostName string="172.19.23.161" />
```

NOTE Do not forget the quotes.

3 Save and close the configuration file.

Specifying the Background Rendering Path in Lustre

The last step in setting up background rendering is to specify the correct mount points in the settings of each project that uses background rendering.

To add the storage path in Lustre:

1 Start Lustre.

2 In the Setup menu, click Settings.

3 Select your project in the Project drop-down list, and click Edit.
The Project Settings menu appears.

4 Click Network Rendering, then click Backburner in the Configuration section.

5 Type the location of the Project Home, Scans Full Home, Scans Half Home, Renders Full Home and Renders Half Home, as seen from the Linux render nodes. You only need to enter those locations that are defined for the project in the local project setup, located in the Setup | Project menu.

The path required in these fields is comprised of the directory defined for the mount point and the actual storage folder. For example, if the Project Home on the Lustre workstation is *E:\SCENES/mnt/md0/SCENES* in the Project Setup menu, and the mount point for the Lustre workstation storage on the Linux render nodes is */mnt/Lustre_storage*, the Project Home path to enter in this menu is:

/mnt/Lustre_storage/SCENES

6 Press `Enter`.

7 To verify that you have entered the proper path for each project, look at the project configuration file in */usr/autodesk/lustre*`<application_version>`*/project/*`<project`

name>*/project.config*. The value in the `burn_project_home` line is sent to the render node. This value must match a mount point on the background render nodes.

For help setting up and managing projects, refer to the Lustre Help.

Configuring Lustre BrowseD

About BrowseD

BrowseD is a Lustre file server with its own high level protocols that optimize network transfers. BrowseD provides Lustre workstations, Slave Renderers, and background render nodes with high-speed access to centralized storage, especially if your network uses InfiniBand®, without mounting the storage on the render nodes.

Consider using BrowseD if your facility uses a storage area network (SAN), network attached storage (NAS), or a network file server. In these configurations, you install and configure BrowseD on the computer that is connected to the centralized storage and configure all workstations to use the BrowseD server to access Lustre project files.

The recommended Lustre configuration in a digital film or high-resolution workflow is to store the full-resolution images on a SAN, NAS, or file server, and the proxies locally. For information on configuring your projects, see the *Autodesk Lustre User Guide*.

NOTE You can run BrowseD on the Lustre Master Station to provide render nodes or Slave Renderers high-speed access to local storage for background rendering. However, this is not recommended. The BrowseD process requires extra bandwidth from the storage and adds extra load on the host system's CPU. Real-time functionality cannot be guaranteed with this setup—2K playback speed and video input and output will not function correctly when BrowseD is running on the Lustre workstation, and is serving client requests.

Workflow for Configuring BrowseD

See the following table for a summary of the steps necessary to configure BrowseD.

NOTE If you are working with StandardFS DPX or Cineon footage coming from the local Wiretap Server, you must mount your storage on the render nodes.

Step:	Refer to:
1. Start the BrowseD server.	Starting and Stopping the BrowseD Service (page 94).
2. On all machines (like the Lustre Master Station, the Slave Renderer, and background render nodes), configure the *init.config* file to connect to the BrowseD server.	Configuring Workstations to Use the BrowseD Server (page 95).
3. Make the remote storage accessible from the file browser on the Lustre workstations.	Making BrowseD Directories Accessible from the CEV_ProdName_CEV Browser (page 96).
4. Configure background rendering to use the BrowseD server.	Using BrowseD for Rendering with Burn for Lustre (page 97).
5. Configure the Slave Renderer to use the BrowseD server.	Using BrowseD with the Slave Renderer (page 97).

Starting and Stopping the BrowseD Service

The BrowseD service starts automatically after you install it. You can manually start the service using the command line interface. The service starts automatically whenever the computer running BrowseD restarts.

To start or stop BrowseD:

1 In a terminal, type one of the following.

Command	Description
/etc/init.d/browsed_<version>**stop**	Stops BrowseD.
/etc/init.d/browsed_<version>**start**	Starts BrowseD.

Command	Description
/etc/init.d/browsed_<version>**restart**	Restarts BrowseD if it is not currently running.
/etc/init.d/browsed_<version>**condrestart**	Restarts BrowseD if it is already running.

Configuring Workstations to Use the BrowseD Server

You must configure the *init.config* file for all machines that will use the BrowseD server to access centralized storage, such as Lustre workstations, slave renderers, or Burn render nodes.

Use a text editor to edit the *init.config* file, located at */usr/autodesk/<version>*.

Use the following table as a reference when configuring the *init.config* file.

Keyword	Required values
Username	Enter the administrative user on the BrowseD server, for example, *root* on Linux.
Password	Enter the password for the administrative user as defined above.
Port	All computers on the BrowseD network must use the same port to communicate. Set to 1055, the default. If configuring a render node or a workstation running on a GigE network, set this as 1044. If configuring a render node or a workstation running on a high-speed InfiniBand network, set this as 1055.
BrowsedServer	Defines the IP address or DNS host name for a specific BrowseD Server.
UseInfi	Enable this option if the networking procotol to use with BrowseD is InfiniBand.

Making BrowseD Directories Accessible from the Lustre Browser

When using centralized storage, you must make the directories on the BrowseD server accessible to all Lustre workstations through the browser. You do this by configuring the `BrowsedServerGroup` keyword in the *init.config* file.

The mapped directory will appear in the Lustre browser.
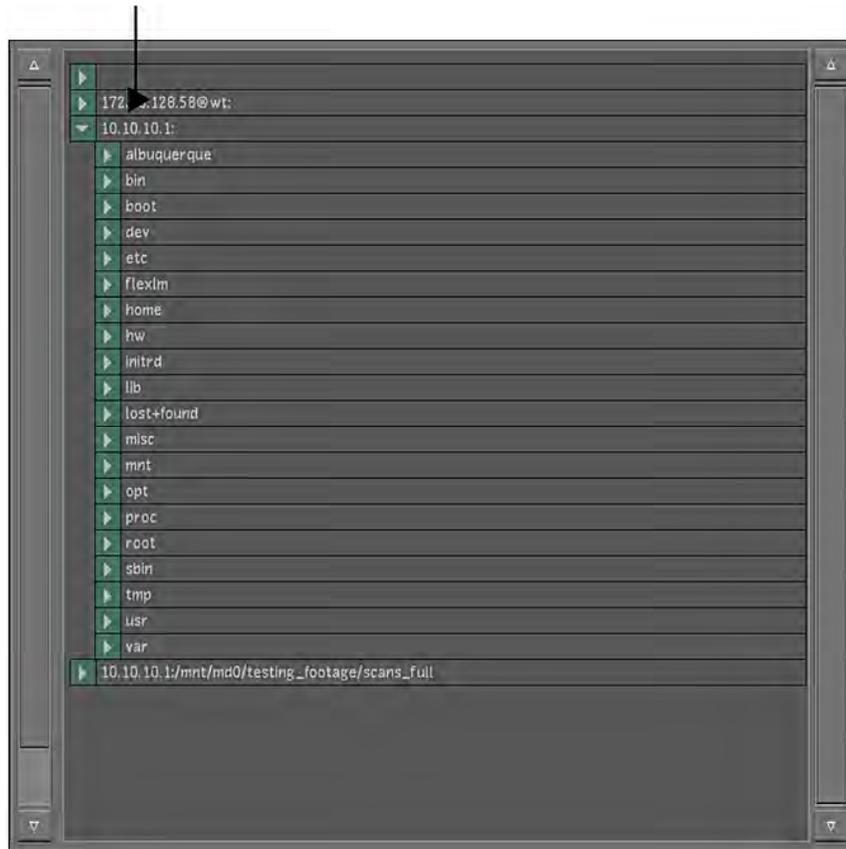
Using BrowseD for Rendering with Burn for Lustre

You must configure the Lustre Render Backburner paths with the IP address of the BrowseD server to use BrowseD to render with Burn for Lustre.

To use BrowseD for rendering with Burn for Lustre:

1   In Lustre, open the Setup menu, and select Settings.

2   Select your project from the Project list, and click Edit.

3   Click Project, and set the local project paths. For information on configuring your projects, see the *Autodesk Lustre User Guide*.

4   Click Network Rendering, and then click the Backburner tab.

5   Enter the IP address of the BrowseD server to each path, or click the ellipsis button to browse to the BrowseD path you want to use.

    For example, if your Project_Home is set to the *bernice*directory on the mount point /mnt/StorageMedia of a station that has an IP address of *172.17.20.146*, the Project_Home path in the Render > Backburner menu should read *172.17.20.146: /mnt/StorageMedia/project/bernice*.

Using BrowseD with the Slave Renderer

You must configure the Lustre Slave Render paths with the IP address of the BrowseD server in order to use BrowseD with the Slave Renderer.

To use BrowseD with the Slave Renderer:

1   In Lustre, open the Setup menu, and select Settings.

2   Select your project from the Project list, and click Edit.

3   Click Project, and set the local project paths. For information on configuring your projects, see the *Autodesk Lustre User Guide*.

4   Click Network Rendering, and then click the Slave Render tab.

5   Enter the IP address of the BrowseD server to each path, or click the ellipsis button to browse to the BrowseD path you want to use.

    For example, if your Project_Home is set to the *bernice*directory on the mount point */mnt/StorageMedia* of a station that has an IP address of

172.17.20.146, the Project_Home path in the Render > Backburner menu should read *172.17.20.146:/mnt/StorageMedia/project/bernice*.

Licensing and Starting

5

Node-Locked Licensing

About Node-Locked Licenses

Node-locked licenses are locked to one computer.

■ All customers receive a temporary node-locked license when they purchase their software.

■ Subscription customers receive a temporary node-locked license when they receive their extension software.

All customers must request permanent licenses:

■ Subscription customers have the choice of requesting permanent node-locked or network licenses.

■ Customers not on subscription are entitled to only node-locked licenses.

Node-Locked Licensing Workflow

If you have purchased subscription and are upgrading to a new release, your new license is sent to you by the Autodesk Media and Entertainment department. After receiving your license, perform the following tasks:

1 Install the new license. See Entering License Codes (page 102).

2 Start the application and accept the license agreement.

If you are installing the application for the first time, use the following steps to obtain and install license codes.

To obtain and install license codes for the application:

1 Request temporary license codes. See Requesting License Codes (page 100).

2 Install the temporary license codes. See Entering License Codes (page 102).

3 Start the application and accept the license agreement. See Starting Lustre for the First Time (page 116).

4 When you receive permanent license codes, install those codes. See Entering License Codes (page 102).

Requesting License Codes

Request licensing codes from the Autodesk M&E Edge support portal: *https://edge.autodesk.com/LicenseAssignment*.

If you do not have an account on the support portal, email *me.support@autodesk.com* to make a request.

NOTE For emergencies, you can acquire an immediate temporary emergency license using the emergency license generator at *http://melicensing.autodesk.com/templicensing/*. A 4-day license code is e-mailed to the address you provide.

To request license codes:

1 Start the software you want to license.

   NOTE You can also start the Wizard manually, from the Autodesk section in the KDE system menu.

2 Double-click the application icon to start the application.

   After a few moments, the License Wizard appears.
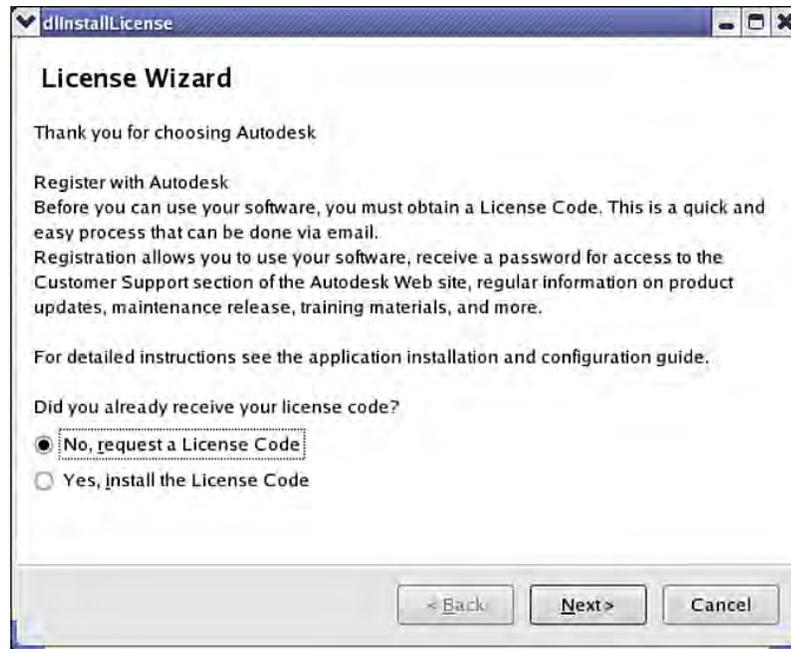
TIP You can also start the Wizard manually, from the Autodesk section in the KDE system menu.

3 Select "No, request a License Code", and click Next.

4 Select Node-Locked License and click Next.
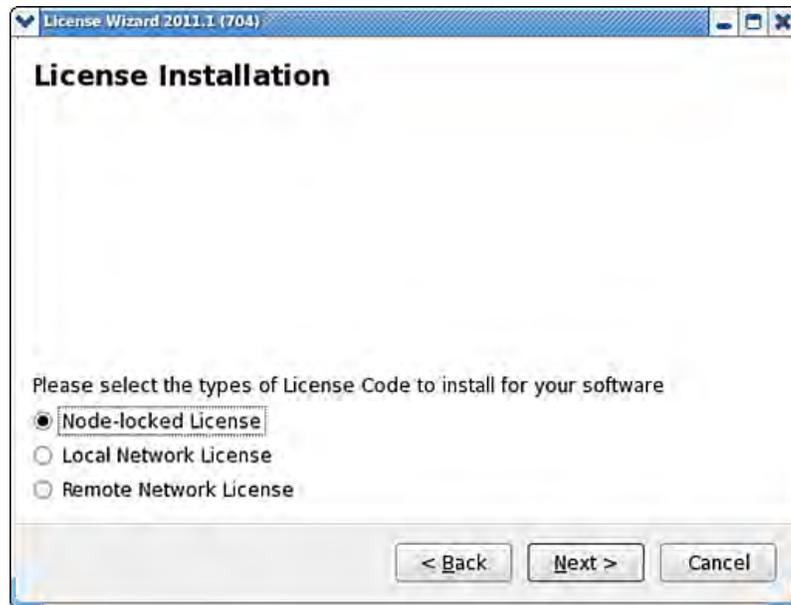
5 Enter your contact information in the license code request form. The host ID and system serial number of your workstation are automatically filled in.

6 Click Next.

7 Log in to support portal: *https://edge.autodesk.com/LicenseAssignment*.

8 Follow the links to get the a node-locked license.

Your license codes are emailed to you.

Use the License Wizard to license your application. The Wizard starts automatically when you launch an unlicensed application.

Entering License Codes

When you receive your license code from Autodesk, run the License Wizard again to enter the license codes for your software.

To enter software license codes:

1 Double-click the application icon on the Linux Desktop to start the application.

After a few moments, the License Wizard appears.
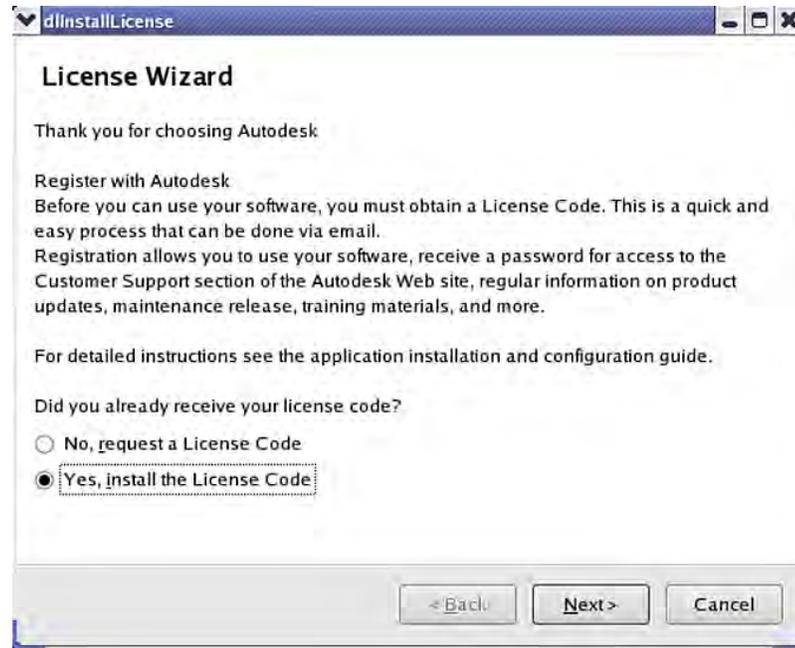
2 Select "Yes, install the License Code", and click Next.

3 Select Node-Locked License and click Next.
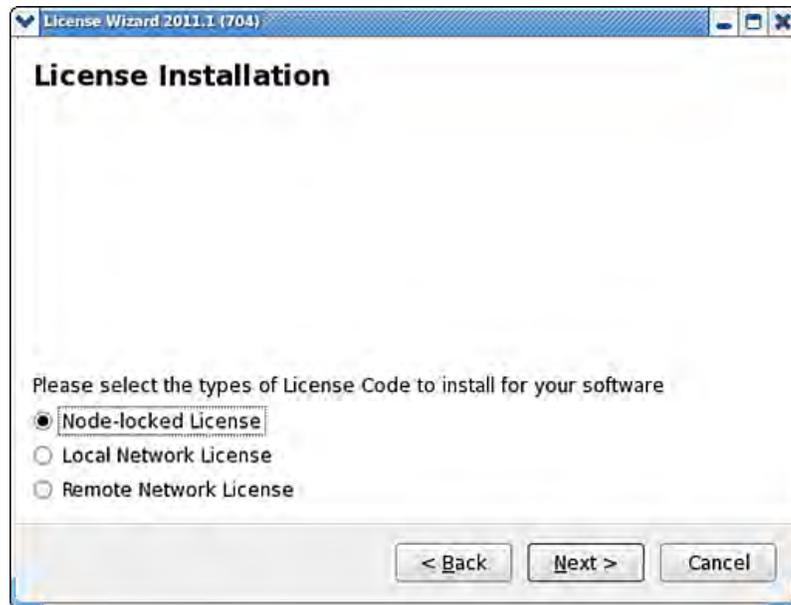
4 Paste the license codes you received from Autodesk in the next window.

> **NOTE** Be very careful to paste the license codes exactly as you have received them. Any incorrect or missing characters will prevent your application from recognizing the license.

5 Click Install to install the license code.

Your application is now licensed.

> **NOTE** License codes are unique for each workstation, and are tied to the system hardware. If you replace major system components in your system, you must request and install a new license code for your software.

6 Click Finish to close the License Wizard.

Your application is now ready to use.

NOTE To add additional license codes you receive from the Licensing Department (such as the DI license), run the License Wizard from the Autodesk section in the KDE system menu.

Network Licensing

About Network Licenses

With a network license, you can deploy your Creative Finishing software on any supported workstation in your facility. A license server manages the number and types of licenses you have available.

Network licenses are available only to customers with subscription to Inferno, Flame, Flame Premium, Smoke Advanced, and Lustre Master Station.

NOTE Network licenses are also available for Autodesk Flare, Autodesk Burn, and Autodesk Burn for Lustre.

All other customers must use node-locked licenses.

Network License Models

You must decide which Linux license server model to use for managing your Autodesk product licenses on a server. Autodesk supports the following network license server models:

- Single license server model
- Redundant license server model

Single License Server Model

In the single license server model, the license server daemon is installed on a single server, so license management and activity is restricted to this server. A single license file represents the total number of licenses available on the server. This single license server can reside on a remote workstation or on the same workstation as your Creative Finishing application.

Advantages of the Single License Server Model

- Because all license management takes place on a single server, you have just one point of administration and one point of failure.
- Of the two license server models, this configuration requires the least amount of maintenance.

Disadvantage of the Single License Server Model

- If the single license server fails, the Autodesk product cannot run until the server is back online.

Redundant License Server Model

In the redundant license server model, you use three Linux servers to authenticate a single license file. One server acts as the master, while the other two provide backup if the master server fails. With this configuration, licenses continue to be monitored and issued as long as at least two servers are still functional.

The license file on all three servers is the same. You must install the license server daemon on each server. In the redundant license server model, all servers must reside on the same subnet and have consistent network communications.

Advantage of the Redundant License Server Model

- If one of the three servers fails, all licenses that are managed in the server pool are still available.

Disadvantages of the Redundant License Server Model

- If more than one server fails, no licenses are available.

- All three servers must reside on the same subnet and have reliable network communications. The redundant server pool does not provide network fault tolerance.

- If you reboot one of the license servers or restart one license server daemon you must also reboot/restart all 3 license servers at the same time so the license servers are synchronized.

- If one of the three servers is replaced, you must request a new license key from Autodesk Creative Finishing Support

Workflow for Configuring a Local or a Single Remote Network License Server

1 Select a license server.

   You can install a license server on your Creative Finishing workstation or any other machine on your network.

NOTE To see if you already have a license server running on a machine, open a Terminal, login as root, and type: `chkconfig --list | grep lic`

2 Install the license server software. See Installing the License Server Software (page 108).

3 Get the *dlhostid* of your license server. See Getting the Unique Host ID of the License Server (page 109).

4 Get your license codes using the Autodesk M&E Creative Finishing Edge support portal: *https://edge.autodesk.com/LicenseAssignment*. See Requesting License Codes (page 109).

5 Start your Creative Finishing software and paste the license file you received from Autodesk into the license wizard. See Licensing the Workstation and Local License Server (page 110).

6 If you are using a remote license server, copy and paste the license you received in the */usr/discreet/licserv/licenses/DL_license.dat* file on your license server.

See Creating the License File for the License Server (page 111).

7 Restart the license server. See Starting the License Server (page 114).

8 Optional: To avoid conflicts with other applications or license servers in your facility, you may need to change the default port setting used by the Creative Finishing Software license server. See Changing the Default Port Used by the License Server (page 113).

Workflow for Configuring Redundant License Servers

1 Select three machines to function as license servers.

NOTE All three license servers must be set to the same date and time.

2 Install the license server software on all three machines. See Installing the License Server Software (page 108).

3 Get the *dlhostid* of each license server. See Getting the Unique Host ID of the License Server (page 109).

4 Get your license codes using the Autodesk M&E Creative Finishing Edge support portal: *https://edge.autodesk.com/LicenseAssignment*. See Requesting License Codes (page 109).

5 Create the workstation license
/usr/local/flexlm/licenses/DL_licenseNetwork.dat. Copy and paste the
SERVER, VENDOR, and USE_SERVER lines from the license you received
from Autodesk. See Creating the License File for the Workstation to Use
Redundant License Servers (page 112).

6 Create the license file */usr/discreet/licserv/licenses/DL_license.dat* on each
license server. Copy and paste the entire license you received from
Autodesk into the each license file.

See Creating the License File for the License Server (page 111).

7 Restart all of the license servers. See Starting the License Server (page
114).

8 Optional: To avoid conflicts with other applications or license servers
in your facility, you may need to change the default port setting used
by the Creative Finishing Software license server. See Changing the De-
fault Port Used by the License Server (page 113).

Installing the License Server Software

The license server is a Linux daemon that provides concurrent licenses to
Creative Finishing Softwareworkstations on the network, as needed.

You can install the license server on a Creative Finishing Softwareworkstation
in your network, or on any 64-bit system running Red Hat Enterprise Linux
Desktop 5.3 with Workstation Option (64-bit), Red Hat Enterprise Linux WS
4, Update 3 (64- bit), CentOS 5.3, or CentOS 4.3.

NOTE For a redundant network license server configuration, you must install the
license server software on all three workstations selected as license servers.

To install the license server:

1 Log in as root to a system designated as a license server.

2 Access your Creative Finishing Software software installation directory.

3 Launch the license server installation script by typing:

./INSTALL_LICSERV

The license server is installed.

Getting the Unique Host ID of a License Server

To obtain license codes for Creative Finishing software, you must obtain the unique host ID of the license server. This ID is used to confirm your Creative Finishing Software license and issue license codes.

> **NOTE** For a redundant network license server configuration, you must get the unique host ID of all license servers.

To get the unique host ID of the license server:

1 Open a terminal and type:

   /usr/local/bin/dlhostid

   A message appears indicating the *dlhostid* of the machine. For example (your value will differ):

   ```
   The Discreet host ID of this machine is
   "DLHOST01=25231AEF83AD9D5E9B2FA270DF4F20B1"
   ```

Requesting License Codes

Request licensing codes from the Autodesk M&E Edge support portal: *https://edge.autodesk.com/LicenseAssignment*.

If you do not have an account on the support portal, email *me.support@autodesk.com* to make a request.

To request license codes:

1 Make sure you have the unique ID of your single license server (local or remote) or a set of three unique IDs for each server of a redundant network license server configuration ready before you request your license codes. See Getting the Unique Host ID of a License Server (page 109).

2 Log in to support portal: *https://edge.autodesk.com/LicenseAssignment*.

3 Follow the links to set up your license server assets and select the software that you want to access using a network license.

   Your license codes are emailed to you.

Licensing the Workstation with a Local or Single Remote License Server

After you receive your license for a local or a single remote license server from Autodesk, use the license wizard to license to license your workstation.

NOTE For a redundant network license server configuration, you must create the license files for the workstation and license server manually. See Creating a License File for a Remote License Server (page 111) and Configuring the Workstation to Use a Set of Redundant License Servers (page 112).

The license wizard creates the following license files:

■ Workstation license: */usr/local/flexlm/licenses/DL_licenseNetwork.dat*

■ If you are running a local license server, it creates the license server license: */usr/discreet/licserv/licenses/DL_license.dat*. If you are using a remote license server, you must create the license file for the license server manually. See Creating a License File for a Remote License Server (page 111).

To license the workstation with a local license or a single remote license server:

1 Start the application. If no license file is found, the License Wizard opens.

2 Select "Yes, install the license code" and click Next.

3 In the License Installation dialog, do one of the following:
   ■ If the license server is on the same machine as your Creative Finishing application, select Local Network License.
   ■ If the license server is on another machine, select Remote Network License.

4 Click Next.

5 Open the license file you received from Autodesk in a text editor or display its contents in the shell.

6 In the Local Network License Installation dialog, select the license text and use the middle-mouse button to copy and paste the text into the field. The text should resemble the following:

```
SERVER servername
DLHOST03=2BB094B4722BDBECFA218BD3A6BB8F83
VENDOR discreet_l
USE_SERVER
PACKAGE smokeadvanced_x86_64_r discreet_l \
```

Ensure the servername (in the first row) is replaced with the actual name of the computer your license server is on.

7   Click Install. If you installed a local license server, restart the license server. See Starting the License Server (page 114). If you are setting up a remote license server, see Creating a License File for a Remote License Server (page 111).

Creating a License File for a Remote License Server

After you receive your license codes, edit the */usr/discreet/licserv/licenses/DL_license.dat* license file on the license server (or all three servers, if you are configuring redundant license servers), and enter the license codes you received from Autodesk.

The license information contains keywords that identify the license server, as well as the license strings for products or features.

Keyword	Description
SERVER	Specifies the hostname of the license server from which to obtain the license, followed by its *dlhostid*.
VENDOR	Specifies the daemon that is serving the license.
USE_SERVER	Indicates whether the system should obtain its license from a license server.
FEATURE	License strings for the software and feature entitlements.

To create the license server file on a license server:

1   Log in as root to the license server.

2   Navigate to the licenses directory by typing:

   cd /usr/discreet/licserv/licenses

3   If the file *DL_license.dat* does not exist in the directory, create it by typing:

   touch DL_license.dat

4   Open the file *DL_license.dat* in a text editor.

5 Enter the information provided by Autodesk in this file. It should be similar to the following shortened example:

```
SERVER server DLHOST01=886C2B75E8E57E4B03D784C3A2100AC0
```

NOTE If you are setting up redundant a network license, there are three SERVER strings.

```
VENDOR discreet_l
USE_SERVER
FEATURE flameEDU_x86_64_b_2011_1 2011.99 02-oct-2010 \
```

NOTE If you are working in an existing *DL_license.dat* file, make sure you do not accidentally alter any of the existing text in the file when entering the Creative Finishing Software license information.

6 Save and close the file.

This file sets up the network licenses available for distribution by the license server to the Creative Finishing workstations on your network.

Configuring the Workstation to Use a Set of Redundant License Servers

To configure the workstation to use a set of redundant license servers:

1 Log in as root to the workstation.

2 Navigate to the licenses directory by typing:

cd /usr/local/flexlm/licenses

3 If the file *DL_licenseNetwork.dat* does not exist in the directory, create it by typing:

touch DL_licenseNetwork.dat

4 Open the file *DL_licenseNetwork.dat* in a text editor.

5 Enter the SERVER, VENDOR and USE_SERVER strings provided by Autodesk. It should be similar to the following shortened example:

```
SERVER host01 DLHOST01=886C2B75E8E57E4B03D784C3A2100AC0
27005
```

```
SERVER host02 DLHOST01=886C2B75E8E57E4B03D784C3A2100AC1
27005
```

```
SERVER host03 DLHOST01=886C2B75E8E57E4B03D784C3A2100AC2
27005
VENDOR discreet_l
USE_SERVER
INCREMENT flameEDU_x86_64_b_2011_1 2011.99 02-oct-2010
\
```

6   Save and close the file.

This file sets up the floating licenses available for distribution by the license server to the Creative Finishing workstations on your network.

Changing the Default Port Used by the License Server

To avoid conflicts with other applications or license servers in your facility, you may need to change the default port setting used by the license server, or license servers in the case of a redundant configuration. This requires a minor change to the *DL_license.dat* file on your license server(s), as well as on every Creative Finishing Softwareworkstation on the network.

To change the default port used by a license server:

1   Log in as root to the system where the license server is installed.

2   Open the */usr/discreet/licserv/licenses/DL_license.dat* file in a text editor.

3   Find the SERVER line.

By default, no port number is specified at the end of the SERVER line for a single license server and the license server uses default port number in the range of 27000-27009. By default, redundant license servers are set to port 27005.

Single license server example:

```
SERVER server DLHOST01=886C2B75E8E57E4B03D784C3A2100AC0
```

Redundant license servers example:

```
SERVER host01 DLHOST01=886C2B75E8E57E4B03D784C3A2100AC0
27005
SERVER host02 DLHOST01=886C2B75E8E57E4B03D784C3A2100AC1
27005
SERVER host03 DLHOST01=886C2B75E8E57E4B03D784C3A2100AC2
27005
```

4 Enter a different port at the end of the SERVER line.

The license server can use a default port number in the range of 27000-27009.

For example, to use port 62222, add 62222 to the end of the line:

```
SERVER server DLHOST01=886C2B75E8E57E4B03D784C3A2100AC0
62222
```

5 Save and close the file.

6 Verify that the new port settings are correct. Type the following commands:

/etc/init.d/license_server stop

/etc/init.d/license_server start

cat /usr/discreet/licserv/log/license_server.log

NOTE For redundant license servers, reboot each server in close sequence to properly restart the license system.

7 Look for messages similar to the following examples in the output, and verify that the port numbers are what you requested:

```
15:08:49 (lmgrd) lmgrd tcp-port 6222215:08:49 (lmgrd)
Starting vendor daemons ...15:08:49 (lmgrd) Using vendor
daemon port 12344 specified in license file15:08:49
(lmgrd) Started discreet_l (internet tcp_port 12344 pid
5013)
```

8 Log in as root to each workstation on the network, open the file */usr/local/flexlm/licenses/DL_license.dat* in a text editor, and repeat steps 3 and 4 in this procedure, using the same port as the one you set for the license server.

Starting the License Server

NOTE For redundant license servers, reboot each server in close sequence to properly restart the license system.

To start the license server:

1 Type the following in a shell:

/etc/init.d/license_server start

WARNING The license server cannot start unless the license is entered correctly in *DL_license.dat*. Check the *boot.log* file to make sure the license server is started and working properly.

NOTE The license server starts/stops automatically when the machine is booted/shut down. You can stop and restart the server manually by running one of the following commands:

/etc/init.d/license_server stop

/etc/init.d/license_server restart

Starting the Software for the First Time

Starting Visual Effects and Finishing Software for the First Time

You must license Inferno, Flame Premium, Flame, Flint, Smoke Advanced, Smoke and Backdraft Conform before you can start.

To start the software for the first time:

1 Log into your workstation.

2 Do one of the following:
   - If this is the first time you are starting the software on a new system, open a terminal and type **/usr/discreet/**`<product_home>`**/bin/startApplication -v** to initialize the framestore and start the application. For example:

     /usr/discreet/flame2010.1/bin/startApplication -v

     WARNING The **-v** option deletes all material on the framestore. Use this option only if you have no material that you want to preserve on the framestore.

     NOTE For information on all the command-line options available, see Overview (page 178).

   - If you want to start the application without initializing the framestore, double-click the application icon on the Desktop.

   The Autodesk Master Systems Software License Agreement appears.
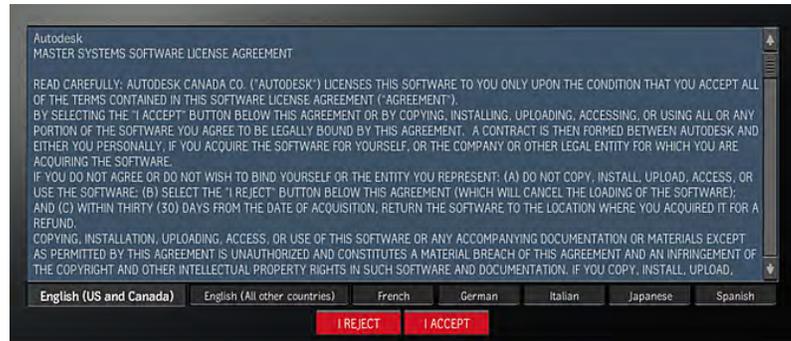
3 (Optional) To view the License Agreement in another language, click a language tab at the bottom of the License Agreement window.

The localized version of the License Agreement appears in a new window.

4 Read the text carefully, and click I Accept if you agree to all the terms in the License Agreement.

The Project Management menu appears.

5 Use the Project Management menu to create a project and a user.

For information on creating projects and users, see the application Help.

6 Click Start or press `Enter`.

7 When prompted to confirm the framestore initialisation, answer "yes".

After a few moments, the following message appears:

`Startup complete.`

You are ready to start working in the application. If you see a splash screen rather than the application interface, click anywhere on the screen.

NOTE If you use the pen and tablet while the application is starting, the tablet will fail to initialise. Press `Shift+T+Insert` to initialise the tablet if it does not function during a work session.

Starting Lustre for the First Time

You must license the Lustre software before you can start.

To start Lustre, log into the application's Linux user account and double-click the Lustre icon on the KDE desktop.

Lustre starts. The license you installed determines the type of station you are using (Lustre Master Station or Lustre Station).

When the Lustre UI appears, you can begin working.

For a new project, this includes:

■ Creating the project and setting its parameters (such as logarithmic or linear colour space).

■ Specifying the location of footage.

■ Setting up user profiles.

NOTE For information on creating projects and users, see the "Project Management" topic in the *Lustre Help*. To open the Help, press `Shift+F1`.

Advanced Configuration and Troubleshooting

6

Visual Effects and Finishing Configuration Files

Overview

Configuration files are ASCII text files that define the working environment for your application. Keywords in the configuration files define certain global and project-specific parameters such as I/O devices, workstation display settings, project environment directory paths, framerate, timecode, and colour space. You can open a configuration file using any text editor, such as *kedit*.

There are two kinds of configuration files:

The software initialisation configuration file	is parsed when you launch the software.
The project configuration file	is parsed when you select a project in the software.

Although you should not have to regularly edit these files, it is important that you become familiar with them for the occasional instance where editing them is necessary.

About the Software Initialisation Configuration File

The software initialisation configuration file (by default, */usr/discreet/*`<product_home>`*/cfg/init.cfg*) is parsed when you launch the software.

Keywords in the configuration file define parameters such as I/O devices, peripherals, workstation display settings, and environment directory paths. These parameters are independent of the project you select from the Project Management menu when you begin a session.

NOTE For information on each keyword in the software initialisation configuration file, see Software Initialisation File Keywords (page 127).

When you first install the software, you are prompted to edit the default software initialisation configuration file.

As you work with the software, you may have to edit this file under certain conditions, such as:

■ After making changes to your hardware configuration

■ When setting session-based keyword values, such as specifying an archiving device

You may choose to create different versions of the software initialisation configuration file to accommodate session-based needs that change through the course of a project.

To view the default software initialisation file, browse to the */usr/discreet/*`<product_home>`*/cfg/* directory, and open the *init.cfg* file in any text editor.

The file contains comments that provide brief descriptions of each keyword, keyword syntax, as well as usage examples.

In some cases keyword examples are not commented out and are used as the default settings. They are parsed during software initialisation and the corresponding parameter is set.

Sample Software Initialisation Configuration Files

The Visual Effects and Finishing application installation packages come with sample software initialisation configuration files for the usual needs of working with NTSC, PAL, HDTV, or film material.

These samples are located in the */usr/discreet/*`<product_home>`*/cfg/linux-x86_64/sample* directory.

You should never edit the sample configuration files. Instead, either make a copy of the sample software initialization configuration file that you like or just edit the real *init.cfg* file.

These samples are starting points toward configuring the software to best suit your needs.

Using a text editor, you can open a software initialisation configuration file for viewing. For example, you may want to read the keywords in different software initialisation configuration samples to see the differences. This is a good way to learn more about configuration keywords.

Modifying a Software Initialisation Configuration File

Software initialisation configuration files have open read/write permissions, and anyone can modify them. In some cases, you may want to restrict access to the files using the *chmod* command.

Here are some basic guidelines for modifying software initialisation configuration files:

- Do not modify the sample software initialisation configuration files. Make copies of the samples and work on those instead.

- Create a backup of a file before making any changes to it.

- After modifying a software initialisation configuration file, restart the software for the changes to take effect.

- To launch the software with a custom software initialisation configuration file, use the **-c** start-up option. For example:

  smoke -c /usr/discreet/smoke_2011/cfg/my_custom_init.cfg

  NOTE If you saved your custom software initialisation configuration file in a directory other than */usr/discreet/<product_home>/cfg*, be sure to use the full directory path. If the custom file is not found, the application automatically uses the default *init.cfg* file.

Updating the Software Initialisation Configuration File

The *xxdiff* utility is normally used during software installation to transfer facility-specific modifications in a customized software initialization configuration file from an earlier application version to the current one.

The following procedure describes how to manually run *xxdiff* to compare old and new configuration files. While you may never need to do this, this procedure can be useful for troubleshooting problems in your facility. For example, you can use *xxdiff* to troubleshoot problems in a workstation by comparing its *init.cfg* file against the *init.cfg* file of a workstation without problems.

NOTE If you are upgrading from an earlier version of the application, several changes may have been applied to the software initialisation configuration file. Because of this, you may want to select the entire contents of the new *init.cfg* configuration file and then customize it to avoid adding obsolete keywords.

To update the software initialisation configuration file:

1   In a terminal, start *xxdiff* and load the two files for comparison by typing the following:

    xxdiff`<old_file> <new_file>`

    For example, to launch *xxdiff* to compare the previous and current software configuration files, you would type:

    xxdiff /usr/discreet/`<product_home1>`**/cfg/init.cfg**
    /usr/discreet/`<product_home2>`**/cfg/init.cfg**

    where `<product_home1>` is the home directory for the previous version and `<product_home2>` is the home directory for the current version.

    The previous and new configuration files appear beside each other in an *xxdiff* window. The left panel displays the previous file settings and the right panel displays the current file settings. The path to both files is displayed in the title bar. The vertical bar on the right indicates where differences exist between the two panels.

2   Scan the old configuration file for custom directories or keywords that you want to copy to the new *init.cfg* file.

    On a line-by-line basis, the differences between the old and new configuration files are highlighted.

3 For each highlighted difference, click the line that you want to keep:

■ To keep the setting in the old configuration file, click the line in the left window.

■ To keep the setting in the new configuration file, click the line in the right window.

Highlighted lines change colour when you click them. You must select a line in the left or right window for each difference detected by *xxdiff*.

4 When you have finished updating, choose File | Save As Right to save the selected line differences under the name *init.cfg*.

5 Click OK to confirm, and then exit *xxdiff*.

About Project Configuration Files

A project configuration file is a text file that is parsed when you select a project in the application. A project configuration file uses the same name as the project, but with a *.cfg* extension. For example, the project configuration file for a project called "my_project" uses a project configuration file called *my_project.cfg*.

Project configuration files are located in the */usr/discreet/project/<project_name>/cfg* directory. You can open a project configuration file in any text editor.

NOTE Although you can modify project configuration files, the Project Management menu provides editing tools to modify projects from session to session as needed. In general, you should not have to modify project configuration files manually.

Project Configuration File Templates

When you create a project, you select a template from which your project's configuration file is generated. These templates are based on a resolution and framerate, and define the display environment. Project templates do not prevent you from working with clips of other resolutions and frame rates. However, some functions, such as display on a broadcast monitor, may not be supported.

These templates are located in the */usr/discreet*`<product_home>`*/cfg/linux-x86_64/template* directory.

TIP This directory contains an extensive list of templates for all supported display environments, some of which may not be suitable for your workflow. To make template selection easier, use the **mkdir** command to create a directory called *unused* in the directory */usr/discreet/*<product_home>/*cfg/linux-x86_64/template.* Use the **mv** command to move templates you do not use to this folder. When you use the Create Project menu, only those templates that remain will be available.

Select the template from the following table that corresponds to the resolution of your project.

The configuration file template:	Sets parameters for projects to work with:
ntsc.cfg	NTSC 601 material. The external sync source is NTSC black-burst.
pal.cfg	PAL 601 material. The external sync source is PAL blackburst.
24p.cfg	24p material of any resolution.
24p_ntsc.cfg	Film-based material that has been transferred to NTSC 601 video. The external sync source is NTSC blackburst.
film.cfg	Film-based material.
film_ntsc.cfg	Film-based material that has been transferred to NTSC video. The external sync source is NTSC blackburst.
24p_1920x1080@5994i.cfg *24p_1280x720@5994p.cfg* *24p_1280x720@5994p_free.cfg* *24p_1280x720@6000p.cfg* *24p_1280x720@6000p_free.cfg* *24p_1920x1080@5994i.cfg*	Film-based material that has been transferred to HD video. The external sync source is NTSC blackburst.
1280x720@5994p_free.cfg *1280x720@6000p_free.cfg* *1920x1080@23976psf_free.cfg* *1920x1080@24000psf_free.cfg* *1920x1080@25000psf_free.cfg* *1920x1080@50i_free.cfg*	The corresponding HD standard on workstations to which no external sync source is connected.

The configuration file template:	Sets parameters for projects to work with:
1920x1080@5994i_free.cfg *1920x1080@6000i_free.cfg*	
1280x720@5994p.cfg *1280x720@6000p.cfg* *1920x1080@23976psf.cfg* *1920x1080@24000psf.cfg* *1920x1080@25000psf.cfg* *1920x1080@50i.cfg* *1920x1080@5994i.cfg* *1920x1080@6000i.cfg*	The corresponding HD standard on workstations. The external sync source corresponds with the HD format.

Supported HD Formats in Project Templates

The names of project templates for HD formats contain a string that identifies the format's resolution, framerate, and scan mode.

Project configuration templates with the following string in their name:	Correspond to the following HD format:
1280x720@5994p	1280x720 progressive scan at 59.94 frames per second (720/59.94p, SMPTE-296M)
1280x720@5000p	1280x720 progressive scan at 50 frames per second (720/50p, SMPTE-296M)
1280x720@6000p	1280x720 progressive scan at 60 frames per second (720/60p, SMPTE-296M)
1920x1080@23976psf	1920x1080 progressive segmented frame at 23.976 frames per second (1080/23976psf, SMPTE-274M, RP-211)
1920x1080@24000psf	1920x1080 progressive segmented frames at 24 frames per second (1080/24psf, SMPTE-274M, RP-211)

Project configuration templates with the following string in their name:	Correspond to the following HD format:
1920x1080@25000psf	1920x1080 progressive segmented frames at 25 frames per second (1080/25psf, SMPTE-274M, RP-211)
1920x1080@50i	1920x1080 interlaced scan at 50 fields per second (1080/50i, SMPTE-274M)
1920x1080@5994i	1920x1080 interlaced scan at 59.94 fields per second (1080/59.94i, SMPTE-274M)
1920x1080@6000i	1920x1080 interlaced scan at 60 fields per second (1080/60i, SMPTE-274M)

Modifying Project Configuration Files

Project configuration files have open read/write permissions and anyone can modify them. In some cases, you may want to restrict access to them using the *chmod* command.

Here are some basic guidelines for modifying project configuration files:

■ Do not modify the template project configuration files. These are used by the software to generate new project configuration files and should not be altered. Create a copy of the desired project *.cfg* file, and modify that in a text editor as needed.

■ Create a backup of the project configuration file before making any changes to it.

■ Modified project configuration files must end with *_custom.cfg*.

■ You cannot use multiple project configuration files with a single project. A project configuration file must have the same name as its corresponding project and it must be located in the project's *cfg* directory.

Software Initialisation File Keywords

This section describes each keyword in the software initialisation configuration file */usr/discreet/*`<product_home>`*/cfg/init.cfg*.

DiskHealthNotify

The `DiskHealthNotify` keyword defines an e-mail address to which a message is sent if any framestore disk health problems occur. Only one address can be listed. The mailer daemon for the workstation must also be properly configured for notification e-mails to be sent.

The DiskHealthNotify keyword uses the following syntax

```
DiskHealthNotify <e-mail_address>
```

where `<e-mail_address>` is the address to which you want the message sent.

Example of use

```
DiskHealthNotify sysadmin@here.com
```

MemoryApplication

The `MemoryApplication` keyword allocates memory for frame buffers. A frame buffer is a chunk of memory set aside to store image data in RAM for the fastest possible display of frames while working in the application.

In most cases, the `MemoryApplication` keyword should be commented out. When it is commented out, the default memory configuration is used, and is based on the physical memory installed on the system. This default configuration is sufficient for most projects, regardless of the different resolutions you may be working with.

Uncommenting the `MemoryApplication` keyword overrides the default memory configuration. You should only override the default memory configuration by uncommenting the `MemoryApplication` keyword under the following conditions:

■ You are working with 4K 12-bit material.

■ You are working in the application and receive "cannot allocate memory buffer" messages.

■ Your system fails due to memory panic.

The MemoryApplication keyword uses the following syntax

```
MemoryApplication <megabytes>
```

where `<megabytes>` is the amount of memory dedicated to the corresponding token, in megabytes.

Example of use

```
MemoryApplication 400
```

Working with 4K 12-bit Film

Even with the recommended amount of RAM, working with 4K 12-bit film puts great demands on memory management. Set the RAM allocated for buffering frames to between 2000 and 4000.

The ability to mix different resolutions in a 4K 12-bit film project increases memory requirements even further. Even with an override of the default memory management configuration, working at this resolution may not support some advanced features that require a large amount of memory such as the Colour Warper™.

Cannot Allocate Memory Buffer Messages

"Cannot allocate memory buffer" messages indicate a shortage of memory allocated to frame buffering, so uncommenting the `MemoryApplication` keyword and defining an amount of memory for frame buffers can resolve this problem. The amount to set in this case depends on how much RAM your system has: start with 400 MB for Finishing applications or 700 MB for Visual Effects applications. Allocate more memory to frame buffers in increments of 50 MB as required.

Memory Panic

A memory panic indicates the system is running out of heap memory. Heap memory is memory allocated for use by applications and the operating system.

Make sure no other applications are running. If memory panic still occurs, lower the amount of memory allocated to frame buffers.

Remember that overriding the default memory management configuration is only required under these specific conditions. If your system experiences problems related to memory management under normal conditions, contact .

Video

The Video keyword initialises the video device and specifies the video input type.

Video device	The video input devices supported are AJA_KONA3G and AJA_OEM2K.
Default video input format	The default video input format that corresponds to the input configuration of the video card. This setting is overridden by the video input formats specified by the lines you uncomment for each VTR under the Vtr keyword heading.

The Video keyword uses the following syntax

    Video <video_device>, <default_video_format>

Where:	Is:
`<video_device>`	The video card used by your workstation. It can take the following values: AJA_KONA3G AJA_OEM2K
`<default_video_format>`	The video input type. For all AJA cards, the supported video input types are `Serial1` and `SerialDual`.

Examples of use

    Video AJA_KONA3G, Serial1

DviRamp

The `DviRamp` keyword enables control of the Miranda DVI-Ramp external device. The DVI-Ramp outputs a standard definition serial digital video signal (SMPTE-259M-C) or a high-definition serial digital video signal (SMPTE-292M).

NOTE Enable this keyword only if you are using the Miranda DVI-Ramp external device. The `DviRamp` keyword is automatically commented out if the application installation script detects a NVIDIA® SDI card connected to the system.

The DviRamp keyword uses the following syntax

```
DviRamp <serial port>, <baud rate>, <parity>, <size>,
<stopBits>
```

Where:	Is:
`<serial_port>`	The type of connection to the DVI-Ramp external device. The values are: ■ `ttyS0` for a DVI-Ramp1 connected on a serial port. ■ `usb/dviramp` for a DVI-Ramp1 connected on a USB port using a USB-Serial adaptor. ■ `usb/dviramp2` for DVI-Ramp2 connected on a USB port.
`<baud rate>`	The baud rate used to communicate with the DVI-Ramp.
`<parity>`	The parity setting to communicate with the DVI-Ramp. It can be set to EVEN, ODD, or NOPARITY.
`<size>`	The number of transmitted bits per packet.
`<stopBits>`	The size of the stop bit.

Example of use

```
DviRamp ttyS0, 38400, none, 8, 1
```

VideoPreviewDevice

The `VideoPreviewDevice` keyword specifies the device used for the graphics-to-video display.

Specify entries for the resolutions of the projects on which you will be working as well as the resolutions supported by your hardware configuration.

The *init.cfg* file contains all supported tokens for the `VideoPreviewDevice` keyword. The project configuration file will determine which one of these should be enabled for a given project.

`VideoPreviewDevice` is also used in conjunction with the `VideoPreviewWindow` keyword in the project configuration template files; `VideoPreviewWindow` specifies the width and height of the window and the refresh rate. See VideoPreviewWindow (page 158).

When the application is started and a project is selected, the `VideoPreviewWindow` must match one of the `VideoPreviewDevice` entries. If no match is found, graphics-to-video output is unavailable.

The `VideoPreviewDevice` keyword applies to workstations configured with either a Miranda DVI-Ramp external device, a NVIDIA SDI card, or graphics readback (SD timings only) if the hardware is not available.

The VideoPreviewDevice uses the following syntax

```
VideoPreviewDevice <description>, <device>, <channel_number>,
<xsize>, <ysize>, <refresh>, <syncsource> [, <xoffset>,
<yoffset>
```

Where:	Is:
`<description>`	A string describing the resolution and sync configuration of the video preview device: `ntsc`, `pal`, or an HD resolution and timing.
`<device>`	The graphics-to-video configuration for your system. Set to `dviramp` (if you are using a DVI ramp), `nvidia` (if you are using a NVIDIA SDI card), or `readback`.
`<channel_number>`	The channel on which the signal is output.
`<xsize>`, `<ysize>`	The width and height of the area of the image window output to the graphics-to-video display.

Where:	Is:
`<refresh>`	The refresh rate of the corresponding resolution.
`<syncsource>`	The reference sync. Set to `601sync` when you are using NTSC or PAL timing and you are connected to an external sync generator. Set to `DTVsync` when you are using HD timing and you are connected to an external sync generator. Set to `freesync` when you are not connected to a sync generator.
`<xoffset>, <yoffset>`	The horizontal and vertical offset of the video sent to the broadcast monitor relative to the graphics display. These parameters are required for some resolutions to display accurate field dominance during playback. Use these optional parameters with NTSC.

Examples of use

```
VideoPreviewDevice ntsc, dviramp, 1, 720, 486, 30, 601sync,
0, 1
```

```
VideoPreviewDevice pal, nvidia, 1, 720, 576, 25, 601sync
```

```
VideoPreviewDevice 1920x1080@50i, nvidia, 1, 1920, 1080, 50i,
DTVsync
```

When working with variable framerate material, enable the video preview device token corresponding to the 720p timings, for example:

```
VideoPreviewDevice 1280x720@5994p, nvidia, 1, 1280, 720,
5994p, 601sync
```

Configuring Readback Mode

On systems without a DVIramp or a NVIDIA SDI card, you must enable the following lines to ensure video preview using the readback mode:

```
VideoPreviewDevice pal, readback, 2, 720, 576, 25, 601sync
```

```
VideoPreviewDevice ntsc, readback, 2, 720, 486, 30, 601sync
```

TabletDriver

The `TabletDriver` keyword identifies the tablet driver. Only Wacom®
Intuos-series USB tablets are currently supported.

The TabletDriver keyword uses the following syntax

```
TabletDriver wacom
```

MidiDevice

The `MidiDevice` keyword identifies MIDI devices that are used with Autodesk
Visual Effects and Finishing systems. Only one device is recognized at a time.

The MidiDevice keyword uses the following syntax

```
MidiDevice <name>, <device_configuration_file>, <serial_port>,
<protocol>, <baud_rate>, <parity>[, <stopbit>]
```

Where:	Is:
`<name>`	The name by which you want to identify the MIDI device in the application.
`<device_configuration_file>`	The name and path of the device configuration file for the MIDI device. The path is optional.
`<serial_port>`	The serial port to which the MIDI device is connected.
`<protocol>`	The protocol used to communicate with the MIDI device. It can be either `direct_RS422` or `direct_RS232`.
`<baud_rate>`	The baud rate used to communicate with the MIDI device.
`<parity>`	The parity setting to communicate with the MIDI device. It can be set to `EVEN`, `ODD`, or `NOPARITY`.
`<stopbit>`	The size of the stop bit (optional).

Example of use

```
MidiDevice myMidiDevice, midi_LUCID_ADA8824_A232, /dev/ttyS1,
direct_RS232, 9600, NOPARITY
```

Vtr

The Vtr keyword identifies video tape recorders that can be used for clip I/O. You can uncomment VTRs of different video formats. Any enabled VTR can be selected for a project, regardless of the project's video I/O timings.

You can also use the Vtr keyword to identify supported High-Speed Dual-Link (HSDL) devices for clip I/O, such as telecines. HSDL devices appear as separate entries in the list of supported decks for the Vtr keyword.

The Vtr keyword uses the following syntax

```
Vtr <protocol>, <name>, <input_format>,
<timing>,<colorspace_mode>, <output_format>, <output_sync>,
<serial_port>, <timecode_type>, <video_output_delay>,
<video_input_delay>, <pre_roll>, <post_roll>,
<audio_input_delay>, <audio_output_delay>, <video_precision>
[, <cueup_mode>, <TC_transition_delay>, <edit_on_delay>,
<edit_off_delay>, <vtr_command_delay>]
```

Where:	Is:
<protocol>	The VTR control protocol (SONY, BTS, BVW50, TASCAM, or NONE).
<name>	The name by which you want to identify the VTR in the Input Clip and Output Clip menus (D1, DigBeta, D1 BTS, D5, and DVCpro, for example).
<input_format>	The video input format. Set to Serial1 to input using a single-link (4:2:2) connection from a device. Set to SerialDual to input using a dual-link (4:4:4) connection from a device.
<timing>	The I/O timing associated with the video standard of the VTR (NTSC, PAL, or HD, if applicable).

Where:	Is:
`<colorspace_mode>`	The mode that indicates whether colourspace conversion and/or headroom is required for clip I/O. See Configuring Colourspace Conversion for Device I/O (page 136).
`<output_format>`	The video output type. Set to `Serial1` to output using a single-link (4:2:2) connection to a device. Set to `SerialDual` to output using a dual-link (4:4:4) connection to a device.
`<output_sync>`	The sync source used for clip output. For AJA cards, the possible values are `STANDALONE, HOUSE,` or `DIGITAL1`.
`<serial_port>`	The serial port to which the VTR is connected. It takes the value `AJA:0:1` for workstations using an AJA KONA 3G or OEM-2K card.
`<timecode_type>`	The timecode type to be returned by the VTR (`Auto, LTC,` or `VITC`).
`<video_output_delay>`	Video output delay in frames.
`<video_input_delay>`	Video input delay in frames.
`<pre_roll>`	Preroll in frames or seconds: Use integers to specify preroll in frames. Use decimals to specify preroll in seconds.
`<post_roll>`	Postroll in frames or seconds: Use integers to specify postroll in frames. Use decimals to specify postroll in seconds.
`<audio_input_delay>`	The offset value to have sync audio with video on input. Integer units represent frames; this delay should be 0 by default.
`<audio_output_delay>`	The offset value to have sync audio with video on output. Integer units represent frames; this delay should be 0 by default.
`<video_precision>`	The precision of the video interface (`8` or `10 bits`).
`<cueup_mode>`	The method by which the VTR is cued. This parameter is optional. Use `vtrcueing` to make the application use the VTR Cue command

Where:	Is:
	directly to cue the VTR. Use `vtrff` if using `vtrcueing` causes the VTR to react slowly when cued, such as with a Betacam SP™.
`<TC_transition_delay>`	The delay in milliseconds after a vertical sync, before requesting the VTR timecode. This value should only be set or changed with the help of technical support.
	NOTE The value '99' is a special case to be used strictly with HDCAM-SR/720P and AJA. It is not to be used in other circumstances unless specified by technical support.
`<edit_on_delay>`	The delay in frames before the edit sync point to send the ON command. This parameter is optional and it applies only to BVW50.
`<edit_off_delay>`	The delay in frames before the edit sync point to send the OFF command. This parameter is optional and it applies only to BVW50.
`<vtr_command_delay>`	The delay in milliseconds before the application sends certain commands to the VTR. This parameter is optional, but useful for older VTRs such as the BTS.

NOTE The *init.cfg* file contains a comprehensive list of `Vtr` keyword examples for supported devices. In most cases you just have to uncomment the lines for the device that you want to use. In some cases minor changes to parameters are required.

Configuring Colourspace Conversion for Device I/O

When you set the input or output format for a device in the `Vtr` keyword, you should also specify its colourspace conversion method using the `<colorspace mode>` parameter. The supported conversion methods depend on whether a single-link or dual-link connection is used for I/O with the

device. The following table shows the colourspace conversion methods that are available for single-link and for dual-link I/O connections.

Colourspace Conversion Method	Supported for Single-Link Serial (4:2:2) I/O	Supported for Dual-Link Serial (4:4:4) I/O
YCbCR -> RGB	Yes	No
YCbCR -> RGB + Headroom	Yes	No
No Conversion	No	Yes
No Conversion + Headroom	No	Yes

Emulator

Use the `Emulator` keyword to enable the VTR Emulation feature and configure your workstation to emulate a Sony™ VTR that is controllable via the RS-422 serial port. For help configuring the hardware for VTR emulation, see the hardware setup guide for your workstation.

The VTR Emulation feature supports SD and HD video timings. This feature is available in Smoke Advanced.

You can specify more than one emulator. Any enabled emulator can be selected for a project, regardless of the project's video I/O timings.

The Emulator keyword uses the following syntax

```
Emulator sony, <name>, <input_format>, <timing>,
<colorspace_mode>, <output_format>, <output_sync>,
<serial_port>, <emulator_output_delay>,
<emulator_input_delay>, <audio_input_delay>,
<audio_output_delay>, <video_precision (8 to 10 bits)>
```

Where:	Is:
`<name>`	The name for the emulator.
`<input_format>`	The video input format. For a list of the video formats supported by your workstation, see Video (page 129).

Where:	Is:
`<timing>`	The video resolution and timing of the VTR to be emulated.
`<colorspace_mode>`	The colourspace mode for video transfers.
`<output_format>`	The video output format. For a list of the video formats supported by your workstation, see Video (page 129).
`<output_sync>`	The source used to synchronize the video output. By default, this value is set to *Autodetect*.
`<serial_port>`	The serial port to which the VTR control cable is connected.
`<emulator_output_delay>`	The video output delay used by the emulator in frames.
`<emulator_input_delay>`	The video input delay used by the emulator in frames.
`<audio_input_delay>`	The audio input delay used by the emulator in frames.
`<audio_output_delay>`	The audio output delay used by the emulator in frames.
`<video_precision>`	The interface precision used for video transfers.

Examples of use

```
Emulator sony, NTSC, SERIAL1, NTSC, YCBCR_RGB_CONVERSION,
SERIAL1, HOUSE, AJA:0:1,-5, 1, 0.00, 0.00, 8
```

```
Emulator sony, HSDL 1499 sf, SERIALDUAL, 2048x1556_1499SF,
NO_CONVERSION, SERIALDUAL, STANDALONE, AJA:0:1, -6, 1, 0.00,
0.00, 10
```

Audiodevice

The `Audiodevice` keyword initialises the specified audio device.

The Audiodevice keyword uses the following syntax

```
AudioDevice <type>
```

where `<type>` can only be `AJA`.

`AJA` specifies that the audio subsystem is part of the AJA audio/video I/O device, respectively. Note that for the audio to work, the `Video` keyword must be configured with the corresponding video device.

Example of use

```
Audiodevice AJA
```

ClipMgtDevice

The `ClipMgtDevice` keyword defines the devices used for archiving.

The following archiving devices are supported:
■ VTR
■ HDCAM
■ Tape (SAIT and DTF2 fibre archiving devices only)
■ File

You can set multiple archiving devices for use with the application. The first device that is defined will be the default device.

ClipMgtDevice Vtr

You can specify only one VTR for a VTR clip management device. The `ClipMgtDevice Vtr` keyword contains an optional start timecode parameter:
■ If you specify a default start timecode for VTR archiving, the specified timecode is always used. You cannot override the default start timecode using the Archive menu.
■ If you want to determine through the application the start timecode for archives created with a VTR, do not add the start timecode parameter.

The ClipMgtDevice Vtr keyword uses the following syntax

```
ClipMgtDevice Vtr[, <timecode>]
```

where `<timecode>` is an optional start timecode for the archive.

Example of use

```
ClipMgtDevice Vtr, 01:00:00:00
```

ClipMgtDevice HDCAM

To archive to a Sony HDCAM VTR, you must use the `ClipMgtDevice HDCAM` keyword and not the `ClipMgtDevice Vtr` keyword. The `ClipMgtDevice HDCAM` keyword adjusts the metadata encoding mechanism to account for the compression method used by the HDCAM VTR.

The `ClipMgtDevice HDCAM` keyword contains an optional start timecode parameter:

■ If you specify a default start timecode for HDCAM VTR archiving, the specified timecode is always used. You cannot override the default start timecode using the Archive menu.

■ If you want to determine through the application the start timecode for archives created with an HDCAM VTR, do not add the start timecode parameter.

The ClipMgtDevice HDCAM keyword uses the following syntax

```
ClipMgtDevice HDCAM[, <timecode>]
```

where `<timecode>` is an optional start timecode for the archive.

Example of use

```
ClipMgtDevice HDCAM, 01:00:00:00
```

ClipMgtDevice Tape

You can specify several tape devices for clip management. SCSI tape archiving devices are not supported. Use only fibre channel archiving devices, specifically, SAIT and DTF2 devices.

To use a tape device for archiving, you must define the filename, the block size, and the name for the device you are using.

The ClipMgtDevice Tape keyword uses the following syntax

```
ClipMgtDevice Tape, <file_name>, [<block_size>, [<device_name>
```

Where:	Is:
`<file_name>`	The filename of the fixed block size device.
`<block_size>`	The amount of data per block written to tape.
`<device_name>`	The name of the tape device as it will appear in the Archive menu.

Examples of use

Tape Device	Keyword Example
DTF2	`ClipMgtDevice Tape, /dev/st0, 65536, DTF2`
SAIT	`ClipMgtDevice Tape, /dev/st0, 65536, SAIT`

ClipMgtDevice File

You can define a portion of your system disk or another volume as the destination for archives created using the application.

The ClipMgtDevice File keyword uses the following syntax

```
ClipMgtDevice File, <file_name>, <size>
```

Where:	Is:
`<file_name>`	The path for archives created using the application.
`<size>`	The maximum size for a file archive in MB.

You can also modify the parameters of this keyword through the application.

- You can use a path other than the default filename by entering one in the Archive menu.

- You can change the default maximum size of a file archive in the Archive menu in the application.

Example of use

```
ClipMgtDevice File, /var/tmp/archive, 1024
```

MaxLibrarySize

The `MaxLibrarySize` keyword indicates the maximum size for any single clip library, in megabytes. The higher the value, the more memory the application uses. Using a larger value reduces memory fragmentation, which optimizes memory use. However, the value should not be so high as to compromise system performance.

The software also uses this keyword to determine whether there is enough free hard drive space available at start-up. To start the software, you must have at least 10 megabytes of free hard drive space in addition to the value set by this keyword.

The size of your libraries can be determined using the following command in a terminal:

ls -lh /usr/discreet/clip/*/*/*.000.clib

The MaxLibrarySize keyword uses the following syntax

```
MaxLibrarySize <size>
```

where `<size>` is the maximum library size in megabytes.

Example of use

```
MaxLibrarySize 75
```

ArchiveLibrary

The `ArchiveLibrary` keyword identifies the directory to which online HTML and ASCII tables of contents are saved when archiving. Its default value is *usr/discreet/archive*.

You can make the target directory relative to the home directory of the application by prefixing the path with a tilde (e.g. *~/archive*).

NOTE The `ArchiveLibrary` keyword of the project configuration file, if set, has precedence over the one in the software initialisation configuration file.

The ArchiveLibrary keyword uses the following syntax

```
ArchiveLibrary <directory_path>
```

where `<directory_path>` is the path to which online HTML and ASCII tables of contents are saved.

Example of use

```
ArchiveLibrary /usr/discreet/archive
```

SetupArchiveTape

The `SetupArchiveTape` keyword identifies the device to which setup information for an archived project is saved. A project's setups are saved as a *.tar* format archive, creating a single file that can be extracted, preserving the original directory structure.

The SetupArchiveTape keyword uses the following syntax

```
SetupArchiveTape <device_type>
```

where `<device_type>` is the path to the device where you want to save the setup archive. You can set the path to point to a tape device or use a file destination as a virtual device. If you set a file destination, you must add a filename ending with the *.tar* extension that you want to use for the setup archive.

Example of use

```
SetupArchiveTape /var/tmp/setups.tar
```

Environment Directory Pathnames

The three keywords in the Environment Directory Pathnames section specify the paths to directories for resources shared by all projects. These directory paths should not be modified.

Menu

The `Menu` keyword specifies where application menu files are stored.

Example of use

```
Menu ~/menu, menu
```

where the ~ in the directory path stands for */usr/discreet/*`<product_home>`.

Model

The `Model` keyword specifies where 3D models used by Action are stored.

Example of use

```
Model ~/dve
```

where the ~ in the directory path stands for */usr/discreet/*`<product_home>`.

HtmlTemplates

The `HtmlTemplates` keyword specifies where HTML templates used for archive online tables of contents are stored.

Example of use

```
HtmlTemplates ~/templates
```

where the ~ in the directory path stands for */usr/discreet/*`<product_home>`.

HtmlLog

The `HtmlLog` keyword allows you to specify a directory in which to write the Batch module HTML status and log.

The HtmlLog keyword uses the following syntax

```
HtmlLog <directory_path>
```

where `<directory_path>` is the destination directory.

Example of use

```
HtmlLog /usr/discreet/html
```

TextDefaultFont

The `TextDefaultFont` keyword sets the default font for the Text, Paint, and Action modules.

The TextDefaultFont keyword uses the following syntax

```
TextDefaultFont <font>
```

where `<font>` is the name of the font you want to set as the default.

Example of use

```
TextDefaultFont Discreet
```

FontDPSBase

The `FontDPSBase` keyword identifies the directory in which PostScript® fonts are stored. At initialisation, the application creates links in the */usr/discreet/font* directory that point to fonts in the directory identified by this keyword.

The FontDPSBase keyword uses the following syntax

```
FontDPSBase <directory_path>
```

where `<directory_path>` identifies the directory in which fonts are stored.

Example of use

```
FontDPSBase /usr/lib/X11/fonts/Type1
```

If you do not specify the directory pathname, the application uses */usr/lib/DPS/outline/base*. In most cases this directory—created when you installed the Display PostScript software as part of the installation—should be the one identified by the `FontDPSBase` keyword. Using this directory provides access to PostScript fonts.

NOTE To use TrueType fonts with the application, copy these fonts into the */usr/discreet/font* directory. For the application to recognize TrueType fonts in this directory, they must have the *.ttf* extension.

FontDPSAFM

The FontDPSAFM keyword identifies the directory in which font metrics are stored. Font metrics provide information about each font that improves kerning. At initialisation, the application creates links in the directory */usr/discreet/font* that point to font metric files in the directory identified by this keyword.

The FontDPSAFM keyword uses the following syntax

```
FontDPSAFM <directory_path>
```

where <directory_path> identifies the directory in which font metrics are stored.

Example of use

```
FontDPSAFM /usr/lib/X11/fonts/Type1
```

If you do not specify a directory pathname, the application uses */usr/lib/DPS/AFM*. In most cases this directory—created when you installed the Display PostScript software as part of the installation—should be the one identified by the FontDPSAFM keyword.

FontProxyLowString

The FontProxyLowString keyword, along with the FontProxyHighString keword, specifies which characters to draw in font proxies. By default, the proxy string is "Aa". You can override the default string for non-extended character sets by uncommenting and editing the FontProxyLowString keyword option.

Both FontProxyLowString and FontProxyHighString keyword options can be uncommented at the same time. Extended character sets try the FontProxyHighString keyword first. If the values in the FontProxyHighString keyword option do not apply to the font, the FontProxyLowString keyword option is used instead.

The FontProxyLowString keyword uses the following syntax

```
FontProxyLowString <code>[, ...]
```

Where:	Is:
`<code>`	The Unicode value associated with the character that you want to display.
`...`	Up to four (for a total of five) more Unicode values for the font proxy string.

Example of use

```
FontProxyLowString 0x41, 0x61
```

This example displays the default proxy string, "Aa".

FontProxyHighString

The `FontProxyHighString` keyword, along with the `FontProxyLowString` keword, specifies which characters to draw in font proxies. By default, the proxy string is "Aa". If a font includes glyph definitions for extended character sets (such as Asian character sets), you can set a proxy string by uncommenting and editing the `FontProxyHighString` keyword.

Both `FontProxyHighString` and `FontProxyLowString` keyword options can be uncommented at the same time. Extended character sets try the FontProxyHighString keyword first. If the values in the `FontProxyHighString` keyword option do not apply to the font, the `FontProxyLowString` keyword option is used instead.

The FontProxyHighString keyword uses the following syntax

```
FontProxyHighString <code>[, ...]
```

Where:	Is:
`<code>`	The Unicode value associated with the character that you want to display.
`...`	Up to four (for a total of five) more Unicode values for the font proxy string.

Example of use

```
FontProxyHighString 0x3042, 0x30a2
```

This example displays the Japanese "Hiragana A" and "Katakana A" characters.

TextFileEncoding

The `TextFileEncoding` keyword is the list of file encodings that will be supported for importing text files. The encoding must be supported for "iconv". To get the list of supported encodings, type **iconv -l** in a terminal.

NOTE The current local encoding and UCS-2 unicode encoding are always included by default.

The TextFileEncoding keyword uses the following syntax

```
TextFileEncoding <character_set>
```

Examples of use

```
TextFileEncoding ISO8859-1

TextFileEncoding EUCJP
```

Image File Extensions

This list defines the supported image and movie file formats that the application can input or output. These extensions are used to filter files of the corresponding format when you use the file browser. This list may be edited to suit your particular needs.

Image Format	Extension
Alias®	als
Cineon®	cin
Digital Picture Exchange	dpx
Jpeg	jpg
Pict	pict

Image Format	Extension
Pixar	picio
Sgi®	sgi
Softimage®	pic
Targa®	tga
Maya®	iff
Tiff	tif
Wavefront®	rla
Photoshop®	psd
OpenEXR	exr
REDCODE RAW	r3d
Quicktime®	mov
MXF	mxf
MPEG-4	mp4

Audio File Extensions

This list defines the supported audio file formats that the application can input or output. The extensions are used to filter files of the corresponding

format when you use the file browser. This list may be edited to suit your particular needs.

Audio Format	Extension
AIFF (standard)	aiff
AIFFC (extended)	aifc
Sun	au
Microsoft®	wav
Berkeley (BSD)	bsf
AVR (Audio Visual Research)	avr
MPEG-1 Layer3	mp3

DefaultWebBrowser

The `DefaultWebBrowser` keyword identifies the Web browser used by the application to browse the Help and view HTML tables of contents for archives.

The DefaultWebBrowser keyword uses the following syntax

`DefaultWebBrowser <browser>`

where `<browser>` identifies the Web browser you want to use.

Example of use

`DefaultWebBrowser firefox`

BackburnerManagerHostname

The `BackburnerManagerHostname` keyword serves two functions:
■ It locates the Backburner Manager on your rendering network. Enable this keyword if you are running an Autodesk Backburner background processing

network in your facility or will be using Cleaner® XL to encode jobs exported from the application. All four BackburnerManager keywords must be set correctly for jobs to be sent to the rendering network.

- It enables the Background Wire® and Background Proxies buttons in the application; they are otherwise greyed out. Uncomment the `BackburnerManagerHostName` keyword to enable the buttons.

The BackburnerManagerHostname keyword uses the following syntax

```
BackburnerManagerHostname <host_name>
```

where `<host_name>` is the hostname of the Windows® workstation that is running Backburner Manager.

Example of use

```
BackburnerManagerHostname titan.saturn.com
```

BackburnerManagerPriority

The `BackburnerManagerPriority` keyword sets the priority for jobs created on your application for Backburner Manager on your rendering network. Enable this keyword if you are running an Autodesk Backburner background processing network in your facility or will be using Cleaner XL to encode jobs exported from the application. All four BackburnerManager keywords must be set correctly for jobs to be sent to the rendering network.

The BackburnerManagerPriority keyword uses the following syntax

```
BackburnerManagerPriority <priority>
```

where `<priority>` is a value from 0 to 100, where 0 is highest priority and 100 is lowest. The default is 50.

Example of use

```
BackburnerManagerPriority 25
```

BackburnerManagerGroup

The `BackburnerManagerGroup` keyword defines the group of machines to which jobs created in your software application will be submitted. Enable this keyword if you are running an Autodesk Backburner background processing network in your facility or will be using Cleaner XL to encode jobs exported from the application. All four BackburnerManager keywords must be set correctly for jobs to be sent to the rendering network.

The BackburnerManagerGroup keyword uses the following syntax

```
BackburnerManagerGroup <group_name>
```

where `<group_name>` is the name of a group of computers on an Autodesk®
Burn® rendering network.

Example of use

```
BackburnerManagerGroup renderfarm1
```

BackburnerManagerGroupCapability

The `BackburnerManagerGroupCapability` keyword specifies whether the nodes in your rendering network are equipped with GPU-accelerated graphics cards or not. Based on the value of this keyword, the Visual Effects and Finishing application enables or disables the submission of jobs that require a GPU (such as floating point jobs) to the rendering network.

The BackburnerManagerGroupCapability keyword uses the following syntax

```
BackburnerManagerGroupCapability <group_capability>
```

where `<group_capability>` can be *software*, *gpu*, or *hybrid* depending on
the hardware of the nodes in the rendering network.

■ If none of the nodes in your rendering network are equipped with GPUs,
set the value of the keyword to *software*. The application does not send
jobs that require a GPU to the rendering network, but only jobs that can
be rendered in software mode (using OSMesa) by the render nodes.

■ If ALL the nodes in your rendering network are equipped with GPUs, set
the value of the keyword to *gpu*. The application sends all jobs to the
GPU-equipped nodes, even if some jobs do not specifically require a GPU

render node. The GPU-equipped render nodes will render jobs that require a GPU, as well as OSMesa jobs.

NOTE If your rendering network also contains render nodes without a GPU, but the `BackburnerManagerGroupCapability` keyword is incorrectly set to *gpu*, all jobs are sent only to GPU-equipped render nodes, and the nodes without a GPU are never used.

■ If your rendering network contains a mix of nodes with GPUs and without GPUs, set the keyword to *hybrid*. The application sends all jobs to the rendering network, and Backburner Manager distributes each job to the appropriate type of render node. Jobs that require a GPU are sent only to GPU-equipped render nodes, while jobs that do not require a GPU are sent to any available render node (GPU or non-GPU) to be rendered in software mode.

NOTE Set the BackburnerManagerGroupCapability keyword to *gpu* or *hybrid* only if you are sure that at least one node in your rendering network is equipped with a GPU. Attempting to submit a job that requires a GPU to a rendering network with no GPU-equipped render node results in the job being stuck in the rendering queue indefinitely.

Example of use

```
BackburnerManagerGroupCapability software
```

CleanerDestinationPath

The `CleanerDestinationPath` keyword sets the default path on a Windows workstation where clips are saved after encoding by Cleaner XL.

The default path you enter appears in the Cleaner Destination Path field when you select Cleaner in the Format Box of the Image Export menu. You can edit the path there. If there is an ftp path in the output profile, the ftp path is used as an additional destination for encoded jobs. See your application help.

The CleanerDestinationPath keyword uses the following syntax

```
CleanerDestinationPath <path>
```

By default, `<path>` is set to *C:\Tmp*, which is the default file path where Cleaner XL saves exported clips after encoding.

> **NOTE** Make sure the folder to which the path points exists and is accessible from the Windows workstation where Cleaner XL jobs will be encoded. Also make sure that this folder has write privileges and has enough space to accommodate encoded jobs.

Example of use

```
CleanerDestinationPath C:\Cleaner\exported
```

LogDailyReports

The `LogDailyReports` keyword specifies the number of application log files that are kept. When the number of application log files on the workstation exceeds this value, the oldest log is deleted to preserve space.

The LogDailyReports keyword uses the following syntax

```
LogDailyReports <number>
```

where `<number>` is the number of log files to be kept on the workstation. Set this value to 0 to keep all application log files.

Example of use

```
LogDailyReports 10
```

NetworkPanelDisplay

The `NetworkPanelDisplay` keyword filters the list of framestores in the Network panel to show framestores that are either available on the network and mounted, or available on the network but with undetermined mount status.

The NetworkPanelDisplay keyword uses the following syntax

```
NetworkPanelDisplay <filter_setting>
```

where <filter_setting> is set to either ShowMounted or ShowAll.

Use:	To:
ShowMounted	Show only framestores verified as both mounted and available on the Wire network.
ShowAll	Show all framestores available on the network without first checking whether they are mounted. Using this option slightly reduces the time needed for your application to start because no further checks are performed on remote framestores.

Example of use

```
NetworkPanelDisplay ShowMounted
```

Project Configuration File Keywords

Framerate

The Framerate keyword sets the default framerate for the project work session.

The Framerate keyword uses the following syntax

```
Framerate <frame_rate>
```

where <frame_rate> is 23.976, 24, 25, 29.97, 30, 50, 59.94 or 60 frames per second.

Example of use

```
Framerate 29.97
```

Timecode

The Timecode keyword sets the default timecode format used by the project.

The Timecode keyword uses the following syntax

```
Timecode <format>[, DF]
```

Where:	Is:
`<format>`	One of the following timecode formats: 23.976, 24, 25, 29.97, 30, 50, 59.94, or 60.
`DF`	An optional parameter indicating drop-frame timecode for 29.97 or 59.94 fps timecode formats.

Examples of use

Timecode Format	Keyword Example
30 fps	`Timecode 30`
29.97 fps (drop frame)	`Timecode 29.97, DF`

ColourSpace

The `ColourSpace` keyword defines how RGB/YUV conversions are handled. By default, for NTSC or PAL projects, the ITU-R BT.601-5 colour space is used. For DTV projects, the ITU-R BT.709-3 colour space is used.

The ColourSpace keyword uses the following syntax

```
ColourSpace <space_type>
```

where `<space_type>` is the colour space type used by the project.

Examples of use

Project	Keyword Example
NTSC and PAL standard projects (ITU-R BT.601-5)	`ColourSpace 601`
DTV standard projects (ITU-R BT.709-3)	`ColourSpace 709`
1920x1035 projects (SMPTE-240M)	`ColourSpace 240`

Hires

The `Hires` keyword sets the refresh rate of the graphics monitor. Note the following when setting the refresh rate:

■ The rate is measured in Hertz (Hz).

■ When using the video card in conjunction with an external sync source, the refresh rate should match the framerate.

■ If you are using a non-integer framerate (for example, 29.97), set the refresh rate to the nearest integer (in the previous example, 30).

The Hires keyword uses the following syntax

```
Hires [Custom,]<refresh_rate>
```

Where:	Is:
Custom	An optional parameter that overrides the application's refresh rate. For example, you may choose to override the 60-Hz refresh rate for an NTSC project by using the Custom option with a value of 30Hz.
<refresh_rate>	The refresh rate of the graphics monitor. Append Hz for free-running configurations, Hzf for configurations that are fixed to an external sync source.

Examples of use

Project Resolution	Keyword Example
NTSC	Hires 60Hz
PAL	Hires 50Hz
NTSC (external sync source)	Hires 30Hzf
PAL (external sync source)	Hires 25Hzf
Film	Hires 48Hz
24p Multi-Master Editing™	Hires 60Hz

Project Resolution	Keyword Example
Custom	`Hires Custom, 30Hz`

VideoPreviewWindow

The `VideoPreviewWindow` keyword is used when using a Miranda DVI-Ramp device. It locates a preview device that has a matching size and refresh rate in the `VideoPreviewDevice` list specified in the *init.cfg* configuration file. When you start the application or switch to another project from the application, the graphics-to-video display is mapped to the appropriate device according to the width, height, and refresh rate specified with this keyword.

The VideoPreviewWindow keyword uses the following syntax

`VideoPreviewWindow <xsize>, <ysize>, <refresh_rate>`

Where:	Is:
`<xsize>, <ysize>`	The horizontal and vertical size of the area of the image window output to the graphics-to-video display.
`<refresh_rate>`	The refresh rate of the output.

Example of use

`VideoPreviewWindow 720, 486, 30`

ArchiveLibrary

Online HTML and ASCII tables of contents will be saved to the filesystem. You can specify the destination directory by setting this keyword. Its default value is */usr/discreet/archive*. You can make the target directory relative to the project directory by prefixing the path with a tilde (e.g. *~/archive*).

NOTE This keyword, if set, will take precedence over the `ArchiveLibrary` keyword in the *init.cfg* file.

The ArchiveLibrary keyword uses the following syntax

```
ArchiveLibrary <directory_path>
```

where `<directory_path>` is the path to which online HTML and ASCII tables of contents are saved.

Example of use

```
ArchiveLibrary /usr/discreet/archive
```

Environment Directory

Environment directory keywords specify the paths to directories containing resources for all projects. For example, the LUTs that Autodesk provides are located in the directory */usr/discreet/*`<product_home>`*/lut*. You can place resource files in these directories that you want to be able to share between projects more easily.

Not all directories are relevant to the application. However, the inclusion of directories pertaining to other Visual Effects and Finishing products allows for cross-product project compatibility.

The ~ in the following table stands for */usr/discreet/*`<product_home>`.

Resource	Path	Resource File Extension
Action	*~/action*	`.action`
Audio	*~/audio*	`.audio`
AudioDesk	*~/audio*	`.audiodesk`
Audio ImpExp	*~/audio*	`.aiff`
AutoMatte	*~/automatte*	`.automatte`
AutoPaint	*~/paint/autopaint*	`.auto`
Average	*~/average*	`.average`

Resource	Path	Resource File Extension
Axis	*~/axis*	`.dve`
Batch	*~/batch*	`.batch`
BatchClip	*~/batchclip*	`.clip`
Blur	*~/blur*	`.blur`
Burnin	*~/burnin*	`.burnin`
ColourCurves	*~/ccurves*	`.ccurves`
ColourFrame	*~/colourframe*	`.col`
Colour Warper	*~/colourwarper*	`.cw`
Composite	*~/composite*	`.composite`
Compound	*~/compound*	`.compound`
Correct	*~/correct*	`.correct`
Cutout	*~/paint/cutout*	`.sgi`
Deal	*~/deal*	`.deal`
DeGrain	*~/degrain*	`.degrain`
DeInterlace	*~/deinterlace*	`.deinterlace`
Desktop	*~/desktop*	`.desk`
Difference	*~/difference*	`.diff`

Resource	Path	Resource File Extension
Dissolve	*~/dissolve*	`.dissolve`
Distort	*~/distort*	`.distort`
Documentation	*~/documentation*	`.pdf`
DVE	*~/dve*	`.dve`
Edge	*~/edge*	`.edge`
EDL	*~/edl*	`.edl`
Export	*~/export*	`.export`
ExprUserFun	*~/expressions/userfun*	`.expressions`
FieldMerge	*~/fieldmerge*	`.fieldmerge`
Filter	*~/filter*	`.filter`
FilmCompress	*~/filmcompress*	`.compress`
FilmExpand	*~/filmexpand*	`.expand`
Flip	*~/flip*	`.flip`
Geometry (Paint)	*~/paint/geometry*	`.geom`
GMask	*~/gmask*	`.GMask`
Guides	*~/guides*	`.guide`
Histo2d	*~/histo2d*	`.histo2d`

Resource	Path	Resource File Extension
HotKey	~/hotkey	`.hotkey`
Image	~/images	See Image File Extensions (page 148).
Import	~/import	`.import`
Interlace	~/interlace	`.interlace`
Key	~/key	`.key`
Keyer3d	~/keyer3d	`.key3d`
LensDistort	~/lensDistort	`.lensDistort`
Letterbox	~/letterbox	`.letterbox`
LogicOp	~/logicop	`.logicop`
Lumkey	~/lumkey	`.lumkey`
Lut	~/lut	`.lut`
Mask (Paint)	~/paint/mask	`.sgi`
MasterKeyer	~/masterkeyer	`.mkeyer`
Mix	~/mix	`.mix`
Moncal	~/monitor	`.monitor`
ModularKeyer	~/modularKeyer	`.modularKeyer`
Mono	~/mono	`.mono`

Resource	Path	Resource File Extension
Morf	*~/morf*	`.morf`
MotionAnalyse	*~/motionAnalyse*	`.motionAnalyse`
MotionBlur	*~/motionBlur*	`.motionBlur`
NormalMap	*~/normalmap*	`.normalmap`
Note	*~/note*	`.note`
Omf®	*~/images*	`.omf`
Optics	*~/optics*	`.optics`
Output	*~/output*	`.output`
Paint	*~/paint*	`.paintnode.xml`
Picture	*~/paint/picture*	`.sgi`
Play	*~/play*	N/A
Posterize	*~/posterize*	`.posterize`
Pulldown	*~/pulldown*	`.pulldown`
QComp	*~/quickcomp*	`.qcomp`
ReGrain	*~/regrain*	`.regrain`
Repeat	*~/repeat*	`.repeat`
Resize	*~/resize*	`.resize`

Resource	Path	Resource File Extension
Restore	*~/filmrestore*	`.restore`
Spark	*~/sparks*	`.spark`
Stabilizer	*~/stabilizer*	`.stabilizer`
Text	*~/text*	`.ttg`
Timewarp	*~/timewarp*	`.timewarp`
VectorViewer	*~/vectorViewer*	`.vectorViewer`
Wipe	*~/wipe*	`.wipe`

Lustre Software, Project, and User Configuration Files

Configuration File Overview

The configuration files define all the necessary settings your software needs to recognize various hardware and software settings on the Master Station and on the dedicated render station.

Before doing any colour correction in Lustre, you must first log into a project with a valid user profile. See the "Project Management" chapter in the *Autodesk Lustre User Guide*. When you create a user profile and project in Lustre, you are indirectly creating the *user.config*, *project.config*, and *<user>Context.config* files. However, aside from the initial system configuration in the *init.config* file that you must set before you begin working, there is no need for you to manually create or modify any of the other configuration files. All of the configuration settings are set within Lustre's user interface (refer to "System Settings" in the "Project Management" chapter of the *Autodesk Lustre User Guide*).

When you log into a particular project in Lustre for the first time, Lustre creates the *<user>Context.config* file, where *<user>* represents the user name associated with the user profile. The context file is found in the *.../projects/<project>/* directory. The context keywords include some attributes not defined in the Project or User configuration settings. They are saved and reloaded each time the context (same user and project) applies. The context settings include:

- Last scene
- Last grade
- Resolution
- Crop value
- Viewing Options
- Timeline settings and frame rate
- State of the surface control
- Audio file

The following is a list of the Lustre configuration files.

Configuration file	Default Location	Description
init.config	*/usr/autodesk/lustre2012/*	This file stores system settings as well as Wiretap servers, Slave Renderer settings, and film stock information.
login.config	*/usr/autodesk/lustre2012/*	This file records the last user and last project used so at the next session, the user and project login defaults will reflect these values.
project.config	*/usr/autodesk/projects/<project>/project.config*	This file stores project-level information including project settings, calibration, rendering, engineering, and Backburner and Wiretap settings specific to the project.
user.config	*/usr/autodesk/users/<user>/user.config*	This file stores user settings specific to a particular user, such as Autosave, printer light, GUI background/gain/gamma.

Configuration file	Default Location	Description
<user>Context.con-fig	*/usr/autodesk/projects/<project>/<user>Con-text.config*	This file stores several settings relevant to the context of a particular user working on a particular project. The purpose of storing context-specific information is that there are some parameter settings that, by their nature, are more likely to be needed in the next session, but that are not configurable in the Project Management pages.

System Settings

Before you first log into Lustre, you must configure the system settings in the *init.config* file (refer to "Configuring System Settings" in the "Project Management" chapter of the *Autodesk Lustre User Guide*). If these system settings are not configured, Lustre uses the default system settings. The following table describes the relevant system settings that should be configured before the first login.

NOTE All keyword values are case-sensitive.

Parameter	Data type	Default	Function
<Locations>			
<MainProjectHome>	String		Location of the Project configuration settings folder.
<MainUserHome>	String		Location of the User configuration settings folder.
<GlobalGradeBinHome>	String		Defines the location of the global Grade bin.

Parameter	Data type	Default	Function
<LutHome>	String	[LUSTRE_HOME]/lut	Defines the location of the LUTs. **NOTE** If you a re configuring a Slave Renderer system, and [LUSTRE_HOME] is located on a mounted network drive, make sure you use the full UNC path of the network folder instead of the mapped drive letter.
<PresetsHome>	String	[LUSTRE_HOME]/presets	Defines the location of the presets. **NOTE** If you a re configuring a Slave Renderer system, and [LUSTRE_HOME] is located on a mounted network drive, make sure you use the full UNC path of the network folder instead of the mapped drive letter.
<PluginsHome>	String	[LUSTRE_HOME]/plugins	Defines the location of the plug-ins. **NOTE** If you a re configuring a Slave Renderer system, and [LUSTRE_HOME] is located on a mounted network drive, make sure you use the full UNC path of the network folder instead of the mapped drive letter.
<TemplatesHome>	String	[LUSTRE_HOME]/templates	Defines the location of the templates. **NOTE** If you a re configuring a Slave Renderer system, and [LUSTRE_HOME] is located on a mounted network drive, make sure you use the full UNC path of the network folder instead of the mapped drive letter.

Parameter	Data type	Default	Function
<Help>			
<HelpWebBased state>	String	On	Defines the location of the Lustre Help that is opened by the Help button and shortcut. Lustre Help is web-based by default.
<HelpLocalPath>	String		Defines the default location for Lustre Help when it is installed locally.
<Browser>			
<HideFolder> (within <MediaBrowseHide-FoldersGroup>)	String	degrain_cache,original, matte,marry_grade, and 128x96	Hides the degrain cache, original, matte, marry_grade, and 128x96 folders from the file browser. **NOTE** You can hide any additional folder by adding the folder name to the list.
<Wiretap>			
<ServerAuto>	State	On	Specifies whether Lustre automatically scans for Wiretap servers.
<WiretapServer> (within <WiretapServer Group>)	String	0.0.0.0	The IP address or DNS host name for a specific Wiretap server. If one or more <WiretapServer> keywords are set to valid Wiretap server addresses, Lustre lists the content of these servers in the browser before the results of the auto scan. If <ServerAuto> is set to OFF, only the specified Wiretap servers are listed in the browser.

Parameter	Data type	Default	Function
<WiretapGatewayServer> (within <WiretapGate-wayServer Group>)	String	0.0.0.0	The IP address or DNS host name for a specific Wiretap Gateway server.
<PathTranslation> (within <PathTranslationT-able>)	String	0.0.0.0	Allows you to map hostnames to IP addresses, as well as to define path translation rules.
<NTSCImageAspectRatio>	Float	1.3333333333333	Applies the correct aspect ration to NTSC resolution footage rendered to the Wiretap server, but not im-ported from the Wiretap server.
<PALImageAsepectRatio>	Float	1.3333333333333	Applies the correct aspect ratio to PAL resolution footage rendered to the Wiretap server, but not impor-ted from the Wiretap server.

<MonitoringAndCalibration>

Parameter	Data type	Default	Function
<DisplayType>	Enumerated (abstract data)	LCD	Specifies the monitor display type (e.g., LCD or CRT).
<Calibration_Steps>	Integer	10	Specifies the number of monitor calibration steps performed.

<ControlSurface>

Parameter	Data type	Default	Function
<AutodeskPanels>	State	On	When ON, this keyword enables the Autodesk Control Surface (ACS). When OFF, Lustre enables the Tangent CP100 control surface (if applicable). You must also set the path for the control surface rules file in the Panel Setup File field (refer to "Sys-

Parameter	Data type	Default	Function
			tem & Menu Settings" in the "Project Management" chapter of the Autodesk Lustre User Guide).
\<PanelIDs\>		function="0" grading="0" navigation="0"	The ID numbers for the Function, Navigation, and Grading panels of the ACS. You must manually configure these keywords. If your control surface is a Tangent CP100, Lustre does not use these keywords.
\<PanelIPTags\>		function="101" grading="100" navigation="102"	

\<Browsed\>

Parameter	Data type	Default	Function
\<Port\>	Integer	1055	All computers on the *BrowseD* network must use the same port to communicate. **NOTE** For Linux over IP, set the port value to 1044.
\<Username\>	String	root	Administrative user on the *BrowseD* server.
\<Password\>	String	xxx	Password for the administrative user. **NOTE** To encrypt the password, set the attribute `toEncrypt` to *yes*. The next time the application starts, the password string is encrypted in the configuration file.

Parameter	Data type	Default	Function
<UseInfi>	State	Off	Switch to ON if the networking protocol to use with *BrowseD* is the InfiniBand.
<ReadCacheBuffer Num>	Integer	0	
<ReadCacheThread Num>	Integer	0	
<BrowsedServer> (within <BrowsedServer Group>)	String	0.0.0.0	Defines the IP address or DNS host name for a specific *BrowseD* server.

<SlaveRenderer>

Parameter	Data type	Default	Function
<HostName>	String		The IP address or DNS host name for a specific Slave Renderer machine.
<Timeout>	Integer	5000	Sets the timeout duration (in milliseconds) of the automatic Slave Renderer detection.

<Backburner>

Parameter	Data type	Default	Function
<Hostname>	String	localhost	The IP address or DNS host name for a specific Backburner Manager machine.

<Transcode>

Parameter	Data type	Default	Function
<Transcode>	String	localhost	The IP address or DNS host name for a specific WT Gateway that will be used to create and access the transcoded media.

<MatchCustomGroup>

Parameter	Data type	Default	Function
<MatchCustom>	Name	"AliasName" type="s"	Specifies the XML metadata field to be used by the Custom match option selected from the Browse menu. "s" represents string.
<MatchCustom>	Name	"DPXTimeCode" type="tc"	"tc" represents timecode.
<MatchCustom>	Name	"DPXkeycode" type="kk"	"kk" represents keycode.
<MatchCustom>	Name	"DPXTapeNAME" type="s"	"s" represents string.
<MatchCustom>	Name	"EDLReelName" type="s"	"s" represents string.
<MatchCustom>	Name	"DL_EDLClip_ name" type="s"	"s" represents string.
<MatchCustom>	Name	"DL_edlFrameID" type="i"	"i" represents integer.
<MatchCustom>	Name	"DLEDL_startTC" type="tc"	"tc" represents timecode.

<AVIO>

Parameter	Data type	Default	Function
<PlayoutHighSpeed>	State	Off	When OFF, it refreshes the Player as you are performing a playout. When ON, the Player is no longer refreshed and therefore the performance of the playout improves.
<WTTFirstSDLeadIn Correction>	Integer	1	Sets the delay for the SD first lead-in.

Parameter	Data type	Default	Function
<WTTSubsequentSD LeadInCorrection>	Integer	2	Delays all the lead-ins following the SD first shot.
<WTTSDLeadIn Increment>	Integer	0	Corrects the delay after the SD third shot.
<WTTFirstHDLeadIn Correction>	Integer	0	Sets the delay for the HD first lead-in.
<WTTSubsequentHD LeadInCorrection>	Integer	0	Delays all the lead-ins following the HD first shot.
<WTTHDLeadIn Increment>	Integer	0	Corrects the delay after the HD third shot.
<VtrTCTD>	Integer	10	The TimeCode Transition Delay specifies the number of milliseconds the application waits before asking for timecode from the VTR. The default value of 10 ms works for most decks. For HDCAM-SR decks use the following values: ■ 14, for 1080/59i/60 or 720/50/59/60 ■ 21 for other timings

<Miscellaneous>

Parameter	Data type	Default	Function
<DPXHeaderOffsetCheck>	State	On	When this keyword is enabled, Lustre reads the DPX / Cineon® file header of each frame in a sequence. This is useful when using DPX/Cineon files that have a different media offset within the same sequence of files. When disabled, Lustre reads only the first frame header of a sequence.

Parameter	Data type	Default	Function
			WARNING Turning this keyword on can alter playback performance.
\<DisablePanScan Frame>	State	On	
\<NVidiaSDISync>	State	Off	
\<NVidiaPlayoutDelay>	Integer	0	
\<BlockSize>	Integer	2048	
\<AudioResyncTime>	Float	-1	
\<LoadingSetup>	Integer	0	
\<EnableKeykode Sending>	Enumerated (abstract data)	Off	
\<LoginGUIGain>	Float	1.0	
\<LoginGUIGamma>	Float	1.0	
\<ClusterPlayDelay>	Integer	0	Available for Incinerator® only.
\<ClusterSmooth Playback>	State	Off	Available for Incinerator only.
\<ChangeCutOffset>	State	On	Allows the change cut or match grade feature to preserve the key-frame animations based on there-cord timecode whenever a shothas been replaced or moved withina timeline.

Parameter	Data type	Default	Function
			NOTE Be sure to enable only the Record button within the Match Option when performing a change cut or match grade.
<AnimCopy_UsingTrimOff-set_Off>	State	On	"On" enables the first animation keyframe to load on the first frame of the shot. "Off" enables the first keyframe to be loaded according to the source frame number (and thus load with the offset).
<PreallocEnable>	State	Off	Allows Burn rendering from a Lustre Windows workstation to use CXFS SAN preallocation.
<FirstFrameNumber>	Integer	On	
<StartShotFrameNumber>	Integer	On	
<DefaultDelivrablePanScan-Filter>	Enumerated (abstract data)	Fast	Possible values are: ■ **Fast** (Lanczos2 filter) ■ **Quality** (Lanczos2 filter) ■ **Custom** (BSpline filter)

<RenderNaming> See Render Naming Settings

Parameter	Data type	Default	Function
<NormalMode>	String		Normal render mode file path and token based naming convention.
<NoShotIdMode>	String		No Shot Id render mode file path and token based naming convention.

Parameter	Data type	Default	Function
<OneSeqMode>	String		One Sequence render mode file path and token based naming convention.
<SrcGradeMode>	String		Source Grade render mode file path and token based naming convention.
<StereoNormalMode>	String		Stereoscopic Normal render mode file path and token based naming convention.
<StereoNoShotIdMode>	String		Stereoscopic No Shot Id render mode file path and token based naming convention.
<StereoOneSeqMode>	String		Stereoscopic One Sequence render mode file path and token based naming convention.
<StereoSrcGradeMode>	String		Stereoscopic Source Grade render mode file path and token based naming convention.
<NormalPadding>	String		Defines the padding of the file name (the number of digits used for file numbering) for the Normal render mode.
<NoShotIdPadding>	String		Defines the padding of the file name (the number of digits used for file numbering) for the No Shot Id render mode.
<OneSeqPadding>	String		Defines the padding of the file name (the number of digits used

Parameter	Data type	Default	Function
			for file numbering) for the One Sequence render mode.
<SrcGradePadding>	String		Defines the padding of the file name (the number of digits used for file numbering) for the Source Grade render mode.
<StereoNormalPadding>	String		Defines the padding of the file name (the number of digits used for file numbering) for the Streroscopic Normal render mode.
<StereoNoShotIdPadding>	String		Defines the padding of the file name (the number of digits used for file numbering) for the Stereoscopic No Shot Id render mode.
<StereoOneSeqPadding>	String		Defines the padding of the file name (the number of digits used for file numbering) for the Stereoscopic One Sequence render mode.
<StereoSrcGradePadding>	String		Defines the padding of the file name (the number of digits used for file numbering) for the Stereoscopic Source Grade render mode.

<Debug>

Parameter	Data type	Default	Function
<DumpDLEDLPath>	String		Path to place the contents of a DLEDL when it is loaded in the timeline or Shot bin.
<DumpWiretapCreate Clip>	State	Off	When ON, the attributes of a Wiretap clip are placed into a log file each time a clip is created on a Wiretap server. The file *WiretapClip-*

Parameter	Data type	Default	Function
			Dump.log is located in the Lustre home folder. Use this keyword for debugging only.

<FilmTypeGroup>

Parameter	Data type	Default	Function
<FilmType>	Name		Various film types for AGFA, Kodak™, Eastman, and Fuji emulsions. It defines the relationship between the DPX film emulsion code and the DPX film code number.

Visual Effects and Finishing Command Line Start-up Options

Overview

There are a number of options you can use with the application start-up command under special circumstances.

NOTE More than one start-up option can be specified on the command line. These start-up options are case-sensitive.

To start the application using a command line option, open a terminal and type:

```
<application_name>-<option>
```

For example, **smoke -h**.

To see a list of all start-up options, use the **-h** option.

NOTE Running the application start-up command with the **-h** option does not launch the application.

Media Storage-Related Start-up Options

The following table lists the start-up options that enable you to initialize the storage, delete clips from a project, or troubleshoot missing frames.

Type:	To:
-v	Initialise the media storage volume (all projects). **WARNING** When you use the **-v** option, all data on the storage is deleted, with no possibility of recovery. **NOTE** Initialising the storage also removes all audio clips from the project.
-H	Specify the remote host name.
-V	Specify the media storage name.
-i	Remove all clips from the project.
-r	Initialise only the Desktop or Edit Desk for the selected project.
--red-frames	Activate repair mode. This option enables *vic* to search the media library and replace missing frame ids with a LOST frame label. In the library, the names of impacted clips appear red. When these clips are loaded into the record timeline, the names of the affected segments also appear red. Lost frames are displayed with the LOST label over them. The flag set on clips is not permanent, so resaving the clip will remove the red from the names.

Project Management-Related Start-up Options

The following table lists the start-up options that enable you to specify a custom application configuration file, as well as choose the default user and project the application starts with.

Type:	To:
-c`<filename.cfg>`	Use a configuration file other than the default, where `<filename.cfg>` is the name of the file. The specified file supersedes the default configuration file. If

Type:	To:
	you do not use this option, the application looks first for *init.cfg*, and if that is not found, it looks for *flame.cfg*, *inferno.cfg*, *smoke.cfg*, or *conform.cfg*.
-J<project>	Define the project that appears in the Project Management menu when you start the application, where <project> is the name of the project you want to use. If you start the application with both the **-J** and **-U** options, the application bypasses the Project Management menu.
-U<user>	Define the user that appears in the Project Management menu when you start the application, where <user> is the name of the user you want to use. If you start the application with both the **-J** and **-U** options, the application bypasses the Project Management menu.

Miscellaneous Start-up Options

The following table lists start-up options that enable you to install new fonts, handling log files, managing the amount of memory and number of CPU cores used by the application.

Type:	To:
-F	Force the application to install new fonts that you added to the */usr/lib/DPS/outline/base* directory (and the */usr/lib/DPS/AFM* directory, if you have also installed the corresponding font metric file). See the application help.
-B	Prevent the broadcast monitor from blanking when an image viewer or Player is not displayed. The broadcast monitor then displays the screen content that is displayed in the corresponding region. **NOTE** This option is hardware dependent and may not work on all hardware configurations.
-L	Disable writing to the *.log* file. The application keeps a log of your actions in a text file (*.log*) in your application home directory. This file is usually cleared and recreated every time the application starts. (Note that the log file is useful when reporting problems to Autodesk Customer Support.)

Type:	To:
-l	Append the current session to the existing log file. (See description of the log file above.)
-M	Define the amount of memory assigned to frame buffering. This option overrides the default settings, or if it has been uncommented, the value set by the MemoryApplication keyword in the *init.cfg* file.
-p`<n>`	On multi-processor machines, start the application using less than the full array of processors, where `<n>` is the number of processors to be used. Normally, the application uses all available processors.
-f`<filename>`	Use a custom menu file, where `<filename>` is the name of the menu file. For information on custom menus, see the application help.
-J`<pro-ject>`**-U**`<user>`**-b**`<batch_setup>`**-N**	Launch the application and specify a Batch setup to render. The application launches, renders the setup, and exits.
	NOTE Use the **-N** option only if you want to use Burn for rendering.
	WARNING To avoid corrupting the render, which uses the graphics hardware, you should not use the graphics console of the system (such as command shells) while the render is taking place.
-Z	Starts the application, and skips the integrity check.

Visual Effects and Finishing Event Triggers

Overview

You can set up your Visual Effects and Finishing software to execute custom external commands when certain events take place inside the application, for example, when the project or the video preview timing is changed by the user.

The */usr/discreet/*`<application_home>`*/python/hook.py* Python script contains a number of predefined functions that are called by the Visual Effects and Finishing application when certain events occur. These functions are referred

to as "hooks". You can enter Python commands directly inside these functions, or use the functions to call an external script or executable.

> **NOTE** To be able to run shell commands or external script using the `os.system` Python statement, add the following line in the *hook.py* script, before the definition of the first hook:

```
import os
```

Note the following about hook functions:

- Functions in *hook.py* receive certain predefined parameters from the application. These parameters contain specific details on the event that occurred in the application, for example, the name of the project loaded. Use these details to better customize the response action.

- Functions in *hook.py* are executed in a separate thread from the application. Thus, the application does not wait for a hook function to complete.

- When a function is called by the application, it executes with the privileges of the user account from which the application was launched. Make sure your custom code in the *hook.py* file does not call external scripts or executables that require more elevated user privileges.

- Functions do not return any value to the application that called them. Thus, the application cannot provide any feedback on whether your custom code executed properly or not.

- Actions defined in the *hook.py* file for one version of an application are not automatically used by other versions or applications. To carry your code over to another application or version, you must manually copy your customized *hook.py* file into the *python* subdirectory of that application's home directory.

This chapter describes the functions available in the hook API. More functions may be added by Autodesk in subsequent versions. Check the latest *hook.py* file after installing new versions of the software, to see any new functions that may have been added.

> **NOTE** You must be logged in as *root* to edit the *hook.py* file.

void projectChanged(string projectName)

This hook is called by the Visual Effects and Finishing application when the project is changed in the Project Management section of the Preferences menu. This hook is not triggered when a project is loaded at application start-up.

This function receives the following parameter from the application.

projectName	A string that specifies the name of the loaded project. For example, *My Project*.

Usage example:

```
def projectChanged(projectName):
  print projectName
```

When the project is changed in the application, this example function outputs the name of the project in the application terminal.

void previewWindowConfigChanged(string description, int width, int height, int bitDepth, string rateString, string synchString)

This hook is called by the Visual Effects and Finishing application when the video preview timing is changed in the software.

This function receives the following parameters from the application. The values of the parameters are extracted by the application from the `VideoPreviewDevice` keyword in the *init.cfg* software initialization configuration file. See VideoPreviewDevice for details on this keyword.

■ **description** A string that describes the video preview timing, as listed in the *init.cfg* file. For example, *1920x1080@5994i_free*.

■ **width** An integer that specifies the width of the preview timing. For example, *1920*.

■ **height** An integer that specifies the width of the preview timing. For example, *1080*.

■ **bitDepth** An integer that specifies the bit depth of the preview timing. For example, *12*.

■ **rateString** A string that specifies the refresh rate of the preview timing. For example, *6000i*.

■ **syncString** A string that specifies the sync source of the preview timing. For example, *freesync*.

Usage example:

```
def
previewWindowConfigChanged(description,width,height,bitDepth,rateString,syncString):

   print  os.system("/usr/bin/mySyncChangeScript
--sync="+syncString)
```

When the video preview timing is changed in the application, this example function calls a custom external script, `mySyncChangeScript`, and passes the sync value to it. That external script could use the sync value, for example, to trigger a sync change on the sync generator. The output of the external script is displayed in the application terminal.

Manually Configuring Media Storage

Overview

This section provides advanced instructions for manually configuring a storage array. Normally, you do not need to perform any of these tasks, as they are all automatically performed by the *XR Configuration Utility*.

Creating Hardware LUNs

LUNs, also referred to as Logical Units or Logical Drives, are groups of disk drives that are striped together to provide optimal performance and RAID protection. Once configured, LUNs are seen by the Linux operating system as if they were single disk drives.

To create the LUNs on your XR-series storage, use the *XR Configuration Utility* supplied by Autodesk. This utility is installed with the DKU and automates the LUN creation process.

NOTE For systems having two sets of enclosures, you have to configure one set at a time with the *XR Configuration Utility*. Connect the first set of enclosures, and use the utility to configure it. When done, disconnect the first set and connect the second set. When the second set of enclosures is configured, re-connect both sets.

To configure LUNs on XR-series storage:

   1   Open a terminal and log in as root.

2 Go to the directory */usr/discreet/DKU/current/Utils/Storage/current/*, and run the *XR Configuration Utility* by typing:

./XR_config.pl

The utility detects whether a LUN configuration exists on the storage attached to that workstation.

3 If a LUN configuration already exists on the storage, you are prompted for confirmation to overwrite that configuration.

WARNING LUN configuration is destructive. Make sure you want to overwrite an existing configuration before you confirm.

4 After the script detects the number of enclosures and drives, it prompts you to indicate the filesystem your storage uses.

Type 2.

5 When asked if you have a 2-loop or a 4-loop configuration, select the option that applies to your storage.

The *XR Configuration Utility* configures your storage.

6 Type x to exit the *XR Configuration Utility*.

7 Reboot your workstation, so that the newly-created LUNs are rescanned by the operating system.

Note that the *XR Configuration Utility* exits without configuring your storage if any of the following issues are detected:

■ An incorrect number of disks. The total number of disks must be a multiple of 12.

■ One or more of the enclosures do not have the correct firmware.

■ In a dual RAID enclosure environment, the number of expansion chassis on each RAID enclosure is not the same.

■ An odd number of enclosures in a 4-loop configuration. Only even numbers of enclosures are supported.

Partitioning Disks or LUN devices as Primary Partitions

To achieve optimal performance, each disk or LUN in the array should be partitioned as a single primary partition.

On storage arrays with 450 GB drives, use the *parted* utility to create GPT (GUID Partition Table) type partitions. On arrays with smaller drives, use the *fdisk* utility to create Linux LVM type partitions.

NOTE Make sure you do not re-partition the system drive or any other disks that you want to preserve as they are. Partitioning destroys all data on the disks.

To partition disk or LUN devices with 450 GB drives or larger:

1 Reboot your system to reload the fibre channel adapter drivers.

2 Open a terminal, and log in as root.

3 View a list of disks or LUN devices detected by the operating system, using the following command:

   fdisk -l | grep dev

   Identify the disk or LUN devices that are part of the storage array to be configured with a standard filesystem. These devices will be re-partitioned.

4 Use the *parted* command to re-partition each disk device identified in the previous step. Type:

   /sbin/parted -s -- <disk name> **mklabel gpt mkpart primary 0 -1**

   where <disk name> is the name of a disk device identified in step 1, without a partition number, such as */dev/sdb*. For example:

   /sbin/parted -s -- /dev/sdb mklabel gpt mkpart primary 0 -1

5 Repeat the command for each disk.

To partition disk or LUN devices with drives smaller than 450 GB:

1 Reboot your system to reload the fibre channel adapter drivers.

2 Open a terminal, and log in as root.

3 View a list of disks or LUN devices detected by the operating system, using the following command:

   fdisk -l | grep dev

   Identify the disk or LUN devices that are part of the storage array to be configured with a standard filesystem. These devices will be re-partitioned.

4 If you plan to configure a standard filesystem on a former Stone FS storage array, delete the volume label and volume table on each LUN device

that is part of the storage array. Type the following command for each LUN device:

dd if=/dev/zero of=`<LUN device>` **count=4096**

Where `<LUN device>` is the device name of a LUN in your storage array, such as */dev/sdc*.

WARNING When using the *dd* command, be very careful to not delete your system drive (usually */dev/sda*) or any other drive aside from the LUNs in your storage array.

5 Use the *fdisk* command to re-partition each disk device identified in the previous step.

Start the *fdisk* utility for the LUN. Type:

fdisk `<disk name>`

where `<disk name>` is a disk device name without a partition number, such as */dev/sdf*.

The *fdisk* utility starts, checks the disk device, and then displays its prompt.

NOTE When *fdisk* starts, a warning about the number of disk cylinders may appear. You can disregard this warning.

6 Type **n** to display the New partition creation menu.

fdisk displays the type of partitions you can create (primary or extended).

7 Create a primary partition on the disk device by typing **p** at the prompt.

8 When prompted to enter a partition number, type **1** to make the primary partition the first one on the LUN.

NOTE You may have to delete pre-existing partitions by entering **d** when prompted, and repeating step 3.

9 When prompted to set the starting cylinder number, press `Enter` twice to accept the defaults, which are the first and last cylinder on the device.

The *fdisk* prompt reappears.

10 Type **t** to set the partition type.

You are prompted to enter the hexadecimal code of the partition type to be created on the LUN.

11 Type **8e** to set the partition type to Linux LVM.

fdisk sets the partition as Linux LVM and the following output appears:

```
Changed system type of partition 1 to 8e (Linux LVM)
```

12 Type **w** to save the new partition table.

13 Repeat steps 2 through 9 for each disk or LUN device identified in step 1.

Assembling the Disk or LUN Devices into a Logical Volume

After you have formatted each disk or LUN device as a partition, you must assemble the LUNs into a single logical volume on which you create the XFS filesystem.

This procedure does not cover creating fault-tolerance and assumes that the LUNs are RAID-protected, as is the case with Stone Direct XR-series arrays.

To assemble a logical volume:

1 Verify that the disk or LUN devices are detected by the operating system. Type:

fdisk -l | grep dev

All devices appear in a list similar to the following example (your values may vary):

```
/dev/sdb1 1 180482 1449713663+ ee EFI GPT
/dev/sdc1 1 180482 1449713663+ ee EFI GPT
/dev/sdd1 1 180482 1449713663+ ee EFI GPT
/dev/sde1 1 180482 1449713663+ ee EFI GPT
```

Partitions created with the *parted* command for arrays with 450 GB disks are marked "EFI GPT".

Partitions created in *fdisk* for arrays with smaller capacity disks are marked "Linux LVM".

NOTE Other devices of different types may be listed before and after the GPT or LVM devices.

2 Create a physical volume on each of the devices using the following command:

pvcreate<list of devices>

where <list of devices> is a list of all the devices in the storage array. For example, if you have four devices, ranging from */dev/sdb1* to */dev/sde1*, you would type:

pvcreate /dev/sdb1 /dev/sdc1 /dev/sdd1 /dev/sde1

The physical volumes are created, and the command output should be similar to the following example:

```
Physical volume "/dev/sdb1" successfully created
Physical volume "/dev/sdc1" successfully created
Physical volume "/dev/sdd1" successfully created
Physical volume "/dev/sde1" successfully created
```

TIP You can use the command *pvremove* to delete any erroneously entered devices.

3   Verify that the physical volumes were initialized correctly. Type:

pvscan -v

A list of all of the physical volumes you created appears. The following sample output is for the previous example of 4 physical volumes created on devices */dev/sdb1* through */dev/sde1*:

```
Walking through all physical volumes
PV /dev/sdb1 lvm2 [2.03 TB / 2.03 TB free]
PV /dev/sdc1 lvm2 [2.03 TB / 2.03 TB free]
PV /dev/sdd1 lvm2 [2.03 TB / 2.03 TB free]
PV /dev/sde1 lvm2 [2.03 TB / 2.03 TB free]
Total: 4 [8.10 TB] / in use: 0 [0 ] / in no VG: 4 [8.10
TB]
```

4   Create the volume group "vg00" from the physical volumes you created in the preceding step, using the following command:

vgcreate vg00 `<list of volumes>`

where `<list of volumes>` is the list of physical volumes you created in the preceding step. For example:

vgcreate vg00 /dev/sdb1 /dev/sdc1 /dev/sdd1 /dev/sde1

TIP You can use the command *vgremove* to delete any erroneously entered volume.

5   Verify the volume was created and obtain the value of the "Free PE / Size" field. Type:

vgdisplay -v

In the output, find the line that contains the "Free PE / Size" field and write down the value of the "Free PE". For example, in the following example output the "Free PE" value is 2124556.

```
Free PE / Size 2124556 / 8.10 TB
```

6 Create a new logical volume on "vg00", using the following command:

lvcreate -l `<Free_PE_value>`**-i** `<#_of_physical_volumes>` **-I 32 -n lvol1 vg00**

where `<Free_PE_value>` is the "Free PE" value you noted in the preceding step and `<#_of_physical_volumes>` is the number of physical volumes. If we continue with the example used in the previous steps, you would type:

lvcreate -l 2124556 -i 4 -I 32 -n lvol1 vg00

The output confirms the creation of the logical volume:

```
Logical volume "lvol1" created
```

NOTE If the command outputs several lines about a file descriptor leaked on lvdisplay invocation, ignore them.

7 Check if the *adsk_lvm* startup script has been installed by the DKU to enable automatic logical volume reassembly upon reboot:

chkconfig --list | grep adsk_lvm

If the script is properly configured, the command output is the following:

```
adsk_lvm 0:off 1:off 2:on 3:on 4:on 5:on 6:off
```

If the command output is different, enable the script by typing:

chkconfig --add adsk_lvm
chkconfig adsk_lvm on

Creating the XFS Filesystem on the Logical Volume

After having created the logical volume, you are now ready to create and mount the XFS filesystem.

To create and mount an XFS filesystem:

1 Identify the optimal *agsize* value for your array by running the *mkfs.xfs* command. Type:

mkfs.xfs -d agcount=128 -f /dev/vg00/lvol1

This command displays diagnostics information similar to the following (your values may differ):

```
meta-data=/dev/vg00/lvol1  agcount=128, agsize=1066667
blks
          =                 sectsz=512 attr=0
data      =                 bsize=4096 blocks=134400000,
imaxpct=25
          =                 sunit=16   swidth=64 blks,
unwritten=1...
```

2 From the diagnostic information printed in the previous step, note the following values:

- ■ *agsize* on the first line

- ■ *sunit* and *swidth* on the fourth line

3 Depending on the values of *sunit* and *swidth*, calculate a new *agsize* value using one of the following three methods:

- ■ If the values of *sunit* and *swidth* are both equal to 0, multiply the *agsize* value by 4096. For example (your values will differ):

  1066667 * 4096 = 4369068032

  Proceed to step 4 using the value calculated above as the new *agsize* value.

- ■ If the command displays a warning message about the *agsize* being a multiple of the stripe width, multiply the *agsize* value by 4096, and subtract the *sunit* value multiplied by 4096. For example (your values will differ):

  1066667 * 4096 = 4369068032

  16 * 4096= 65536

  4369068032 - 65536 = 4369002496

  Proceed to step 4 using the value calculated above as the new *agsize* value.

- ■ If the values of *sunit* and *swidth* are not equal to 0, and no warning message appears, proceed to step 4 using the *agsize* value displayed by the *mkfs.xfs* command in step 1.

4 Run the *mkfs.xfs* command again to create the XFS filesystem on the device */dev/vg00/lvol1* using the value calculated in one of the previous steps. Type:

mkfs.xfs -d agsize=<new agsize> **-f /dev/vg00/lvol1**

For example (your values will vary):

mkfs.xfs -d agsize=4369068032 -f /dev/vg00/lvol1

The filesystem is created on the storage array.

NOTE If the command fails, redo your calculations starting from step 1.

5   Verify that the storage can be mounted by typing one of the following commands:

- For HP Z800 systems: **mount /mnt/StorageMedia**

- For older systems: **mount /mnt/stoneMedia**

The storage should mount, as the DKU installation script should have created the mount point directory for your storage (*/mnt/StorageMedia* on HP Z800 workstations, or */mnt/stoneMedia* on older workstations), as well as the corresponding entry in the */etc/fstab* file.

If you receive an error message and the storage does not mount, follow the instructions in the next section to manually mount the storage.

Manually Creating a Mount Point and Mounting the Storage

If the mount point directory for your storage was not created automatically by the DKU, or if the storage does not mount, perform the following procedure to create the mount point and mount the storage manually.

To create a mount point and mount the storage:

1   Create the directory that will serve as the mount point for the filesystem, if it does not exist. For example:

mkdir /mnt/StorageMedia

WARNING Do not use the word "stonefs" as the name for your mount point directory. "Stonefs" is a reserved word, and can cause issues if used as the mount point directory name.

2   Mount the XFS filesystem from the logical volume */dev/vg00/lvol1* on the directory you created in the previous step. For example:

mount -av -t xfs -o rw,noatime,inode64 /dev/vg00/lvol1 /mnt/StorageMedia

The filesystem is mounted as */mnt/StorageMedia*.

3   Confirm that the storage is now mounted. Type:

df -h

The output should list */dev/mapper/vg00-lvol1* mounted on your mount point directory. For example:

```
/dev/mapper/vg00-lvol1

 814G 547G 267G 68% /mnt/StorageMedia
```

4 Using a text editor, add an entry in the *etc/fstab* file so that the filesystem gets mounted automatically at startup. For example:

```
/dev/vg00/lvol1 /mnt/StorageMedia xfs rw,noatime,inode64
```

5 Optional: Confirm that the filesystem can mount automatically by rebooting the workstation and using the command *df -h* again.

Visual Effects and Finishing Applications Wiretap Path Translation

About Wiretap Path Translation

The Wiretap path translation service converts the syntax of a file path referred by a Wiretap server running on a Linux workstation to a format that the client can understand. This service allows Wiretap clients, who might be running Windows® or Mac® OS X, to understand file paths that are referenced by an Autodesk Creative Finishing product to clips that have been soft-imported from a NAS, SAN, or any remote mounted storage.

For example, Lustre running on a Windows workstation can work with an Autodesk Visual Effects and Finishing product's soft-imported clip on a SAN or NAS.

For the Windows workstation, the syntax of the path to the media files may resemble this:

```
 N:\myclips\clip1\frame1.dpx
```

On a Linux workstation, the path to the same media files may resemble this:

```
 /CXFS1/myclips/clip1/frame1.dpx
```

When the Wiretap client tries to locate this clip, the path translation service converts the syntax of the path so that the client can access the soft-imported clip from a remote storage device it has mounted locally.

Configuring the Path Translation Database

The path translation service uses a translation database. The path translation database contains a set of rules that specify how to convert (or "translate") a file path referred to by a source workstation (the host running the Wiretap server) into a path that the client can understand. These rules are contained in the */usr/discreet/wiretap/cfg/sw_wiretap_path_translation_db.xml* file on each Autodesk Creative Finshing workstation. This configuration file is read by the Wiretap server.

The translation database is an XML file that must be updated by hand as the network configurations change. The file should be created and maintained by a system administrator who is aware of the configuration of the network.

To configure the path translation database:

1 Consider your network environment and how you can organize hosts to simplify the path translation process.

Do you have many hosts accessing the same storage device? Do they mount the central storage using the same syntax and drive names? Are these hosts running the same application? Are they running the same operating system?

2 Open */usr/discreet/wiretap/cfg/sw_wiretap_path_translation_db.xml* in a text editor.

3 Create the rules for translating the file path. The rules are applied in the order that they appear in the file. The first rule is applied. If that rule matches, then the translation is complete. If the rule doesn't match, the next rule is examined.

If your network includes groups of hosts that mount their storage using the same syntax, you may be able to use a group or platform rule to translate the syntax for all similar hosts. If, however, all of your hosts use different syntax and mount points for their storage, you will have to create a host/path rule for each source and destination host.

See the sections that follow for a description of the possible path translation rules you can create.

4 Save the file.

5 You do not have to restart Stone and Wire to apply changes in the path translation database. The Wiretap server periodically checks the timestamp of the file and, if there are changes, it updates itself accordingly. However, if you want to test the results of your changes, you can restart Stone and Wire manually.

To restart Stone and Wire manually, type:

/etc/init.d/stone+wire restart

Stone and Wire restarts.

About XML Syntax

You must respect XML syntax standards when editing the path translation file. For example, the values assigned to attributes must not include ampersands (&) or left angle brackets (<) in their literal form. All instances of these characters, except left and right angle brackets used to delimit XML tags, should be replaced with & and < respectively. Do not concern yourself with the white-space between attributes; it is ignored when the file is parsed.

Creating a Host/Path Rule for Host-to-Host Translation

Create a host/path rule to translate the path syntax used by the source workstation (the workstation running the Wiretap server) to the path syntax used by the destination workstation.

The syntax of the host/path rule is as follows:

<map src_host="`<src_host>`**" src_path="**`<src_path>`**"**

dst_host="`<dst_host>`**" dst_path="**`<dst_path>`**">**

Enter a value for each attribute. See the following table for a description of the valid values for each attribute.

Attribute	Description
src_host	The host name or IP address of the workstation to which a client connects. If you create a group rule for source workstations, the value of this attribute can be the group name.
src_path	The path to the remote storage as referred to by the source host (or hosts, when mapping a group).
dst_host	The host name or IP address of the client that mounts the same storage referred to by the source, using a different syntax. If you create a group rule for destination workstations, the value of this attribute can be the group name.

Attribute	Description
dst_path	The path to the remote storage as referred to by the destination host (or hosts, when mapping a group).

Example:

<map src_host="flame1"   src_path="/CXFS1"

dst_host="windows1" dst_path="N:\">

Setting up a host-to-host translation for every machine in your network is inefficient. If you have many hosts that will use the same path translation rule, you can set up a group or platform rule. See the sections that follow.

Creating a Rule for Groups of Hosts

Use the group rule when you have several workstations that will use the same rule to convert the path syntax of a single source or group of source workstations. This rule is effective in larger installations where numerous workstations mount storage using identical syntax.

All hosts in a group rule must mount directories using exactly the same syntax.

NOTE Group names must be unique and must not conflict with host names or platform names.

The syntax of the group rule is as follows:

<group name="<group_name>**" os="**<os_name>**">**

[<host name="<host_name>**">]**

</group>

You must enter a value for each attribute. See the following table for a description of the valid values for each attribute.

Attribute	Description
group name	Identifies the name of the group. Create a group name of your choosing. Each group name must be unique. Use the value of this attribute in a host-to-host rule to map all members of the group to the same storage mount point.
host name	Identifies the name of a host that is in the group.

Attribute	Description
os	This attribute is optional. It allows you to link all of the hosts in the group to a single operating system. You can then use a platform rule to map all the hosts in a group to the same mount point. The os attribute is restricted to the following values: Linux, Windows NT, and MacOSX.

Example:

<group name="Autodesk_Stations" os="Linux">

  <host_name="flame1">

  <host_name="inferno1">

  <host_name="smoke1">

</group>

<group name="Lustre_Windows_Stations" os="WindowsNT">

  <host_name="lustre1">

  <host_name="lustre2">

  <host_name="lustre3">

</group>

After you have defined a group, you can use the host/path rule to map all the hosts in the group to the same path by using the group name for either the src_host or dst_host attribute value.

In the following example, all hosts in the Lustre_Windows_Stations group map the Autodesk_Stations storage path to N:\myclips, a mount point they all share:

<map src_host="Autodesk_Stations" src_path="/CXFS1/myclips"

  dst_host="Lustre_Windows_Stations" dst_path="N:\myclips">

You can also use a platform rule to map all the hosts in a group to the same destination path based on the operating system they all run.

In the following example, the platform rule is used to map all Windows users to N:\ when translating /usr/ from a Linux system:

<map src_os="Linux"   src_path="/usr/"

  dst_os="WindowsNT" dst_path="N:\">

Creating a Rule for a Group of Hosts Running on the Same Operating System

The platform rule is similar to the group rule in that the rule is applied to all hosts running the same operating system.

All hosts running the same operating system must mount directories using exactly the same syntax. For example, all Windows workstations must mount the NAS on the `N:\` mount point to use the same path translation rule for the NAS.

NOTE Platform names must be unique and must not conflict with host names or group names.

The syntax of the platform rule is as follows:

<map src_os=“`<src_os>`**” src_path=“**`<src_path>`**”**

    dst_os=“`<dst_os>`**” dst_path=“**`<dst_path>`**”>**

You must enter a value for each attribute. See the following table for a description of the valid values for each attribute.

NOTE The src_os and dst_os attributes are restricted to the following values: Linux, WindowsNT, and MacOSX.

Attribute	Description
src_os	The operating system of the workstation to which a client connects
src_path	The path to the remote storage as referred to by all hosts using the same operating system
dst_os	The operating system of the client that mounts the same storage referred to by the source
dst_path	The path to the remote storage as referred to by the hosts running the destination operating system

Example:

<map src_os=“Linux”   src_path=“/usr/”

    dst_os=“WindowsNT” dst_path=“N:\”>

Sharing the Database Among Multiple Wiretap Servers

When you have many Wiretap servers, you must update the path translation database on each server every time there is a change to the network configuration.

To simplify the management of the database, you can create a symbolic link from the path translation file on all hosts running the Wiretap server to one machine that contains the actual file.

Testing the Path Translation Database

After the path translation database is updated by the system administrator, it must be tested. The **sw_wiretap_translate_path** command verifies that a path requested from the Wiretap server on the local machine can be translated into a path that can be understood by the client machine.

The syntax of the command is the following:

sw_wiretap_translate_path [-h <host name>] [-p <path>] [-f <file containing paths>] [-H <destination host name>] [-O <destination OS>]

The command options are described in the following table.

Option	Description
-h	Specifies the host name or IP address of the Wiretap server. The default is `localhost`.
-p	Specifies the path on the Wiretap server host to translate.
-f	Specifies the file containing the paths on the remote host to translate to the path on the local host, delimited by new lines.
-H	Specifies the destination host name. The default is `localhost`.
-O	Specifies the destination operating system (Linux, Windows NT, Mac OSX).

NOTE Either -p or -f must be specified, but not both.

To test the path translation database:

1 Change to the Stone and Wire tools directory:

   cd /usr/discreet/sw/tools

2 Type the **sw_wiretap_translate_path** command with the appropriate options.

For example, suppose you have a SAN mounted as "/CXFS1" on your Flame workstation (flame1). A Windows PC (windows1) will mount the SAN as "N:\". The path translation file contains the following entries:

  <map src_host="flame1"   src_path="/CXFS1"

   dst_host="windows1" dst_path="N:\">

From the workstation, open a new terminal and type:

./sw_wiretap_translate_path -h flame1-p /CXFS1/myclips -H windows1

The result should return the value of the destination path, which represents how "windows1" will mount"/CXFS1" on "flame1".

```
N:\myclips
```

NOTE If the result returns the value of the source path, the translation database entry was not successfully applied.

Visual Effects and Finishing Standard Filesystem Maintenance

Introduction

Performance of a filesystem depends on a number of factors, including the following:
- Hardware and operating system
- Fragmentation
- I/O sizes
- Filesystem tuning parameters
- Free space available

- Concurrent I/O and metadata operations
- Number of I/O threads and processes in use at a given time

NOTE Standard filesystems tuned for a specific file format can under-perform for other file formats. For instance, a filesystem optimized and capable of delivering real-time performance for 2K 10-bit DPX files delivers lower real-time performance for SD media.

Filesystem performance progressively decreases over time, depending on various factors, such as fragmentation, fill rate, and concurrent use by several applications.

This chapter provides basic usage and maintenance guidelines to help you keep the performance of your standard filesystem at a steady level.

Terminology and Concepts

Before reading this chapter, familiarize yourself with the following concepts.

Fragmentation   Fragmentation is a phenomenon that consists of a file being broken into several blocks, which are stored in non-contiguous sectors on the disk. This decreases filesystem performance, as it takes a lot of time to retrieve and reassemble the scattered pieces of the file before making it usable.

Locality of reference   This principle states that files used in the context of the same application or procedure (for example, consecutive frames in a timeline) should be as close to each other as possible on the disk, in order to minimize seek times between two successive I/O operations. Locality of reference is good in the case of frames written on an empty partition, and decreases as the filesystem fills up and becomes fragmented.

Multi-threaded I/O   Multi-threaded I/O is the use of multiple parallel I/O threads within a disk read process. This method allows the RAID array to optimize the fetching of multiple blocks of data.

Controlling Fragmentation

Filesystem fragmentation is directly related to the amount of mixing and interleaving of blocks of data of different sizes, and is aggravated by multiple I/O clients concurrently writing data to the partition.

Although some standard filesystems (such as XFS) feature advanced space allocation techniques, which make them more resistant to fragmentation, this problem still occurs as the filesystem's free space decreases and frequent file deletion, creation, and resizing operations take place.

Here are some steps you can take to counter filesystem fragmentation:

- Avoid storing mixed resolution projects on a standard filesystem partition. Workflows where many different media formats are used simultaneously tend to become fragmented more rapidly than if a single resolution media format is used consistently.

- Separate your media and metadata partitions. The worst contributors to fragmentation on a media storage device are the files used to store metadata. These include clip libraries, setup files, and databases. It is recommended to keep all metadata on a separate partition.

  NOTE You may keep the audio data on the same partition as the frames. This has no significant impact on performance or filesystem fragmentation.

- Use the defragmentation tools provided by your filesystem vendor on a regular basis.

  NOTE Defragmentation tools are often unable to realign files based on their locality of reference. Therefore, it is advisable to reinitialise your partition periodically, by archiving your projects, formatting the partition, and restoring the archives.

Limiting Concurrent Usage

A filesystem performs best when only one process is accessing it. In some workflows this is not always possible, which leads to a temporary decrease in performance. Furthermore, mixing read and write operations can have a detrimental effect on throughput.

Autodesk strongly advises against allowing any third-party application to access the storage at the same time as your Visual Effects and Finishing applications. The I/O operations from such applications may drastically

decrease the bandwidth available to your Visual Effects and Finishing application and also contribute to filesystem fragmentation.

NOTE Unlike direct attached storage (DAS), a Storage Area Network (SAN) is typically designed to handle concurrent usage and will perform better under such situations.

Configuring Bandwidth Reservation

Stone and Wire provides a mechanism to reserve storage bandwidth for Visual Effects and Finishing applications and tools that use a local mount point to a standard filesystem (DAS or SAN) as their media storage. This ensures that the local application gets the bandwidth it requires and that real-time playback on the local system is not jeopardized by requests from concurrent processes, including access from remote hosts.

NOTE Bandwidth reservation policies apply only to I/O requests from Visual Effects and Finishing applications and tools. They cannot protect your storage bandwidth from I/O requests coming from third-party processes or user interactions. It is your responsibility to avoid using third-party tools with the frame storage. See Limiting Concurrent Usage (page 202).

Bandwidth requests are managed by the Stone and Wire Bandwidth Manager, based on the parameters in the */usr/discreet/sw/cfg/sw_bwmgr.cfg* configuration file.

When Stone and Wire starts up, the Bandwidth Manager automatically creates a `[Device]` section in the configuration file for each partition declared in */usr/discreet/sw/cfg/stone+wire.cfg*. The Manager sets total theoretical read and write bandwidth values for each device, as well as the default reservation values.

Adding a New Device to the Bandwidth Manager

Use the instructions in this section if you want additional standard filesystem mount points (such as mount points to SAN volumes or to any other standard filesystem volume that is not present in the */usr/discreet/sw/cfg/stone+wire.cfg* file) to be managed by the Bandwidth Manager. You may also use these instructions as guidelines for understanding and editing the settings in the predefined `[Device]` sections of the configuration file.

Each partition managed by the Bandwidth Manager must have a `[Device]` section associated to it in the */usr/discreet/sw/cfg/sw_bwmgr.cfg* file.

To add a new device to the Bandwidth Manager:

1 Open the */usr/discreet/sw/cfg/sw_bwmgr.cfg* file in a text editor.

2 Add a new `[Device]` section at the end of the file, using the following syntax:

   TIP You may also copy and paste an existing `[Device]` section, and change its parameters.

```
[Device<device_number>]
Path<path_number>=<device_mount_point>
TotalAvailableReadBandwidth=<total_read_bandwidth>
TotalAvailableWriteBandwidth=<total_write_bandwidth>
DefaultReservation=<default_reservation>
```
   where:

   ■ `<device_number>` represents the number of the `[Device]` section, starting at 0. For example:

   `[Device0]`

   ■ `Path` specifies the partition's mount point. Since a partition can have several paths, `<path_number>` represents the number of the current path, starting at 0 for each device. For example:

   `Path0=/mnt/XYZ`

   `Path1=/usr/local/ABC`

   ■ `TotalAvailableReadBandwidth` represents the estimated total reading bandwidth of the device, in megabytes per second. Based on this value, and on the amount of active reservations for the current partition, the Bandwidth Manager decides whether new reservation requests are granted or rejected. For example:

   `TotalAvailableReadBandwidth=200`

   ■ `TotalAvailableWriteBandwidth` represents the maximum estimated writing bandwidth of the device, in megabytes per second. If this parameter is not specified, its value defaults to the value of `TotalAvailableReadBandwidth`. For example:

   `TotalAvailableWriteBandwidth=150`

NOTE The total bandwidth parameters are estimates of the theoretical maximum bandwidth of the partition. The actual bandwidth is affected by several factors, including multiple applications trying to concurrently read or write to it. The Bandwidth Manager continuously measures partition performance and dynamically maintains the actual total available bandwidth for each partition.

■ `DefaultReservation` sets the bandwidth assigned by default to applications that are denied a reservation request or that don't have individual or group reservations associated to them. The value is expressed in megabytes per second. For example:

   `DefaultReservation=10`

3 After setting up all the partition parameters described above, you can start defining reservations for the current partition. See Reserving Bandwidth for an Application (page 205).

4 Save and close the configuration file, and restart Stone and Wire by typing:

   /etc/init.d/stone+wire restart

Reserving Bandwidth for an Application

You may define a bandwidth reservation for each Visual Effects and Finishing application or tool that uses a certain partition. By default, a group reservation for the major Visual Effects and Finishing applications (Inferno®, Flame®, Flint®, Smoke®, and Backdraft® Conform) running locally is already defined.

Remote applications are identified by application-specific tokens, workstation host name, as well as the user name under which the application is executed. You can use any or all of these parameters to create detailed bandwidth reservation rules.

You can also customize the way bandwidth is redistributed in low bandwidth situations. In low bandwidth situations, instead of reducing all reservations proportionally towards zero, the bandwidth manager first reduces the bandwidth of each process towards the low-bandwidth value specified for that process, and attempts to maintain that minimum for as long as possible. If the device bandwidth keeps degrading, then the bandwidth manager starts reducing bandwidth proportionally from the low-bandwidth values towards zero.

In the following example, low-bandwidth values are not specified in the Bandwidth Manager configuration file. The diagram illustrates the way the Bandwidth Manger redistributes device bandwidth in such a case, as the total available bandwidth decreases from 800 MB/s to 420 MB/s and then to 320 MB/s. Note how bandwidth is proportionally reduced for all processes.
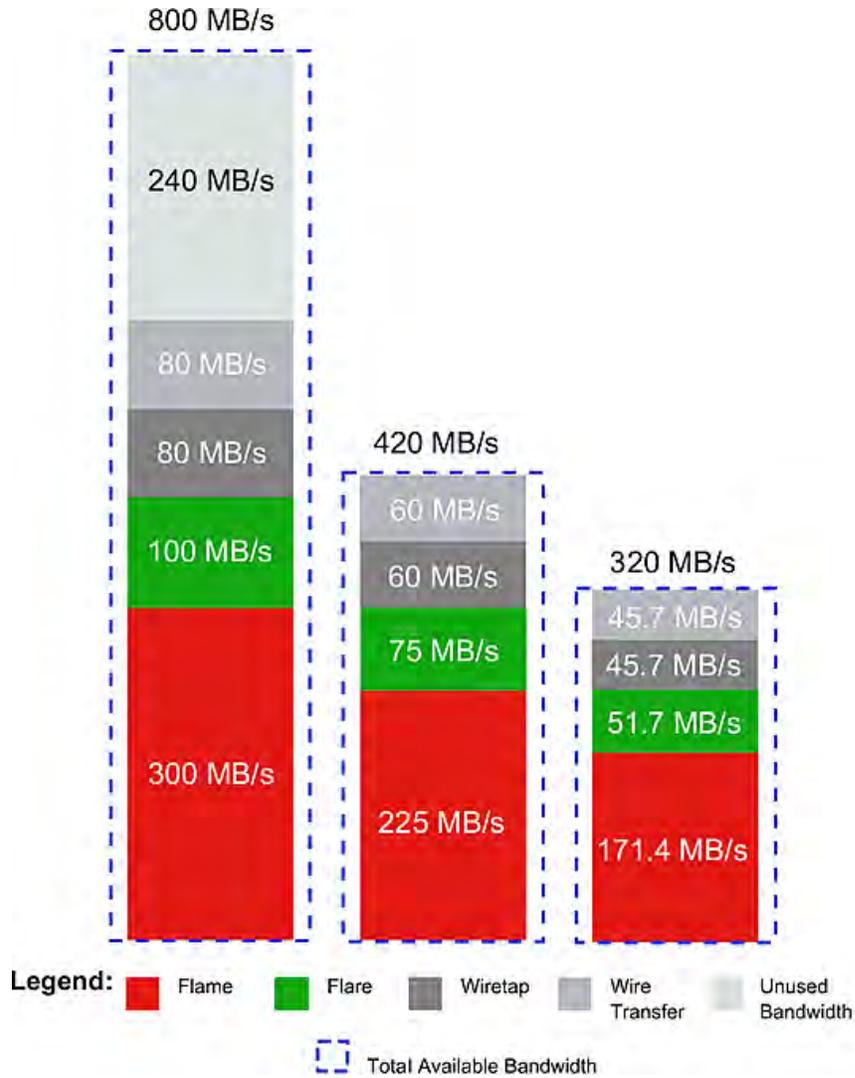
In the following example, low-bandwidth values are configured for each process (300 MB/s for Flame, 100 MB/s for Flare, 10 MB/s for Wiretap and 10

MB/s for Wire). The diagram illustrates the way the Bandwidth Manger redistributes device bandwidth as the total available bandwidth decreases from 800 MB/s to 420 MB/s and then to 320 MB/s.

Note how the Bandwidth Manager keeps the bandwidth for each application at the low bandwidth watermark. When total available device bandwidth falls under the sum of the low bandwidth watermarks, the Bandwidth Manager decreases bandwidth for all processes proportionally towards zero.
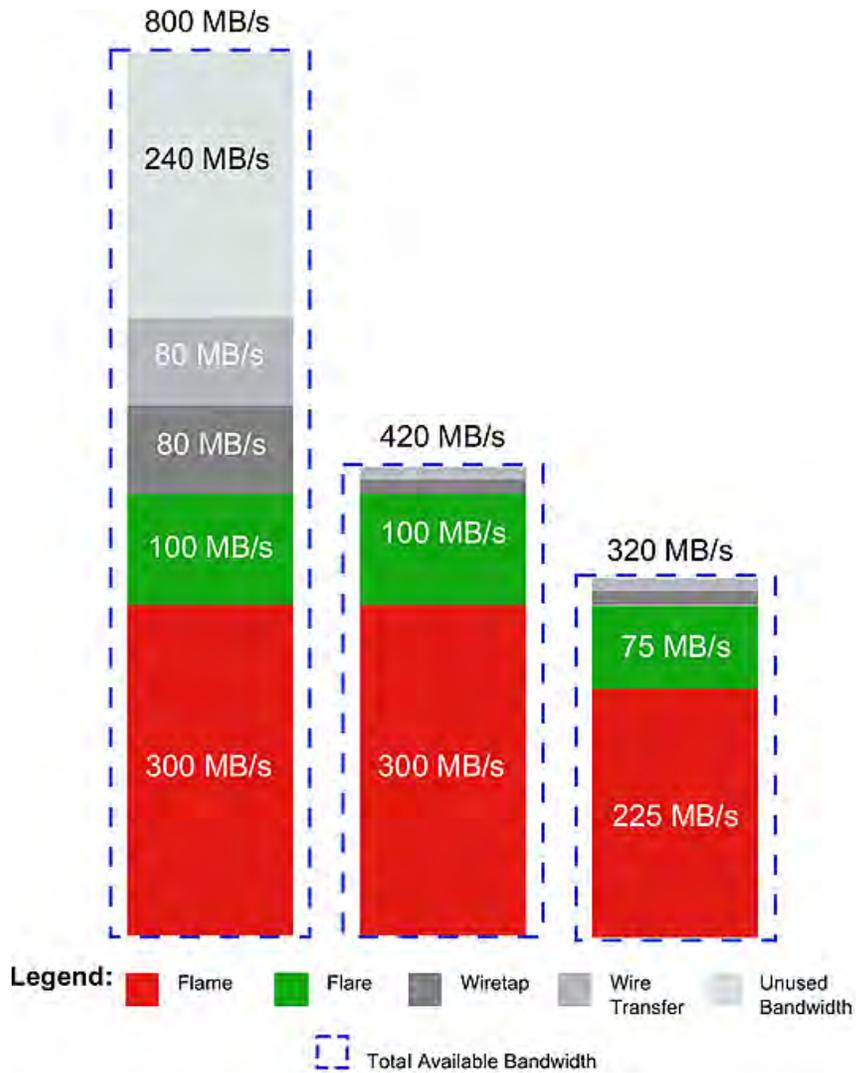
Perform the steps in the procedure below to set up an optimal bandwidth reservation for the local application, as well as for remote workstations, based on your system configuration.

To set up bandwidth reservation:

1 Open a terminal and log in as root.

2 Open the */usr/discreet/sw/cfg/sw_bwmgr.cfg* file in a text editor.

3 Locate the `[Device]` section that corresponds to the standard filesystem partition (by default `[Device0]`), and uncomment it if it is commented out.

4 Add a `Reservation` line for each local or remote application, using the following syntax:

```
Reservation<number>=<application_name> [<user_name>]
[@<workstation_hostname>] <reading_bandwidth>
[(<low_reading_bandwidth>)]
[<writing_bandwidth>][()<low_writing_bandwidth>)]
```

where:

■ `<n>` is the ID of the reservation, starting at 1 for each device.

■ `<application_name>` represents the name of the application that needs the reserved bandwidth. This parameter can take one of the following predefined values: "inferno", "flame", "flint", "flare", "smoke", "imcopy", "stonifiseTool", "publishTool", "S+W Server", "IFFFS Wiretap Server", or "IFFFSTool" for other Visual Effects and Finishing command-line tools. The "smoke" token also includes Backdraft Conform.

   NOTE Application names must use quotes if they contain spaces.

■ `<user_name>` is the Linux user account under which the remote application is running. This parameter is optional.

■ `<workstation_hostname>` is the hostname of the remote workstation that is accesing the storage. This parameter is optional.

■ `<reading_bandwidth>` represents the minimum reading bandwidth required by the application, expressed in megabytes per second. If more bandwidth is available on the device, the Bandwidth Manager gives the application as much bandwidth as possible.

■ `<low_reading_bandwidth>` represents the minimum value towards which reading bandwidth for this application is reduced in low-bandwidth situations. The Bandwidth Manager will attempt to maintain this minimum bandwidth allocation for as long as possible.

If the device bandwidth keeps degrading, then the bandwidth manager starts reducing bandwidth proportionally from the low-bandwidth values towards zero for all applications.

■ `<writing_bandwidth>` represents the minimum writing bandwidth required by the application, expressed in megabytes per second. If more bandwidth is available on the device, the Bandwidth Manager gives the application as much bandwidth as possible. If this parameter is not specified, the Bandwidth Manager automatically calculates a default value, based on `<reading_bandwidth>` and on the total configured read and write bandwidth values of the device.

■ `<low_writing_bandwidth>` represents the minimum value towards which writing bandwidth for this application is reduced in low-bandwidth situations. The Bandwidth Manager will attempt to give the application this minimum amount of bandwidth for as long as possible. If the device bandwidth keeps degrading, then the bandwidth manager starts reducing bandwidth proportionally from the low-bandwidth values towards zero for all applications.

For example:

```
Reservation1=flame 500 (300)
Reservation2=flare artist@flare1 200 (100)
```

5 Save and close the configuration file, and restart Stone and Wire by typing:

/etc/init.d/stone+wire restart

NOTE Bandwidth for an application is actually reserved only when that application is running. Processes that do not have a bandwidth reservation setting in the *sw_bwmgr.cfg* file fall under the default combined reservation of 10 MB/s.

Defining Groups of Applications

The Stone and Wire Bandwidth Manager allows you to set up bandwidth reservations for custom groups of applications. All the applications in a group will share the same reservation.

By default, the configuration file already contains a definition for the "IFFFS" group, which includes the "inferno", "flame", "flint", and "smoke" application tokens.

To set up bandwidth reservation for a group of applications:

1 Open the */usr/discreet/sw/cfg/sw_bwmgr.cfg* file in a text editor.

2 In the `[Groups]` section, add a line for each group of applications you want to define. The syntax of the line is as follows:

   `<group_name>=<application_name> <application_name>`

   where:

   ■ `<group_name>` is the custom name of the group. The group name must not contain spaces and must not be the same as one of the predefined application names.

   ■ Each `<application_name>` represents the name of an application in the group. This parameter can take one of the following predefined values: "inferno", "flame", "flint", "smoke", "imcopy", "stonifiseTool", "publishTool", "S+W Server", "IFFFS Wiretap Server", or "IFFFSTool" for other Visual Effects and Finishing command-line tools.

   For example:

   `[Groups]`

   `IFFFSTools=imcopy stonifiseTool publishTool`

3 After you have defined the group, go to the `[Device]`section for the appropriate partition and add a `Reservation` line for the new group. The syntax is the same as when adding a reservation for a single application. For example (using the "IFFFSTools" group from the previous step):

   `Reservation1=IFFFSTools 50 50`

4 Save and close the configuration file, and restart Stone and Wire by typing:

   /etc/init.d/stone+wire restart

For additional information on all the parameters in the Bandwidth Manager configuration file, and for a list of application names you can use when defining reservations, see the comments inside the configuration file.

Monitoring Partition Fill Rate

As each disk in a storage array gets filled near capacity, its overall read and write performance decreases once a certain threshold is reached.

Archive and delete projects as often as possible to free up space on your disk array. Alternatively, you may use the previously described steps to reinitialise your partition, making sure that you restore from archive only the projects you need to work on.

NOTE For an XFS filesystem on XR-series disk arrays, keep the partition fill rate below 90% for best performance.

Moreover, the Stone and Wire database may experience corruption if there is too little available space on the disk where clip metadata is stored.

To avoid this problem, you can set the Stone and Wire database to automatically shut off when free space falls below a certain threshold.

To configure the Stone and Wire database to automatically shut off when there is too little free space:

1 As root, open the */usr/discreet/sw/cfg/sw_dbd.cfg* file in a text editor.

2 Scroll to the `[Disk Check]` section.

3 Uncomment the `Enabled` and `Minimum` keywords.

4 Set the value of the `Minimum` keyword to the minimum allowed amount of free space on the disk (in Megabytes).

   If the amount of free space on the disk falls below this threshold, the Stone and Wire database daemon automatically shuts off.

5 Optional: Uncomment the `Timeout` keyword, and set how often the Stone and Wire daemon should check the available disk space. The default value is 30 seconds.

6 Save and close the configuration file.

7 Restart Stone and Wire. Type:

   /etc/init.d/stone+wire restart

Using Multi-Threaded Direct I/O

Most filesystems perform best when the I/O is parallelised across multiple threads/processes and sent asynchronously. This allows the device to buffer I/O operations and reorganize requests for optimal performance.

Some applications perform better than others on the same storage device, based on how they perform their I/O. Applications that use single-threaded buffered I/O can be slow.

Your Visual Effects and Finishing applications use multi-threaded asynchronous direct I/O when capturing, playing back, and performing other I/O intensive operations.

Make sure that any third-party application you use in conjunction with your Visual Effects and Finishing application or to handle your media files uses direct I/O. This prevents needless CPU-intensive in-memory data copying, thus maintaining optimum storage performance and freeing resources for your Visual Effects and Finishing application.

Enabling Media Pre-allocation

By default, disk space for video files is allocated on-the-fly as the files are rendered or written. Concurrent I/O operations from multiple clients or background processes can generate frames with poor locality of reference, which leads to clip fragmentation and decreases playback performance.

To prevent this problem, Stone and Wire provides media file pre-allocation mechanisms that reserve the necessary space on the disk before the actual writing of the media files takes place. Pre-allocation reserves as many available contiguous disk sectors as needed for the file sequence to be written.

Depending on the filesystem you are using, you may choose between two media pre-allocation methods.

FALLOCATE The disk space needed for the file is allocated and filled with zeroes. On most filesystems, the performance impact of this operation is as serious as when writing the actual file.

RESERVE The disk space needed for the file is reserved, but not written to. The apparent size of the frame remains zero. This operation is much faster than FALLOCATE, but is supported only on certain filesystems, such as XFS.

By default, pre-allocation is turned off in Stone and Wire (the method is set to NONE). In order to enable it, you must modify a parameter in the *stone+wire.cfg* configuration file.

NOTE This is not a global setting. Pre-allocation must be individually enabled for each partition.

To enable media pre-allocation for a partition:

   1 Open the*/usr/discreet/sw/cfg/stone+wire.cfg* file in a text editor.

2 Uncomment the `Preallocation` parameter in the section that configures the respective partition (for example `[Partition7]`).

3 Change the parameter value from the default `NONE` to the pre-allocation method you want to use. For example:

Preallocation=RESERVE

4 Save the configuration file and restart Stone and Wire.

NOTE If you are using XFS and unwritten extent flagging was disabled when creating the filesystem, the RESERVE pre-allocation method is not available for it. For more information about unwritten extent flagging, consult the Man page for *mkfs.xfs*. Type **man mkfs.xfs** in a terminal.

Testing Filesystem Performance

Each standard filesystem comes with its own set of tools to measure performance.

XFS comes with the *xfs_db* command line tool for troubleshooting various aspects of the filesystem, including fragmentation. For information on using the *xfs_db* tool, consult the man page for *xfs_db*.

Stone and Wire comes with a command line tool to measure filesystem performance called *sw_io_perf_tool*. This tool simulates I/O requests (audio and/or video) that would be issued by the Visual Effects and Finishing application, and reports the throughput and maximum latency of all the requested I/O operations.

Use *sw_io_perf_tool* to measure read and write performance of a volume declared and mounted by Stone and Wire. Several options are available to simulate many file formats and writing/reading patterns. Launching the command will display a list of available options.

NOTE To list all options available for *sw_io_perf_tool*, open a terminal and type **/usr/discreet/sw/tools/sw_io_perf_tool -h**.

To test a standard filesystem partition's performance for a specific resolution:

1 Go to the directory */usr/discreet/sw/tools* and type:

./sw_io_perf_tool `<-p #>` `<options>`

Where `<-p #>` specifies a partition ID (0 to 7), and`<options>` can be a combination of several media format attributes.

For example, to test read performances for 10-bit HD on the partition7:

./sw_io_perf_tool -p 7 -v HDTV -d 10

You may also use *sw_io_perf_tool* with custom resolutions. For example, to test read performances for frames with a width of 1600 pixels and a height of 900 pixels:

./sw_io_perf_tool -p7 -v 1600,900

The command displays the throughput (in MB/sec) and maximum IO latency detected (in ms), as shown below:

```
Running Stone+Wire IO Performance Test (Partition 7)

Video: 1 stream(s) of HDTV @ 10bit

Proxy: None

Audio: None

This test will take approximately 11 seconds

*** Read Test ***

Bandwidth: 275.456 MB/s

Frame Rate: 39.351 Frames/s

Max Latency: 335.284 ms
```

Notes:

■ The command can be run while other operations are ongoing to simulate how the application will behave under these conditions.

■ The media used by *sw_io_perf_tool* is allocated and freed automatically.

■ *sw_io_perf_tool* will not reserve any bandwidth on the storage device.

■ Partitions/Volumes must be properly configured in the Stone and Wire configuration file.

Visual Effects and Finishing Troubleshooting

General Application Troubleshooting Procedures

Forcing an Application to Exit

If a Visual Effects and Finishing application hangs, but you can still interact with the operating system, you can terminate it using either a custom command alias for your application, or the generic Linux **kill** command in conjunction with the application's process ID.

WARNING These procedures should only be used if the application is hung or otherwise not responding. Killing processes indiscriminately can cause system problems. Under normal conditions, you should use the normal exit procedures described in the application help.

Killing All Application-Related Processes

When you install a Visual Effects and Finishing application, a special command alias is defined that can be used to kill all Visual Effects and Finishing application processes at once.

To kill all application-related processes:

1 Open a terminal, and type one of the following commands:

  kf to kill Flame, Flint or Flare

  ks to kill Smoke

  ki to kill Inferno

  kc to kill Backdraft Conform

2 To verify that all application-related processes have been killed, type:

  ps -fu<application account name>

  This command displays information about all active processes, including the application with which each process is associated, and a process ID number. If all application processes have been successfully killed, this command should return no results.

WARNING Never restart an application if there are still processes left over from a previous session. If some processes persist after using the kill alias, perform the following procedure to kill them individually.

Killing Processes Individually

Use the **kill** Linux command to kill processes individually, using their process ID number.

To kill processes individually:

1 Press Alt+F3 to send the application to the background.

2 Open a terminal and type:

ps -fu`<application account name>`

This command displays information about all active processes, including the application with which each process is associated, and a process ID number.

3 Determine the process ID number of the Visual Effects and Finishing application.

4 At the command line, type:

kill`<process number>`

where `<process number>` is the process number you determined in the previous step. This command terminates the Visual Effects and Finishing process that is currently executing.

NOTE There may be more than one Visual Effects and Finishing process running at any time. For example, there may be one process per CPU, plus some additional processes to manage the framestore. Kill each of these processes.

5 Restart the application.

Troubleshooting Filesystem Problems

This section describes some common filesystem problems and steps you can take to solve them.

When troubleshooting storage or wire issues, you should always start by verifying that Stone and Wire processes are running properly, and by checking the log files.

Verifying that Stone and Wire Processes Are Running

There are five processes that must be running for Stone and Wire to work:
- *sw_serverd*
- *sw_probed*
- *sw_dbd*
- *sw_bwmgr*
- *ifffsWiretapServer*. This process is started by the *sw_wiretapd* script.

To verify that Stone and Wire processes are running:

1 Type the following command:

   ps -ef | grep sw_

   This command should return several lines similar to the following:

   ```
   root 9447 1 0 Feb25 0:00:35 /usr/discreet/sw/sw_probed
   --daemonize
   root 9455 1 0 Feb25 0:00:35 /usr/discreet/sw/sw_dbd -d
   root 9463 1 0 Feb25 0:08:11 /usr/discreet/sw/sw_bwmgr -d
   root 9472 1 0 Feb25 0:08:11 /usr/discreet/sw/sw_serverd
   -d
   root 9513 1 0 Feb25 0:00:00 /bin/sh
   /usr/discreet/sw/sw_wiretapd
   ```

2 To verify that the *ifffsWiretapServer* process is running, type:

   ps -ef | grep Wire

   The command should return a line similar to the following:

   ```
   root 9522 9513 0 Feb25 00:03:29
   /usr/discreet/wiretap/2010/ifffsWiretapServer -c
   /usr/discreet/wiretap/cfg/wiretapd.cfg
   ```

3 If Stone and Wire processes are not running, start them.

Starting Stone and Wire Processes

All Stone and Wire processes should start when you restart your system. If they are not running, check that they are configured to run and then restart Stone and Wire. To start Stone and Wire processes, you must modify a text file listing all processes.

To start Stone and Wire processes:

1   Use a text editor to open the Stone and Wire configuration file */etc/sysconfig/stone+wire*.

    The file should include the following:

    ```
    dl_stone=on
    ```

    ```
    dl_wire=on
    ```

    ```
    dl_wiretap=on
    ```

    ```
    dl_vic=on
    ```

    If any of the processes you need are set to "off", change them to "on" and then restart Stone and Wire.

2   To restart Stone and Wire, type:

    /usr/discreet/sw/sw_restart

    If any Stone and Wire process is still not running, check the error logs.

Checking Stone and Wire Error Logs

Every Stone and Wire process has a log file. You can find these log files at */usr/discreet/sw/log*.

The current log file is named `<process>`.log, where `<process>` is the name of the Stone and Wire process or daemon. The next time Stone and Wire creates a log file for the process, it renames the previous log file by adding a number to the file name.

For example, the *sw_served* process log file is named *sw_served.log*. The next time the process is launched, the first log file is renamed to *sw_served.log.1*. Each time Stone and Wire creates a new log file, the old log file is renamed by incrementing the number that appears at the end of the filename.

You can also use the terminal to list the most recent version of the log files. Type the following in a terminal:

ls -altr

Enabling Verbose Stone and Wire Driver Error Reporting

Use the *swr_set_debug* utility to enable verbose reporting of Stone and Wire driver errors.

When you turn on error reporting, Stone and Wire outputs error messages in the following locations:

■ */var/log/messages*

■ Console window

NOTE You must disable verbose error reporting after you have finished troubleshooting. The messages generated by this process can fill up the system drive.

To enable verbose Stone and Wire driver error reporting:

  1 Run *swr_set_debug*. Type:

    /usr/discreet/sw/tools/swr_set_debug`<level>`

    where `<level>` is the number representing the verbosity of the report.

    Choose from one of the following levels:

    ■ **1** - Turns on error reporting. This is the default level.

    ■ **2** - Provides more detailed reporting. All operations and errors are logged.

    ■ **0** - Disables error reporting.

  2 Check the terminal and the log files for error information.

Troubleshooting Missing Frames

If the application cannot find some of the frames referenced by your clips, the following error message is displayed when the application starts:

■ In the terminal:

```
WARNING: <nnnn> <volume_type> frames are missing on this
volume; see the application installation guide for
corrective measures.
```

Where <nnnn> represents the number of missing frames, and
<volume_type> can be stonefs or standardfs.

■ In the application start-up screen:

```
VOLUMEMGT: WARNING: Some frames not found;see the
application installation guide for corrective measures.
```

NOTE By default, the message only appears in the terminal. To have it appear
also in the start-up screen, open the ~/.cshrc file in a text editor and add the
following on a new line: setenv **DL_IC_NOTIFY_MISSING 1**. Then, log out of the
KDE desktop, and log back in.

The error message appears in one or both of the following situations:

■ Some clips in your libraries or desktops reference missing frames on a
remote volume.

■ Some clips in your libraries or desktops reference local frames that no
longer exist.

To identify the type of issue:

1 Open a terminal and type:

   vic -v<volume_name> **-s remote -s lost**

   where <volume_name> is the name of the affected volume, for example,
   stonefs or standardfs.

   The output of this command should be similar to the following:

   ```
   Checking libraries for remote and lost frames...
   /usr/discreet/clip/stonefs/My_Project1/editing.000.desk
   has none
   /usr/discreet/clip/stonefs/My_Project1/Default.000.clib
   references 30 missing frames.
   /usr/discreet/clip/stonefs/My_Project2/editing.000.desk
   has none
   /usr/discreet/clip/stonefs/My_Project2/from_caplan.000.clib
   references 70 remote frames
   ```

2 Depending on the result of the previous command, do one of the following:

- For clips with frames listed as missing, contact your local Autodesk support representative for help on resolving this issue.

- For clips with frames listed as remote, perform the following procedure.

To recover remote frames:

1 Archive all clip libraries that are reported as containing remote frames. In the previous example, the fourth library (from_caplan) contains remote frames.

2 Rename the libraries you just archived, such as by adding the remote suffix to the library name.

3 Restore the clip libraries from the archive. All the frames referenced by the clips should now be stored locally.

4 Delete the original libraries.

Verifying that the Disks are Available to the Operating System

If your disks or logical drives are not visible in the output of the **sw_df** command, verify that they are visible to the operating system.

To verify that the disk/controller is available to the operating system:

1 Run the command that will provide you with a list of all the controllers or disks on your system.

cat /proc/scsi/scsi

A listing of all controllers and hard disks attached to your system appears.

2 If any controllers or hard disks in your configuration are not visible, check the connections and cabling to your storage device, and ensure that your storage hardware is running.

If you must reconnect or tighten connections to your storage hardware, it is a good idea to rescan your hardware to ensure that the operating system is able to communicate with it.

To scan your storage hardware:

1 Unload the Stone and Wire driver. Type:

/etc/init.d/stone+wire unload

2 Unload and reload the fibre channel drivers. Type one of the following sequence of commands, according to your fiber channel card model.

■ If you have a QLogic™-based card, type:

rmmod qla2300

rmmod qla2400

rmmod qla2xxx

rmmod qla2xxx_conf

modprobe qla2xxx

modprobe qla2300

modprobe qla2400

■ If you have an ATTO-based card, type one of the following sequences of commands:

For HP Z800 workstations, type:

rmmod celerity8fc

modprobe celerity8fc

For older HP workstations, type:

rmmod celerityfc

modprobe celerityfc

NOTE Depending on the storage you are running, your system might not use all of the drivers listed. If your system does not use a driver listed, the commands to unload or reload the drivers will fail. You can ignore these failures. They just indicate that the driver is not required by your system.

3 Reload the Stone and Wire driver. Type:

/etc/init.d/stone+wire reload

Your filesystem should now be mounted.

Verifying Disk Usage

You cannot write to storage partitions that are completely full.

If you receive error messages stating that the framestore is full, use the **sw_df** command to check disk usage.

To use the *sw_df* command to see if the filesystem is full:

1 Go to the Stone and Wire directory */usr/discreet/sw* and run **sw_df**:

   ./sw_df

   This command returns the amount of free or used space on your filesystem.

If your filesystem is full or almost full, consider deleting or archiving older material from clip libraries to free disk space.

NOTE VTR archiving in Visual Effects and Finishing applications requires free space on the framestore to generate slates, test patterns, headers, and audio encoding.

You can also clear the Undo Buffer to free up space.

Clearing the Undo Buffer

The Undo Buffer takes up space on your filesystem. Consider clearing it if your filesystem is full or almost full to free disk space.

WARNING Clearing the Undo Buffer cancels any undoable operation. Consider what operations might be stored in the Undo Buffer before you clear it.

To clear the Undo Buffer:

1 Open your Visual Effects and Finishing application, and go into the Preferences menu.

2 Click Clear Buffer in the Undo section.

Solving a Framestore ID Mismatch

The framestore ID must match the filesystem ID for the workstation to mount your filesystem.

If the IDs do not match, you must change your framestore ID to ensure that all hosts on your Wire network have a unique ID.

To resolve a Framestore ID mismatch error:

1 Open a terminal as root, and list the contents of the the */usr/discreet/sw/cfg/sw_storage.cfg*file, by typing:

cat /usr/discreet/sw/cfg/sw_storage.cfg

The output of the command includes the [Framestore] section, which lists the framestore ID.

2 Open the */usr/discreet/sw/cfg/sw_framestore_map* file in a text editor.

The *sw_framestore_map* file includes a section similar to the following:

```
[FRAMESTORES]
FRAMESTORE=london HADDR=172.16.129.32 ID=32
```

The Framestore ID value in the *sw_framestore_map* file must match the Filesystem ID value listed in the *sw_storage.cfg* file.

NOTE If the framestore ID in the *sw_storage.cfg* file is not the same as the one in the *sw_framestore_map* file, the ID listed in *sw_storage.cfg* has priority.

3 Change the Framestore ID value in the *sw_framestore_map* file to match the Filesystem ID listed in the *sw_storage.cfg* file.

NOTE The last sequence of numbers in the IP address defined by the HADDR keyword in the *sw_framestore_map* file does not have to match the Framestore ID. These values are often the same by default, but it is not a requirement for Stone and Wire operation.

4 Save and close the file.

5 Restart Stone and Wire. Type:

/usr/discreet/sw/sw_restart

6 If you continue to get error messages, contact Customer Support.

Solving a Partition ID Conflict

Each partition must have a different partition ID. If there are multiple partitions using the same ID, you will receive an error message.

To solve a Partition ID conflict:

1 Note the ID listed in the error message.

2 Stop Stone and Wire. Type:

/etc/init.d/stone+wire stop

3 Open the */usr/discreet/sw/cfg/stone+wire.cfg* file in a text editor.

4 Find the [PARTITION#] section, where # is the ID listed in the error message. Change the # for an unused ID.

NOTE Partition identifiers go from 0 to 7, giving a total of 8 possible partitions. Stone filesystem and standard filesystem partitions share the same pool of identifiers, but only standard filesystem ones are listed in *stone+wire.cfg*. It is recommended to start numbering standard filesystem partitions at 7, decrementing with each additional standard filesystem partition.

5 Save and close the *stone+wire.cfg* file.

6 Start Stone and Wire. Type:

/etc/init.d/stone+wire start

A message appears in the terminal indicating that Stone and Wire is restarted.

7 Start your Visual Effects and Finishing application to verify that the partition ID conflict is resolved.

Removing Invalid Frame References from the Stone and Wire Database

Stone and Wire has a database inconsistency detection mechanism. If you change the framestore ID of a Stone filesystem or standard filesystem partition in your system without reformatting the Stone and Wire database, the inconsistency check mechanism detects invalid frame references in the database, and prevents the Stone and Wire database server (*sw_dbd*) from starting.

If you receive an error message that *sw_dbd* could not be started when you start a Visual Effects and Finishing application or Stone and Wire, perform the following steps to determine if the problem is related to invalid frame references, and to remove those references from the Stone and Wire database.

To remove invalid frame references from the Stone and Wire database:

1 Open a terminal and log in as root.

2 Search the */usr/discreet/sw/log/sw_dbd.log* file for the text "Bad framestore ID":

cat /usr/discreet/sw/log/sw_dbd.log | grep -i bad/ framestore/ id

■ If the search returns one or more lines, the problem is related to invalid frame references. Perform the remaining steps in this procedure to clean up the Stone and Wire database.

■ If the search returns no results, the problem is not related to invalid frame references. Read the */usr/discreet/sw/log/sw_dbd.log* file to determine the cause of the problem.

3 Stop Stone and Wire by typing:

/etc/init.d/stone+wire stop

4 Clean up the Stone and Wire database by typing:

/usr/discreet/sw/sw_dbd -C

A message is displayed in the terminal asking you to confirm the deletion of invalid entries from the database.

5 Type **Y** to confirm the operation.

Invalid entries are removed from the Stone and Wire database.

6 Restart Stone and Wire by typing:

/etc/init.d/stone+wire start

Troubleshooting Wire Network Problems

This section describes some common Wire problems and steps you can take to solve them.

Testing Wire Using sw_framestore_dump

Use *sw_framestore_dump* to identify the Wire hosts that are reachable from one workstation. You should do this if you are having any problems with Wire or after you configure Wire for the first time.

NOTE You must have at least two Wire hosts set up to test Wire connectivity.

To verify that you can access the framestores on other Wire hosts:

1 Type:

/usr/discreet/sw/tools/sw_framestore_dump

A list of all the available framestores appears in the terminal. Review the output and verify that all the framestores on your Wire network appear in the list.

You can also verify that each framestore on your network is using a unique framestore ID.

If:	Then:
You are unable to see other Wire hosts on your network	Check: ■ The filesystem and networking configurations in the *sw_framestore_map* file. ■ The *sw_probed.cfg* port number. ■ You may also have problems with your Wire network. Use *sw_ping* to test that you can connect to other Wire hosts. See Using sw_ping to Test Wire Communication (page 228).
You see only some of the Wire hosts (as opposed to all or none)	Check that each framestore has a unique Framestore ID.

2 Repeat this procedure on each Wire host.

Using ping to Test Network Communication

Try to ping your local host from a client machine. If this works, ping all other machines that should be accessible through Wire.

To use *ping* to test network communication:

1 Type the following command:

ping <host_name>

2 If ping fails, try using the machine's IP address (for example, 172.16.100.23) instead of its name. Type:

ping<IP_address>

If this is successful, verify how the machine resolves host names on the network. You should set the order of host name resolution to first look at the local setup file, then validate on the network. The */etc/nsswitch.conf* file should include a "hosts" line that has the name validation process in the following order:

hosts: files nis dns

Using sw_ping to Test Wire Communication

If you are unable to connect to a Wire host, use *sw_ping* to determine if you can connect to the other Wire host over the Wire network.

To use *sw_ping* to test Wire communication:

1 Change to the Stone and Wire directory:

cd /usr/discreet/sw

2 Start *sw_ping*. Type:

./sw_ping -host <host_name> **-loop** <count> **-size** <packetsize> **-r -w**

Where:	Is:
<host_name>	The name of the host to ping.
<count>	The number of times to execute this test.
<packetsize>	The size of the read/write buffer (in bytes).

For example, type:

./sw_ping -framestore tunisia -loop 4 -size 9000000 -r -w

Results similar to the following are reported to the terminal:

```
Using Stone+Wire API 2009 [ProductSW_1166] (64 bits).

Stone+Wire TCP port set to 7000.

Ping successful to host tunisia (186)

Buffer size = 9043968

Sending data to tunisia, filesystem id 186

MB/sec: 111.460127

Ping successful.

Buffer size = 9043968

Sending data to tunisia, filesystem id 186

MB/sec: 111.509057
```

```
Ping successful.

Buffer size = 9043968

Sending data to tunisia, filesystem id 186

MB/sec: 111.500577

Ping successful.

Buffer size = 9043968

Sending data to tunisia, filesystem id 186

MB/sec: 111.516597

Ping successful.

Minimum MB/sec: 111.460127

Maximum MB/sec: 111.516597

Average MB/sec: 111.496590
```

If:	Then:
An *sw_ping* from machine 1 to machine 2 is successful and you still cannot access the remote framestore through your Visual Effects and Finishing application	Check that the remote system has exported its directories so that the local system has read/write access. See Verifying Remote Clip Library Access and Permissions (page 230).
An *sw_ping* from machine 1 to machine 2 is unsuccessful	■ Validate the *sw_framestore_map* file on both machines. ■ Check that *sw_probed* and *sw_serverd* are running on both machines.

Using sw_ping to Test Network Performance

Use the *sw_ping* command to test network performance. For more significant results, run the test 100 times.

To test network performance:

1 Start *sw_ping*. Type:

/usr/discreet/sw/sw_ping -framestore <framestore_name> **-r** **-w -size** <packetsize> **-loop** <n>

Option:	Description:
-framestore <framestore_name>	Is the name of the framestore to ping.
-r	Reads a buffer from the remote framestore.
-w	Writes a buffer to the remote framestore (non-destructive).
-size <packetsize>	Reads/writes a buffer of packetsize bytes.
-loop <n>	Executes this test n times.

The following is an example for an NTSC (720 x 486 x 3) frame format, sent 100 times in read and write mode to framestore *my_framestore*:

```
/usr/discreet/sw/sw_ping -framestore my_framestore -read
-write -size 1049760 -l 100
```

Verifying Remote Clip Library Access and Permissions

Verify that the local host has write permissions to the clip library directory of the remote host.

To verify remote clip library access and permissions:

1 Try to access the clip library directory of the remote host. Type:

   cd /hosts/<remote_machine>**/usr/discreet/clip**

2 If an error message appears on the client machine, check to see that the required network daemons are on.

3 If you have access to *hosts/<remote_machine>/usr/discreet/clip*, make sure you have root privileges to read and write to this directory.

   Type the following command:

   touch /hosts/<remote_machine>**/usr/discreet/clip/deleteme**

If error messages appear, verify the permissions in the */etc/exports* file on the server machine. Read and write permission must be given to the hosts. The appropriate line should look similar to the following:

/usr/discreet/clip *(rw,sync,no_root_squash

Verifying that NFS and Automounting Daemons are Running

Network File System (NFS) and automounting daemons (AMD on Linux) must be running for Wire transfers to work.

These daemons are installed and running by default. Use the following procedures to verify that these daemons are running.

To start NFS and AMD:

1   Type the following command:

chkconfig --list | grep nfs

chkconfig --list | grep amd

2   Verify that NFS and AMD are set to "on" for run levels 3,4, and 5.

3   If NFS or AMD is off on any of those run levels, type:

chkconfig nfs on

chkconfig amd on

4   Restart your network. Type:

/etc/init.d/network restart

You might also consider rebooting your workstation.

Checking the Status of Network Interfaces

If you continue to have problems with your network, you should verify that your network interfaces are up and running.

To check the status of your network interfaces:

1   Check the status of your network interfaces, by typing:

ifconfig

■ If your network interface is up and running, an "up" appears in the broadcast report for the interface. The report includes a line similar to the following:

```
UP BROADCAST RUNNING MULTICASTMTU:1500Metric:1
```

■ If your network interface is not up and running, check the connections on your network card. A green light appears when there is a good connection between your network card and its destination.

If you must reconnect cables on Linux, you must restart the network interface.

To restart a network interface:

  1 Type the following command:

     ifconfig <interface_name>**up**

Viewing InfiniBand Statistics

This section describes how to view port statistics for the workstation and InfiniBand switch, and how to obtain InfiniBand log files.

For more information about configuring, monitoring, and managing the InfiniBand switch and network, see the documentation provided by the manufacturer of your InfiniBand switch.

Viewing InfiniBand Information for a Workstation

The InfiniBand driver installed on your workstation provides two commands that output statistics and information about InfiniBand ports.

To review statistics and information about a port:

  1 Log in as root, and type the following commands in a terminal:

     cat /proc/iba/<driver_id>**/1/port**<x>**/stats**

     cat /proc/iba/<driver_id>**/1/port**<x>**/info**

     where <x> is the port number on the device, and <driver_id> is the
     HCA driver ID for your device, for example: *mt25218*.

> **NOTE** To find out the HCA (Host Channel Adapter) driver ID, type **ls /proc/iba/** | **grep mt**. The driver number, beginning with *mt*, will be returned.

A report appears in the terminal for each command.

Generating InfiniBand Log Files for Support

The script */sbin/iba_capture* gathers information about your system and InfiniBand configurations that can be useful for troubleshooting InfiniBand issues. This script outputs a GZIP file.

To run the InfiniBand log file capture script:

1 As root, open a terminal on your workstation and type:

   /sbin/iba_capture `<path and name of output gzip file>`

   A GZIP file is generated that includes a number of log files from your system.

   > **NOTE** The help for this command incorrectly indicates that the output is a TAR file, when it is in fact a GZIP file.

Viewing Port Statistics for a Mellanox IS5030 Switch

The Mellanox IS5030 switch is the recommended switch model for QDR InfiniBand networks. It comes with a Web-based management console that you can use to configure the switch and monitor the InfiniBand network.

When you suspect that performance is not optimal on your QDR switch, it is a good idea to view statistics for each connected port, to determine if they are running normally. The nominal speed for DDR InfiniBand connections is 5 Gbps, and the nominal speed for QDR connections is 10 Gbps.

To access the management console, your browser must be on the same subnet as the InfiniBand switch, which has a default IP address of 10.10.10.252.

To view switch port statistics:

1 Open a Web browser and enter the IP address of the InfiniBand switch. The default IP for switches shipped from Autodesk is 10.10.10.252.

   The login page appears.

2 Enter your user name and password. The defaults are *admin* / *admin*.

The Summary page appears.
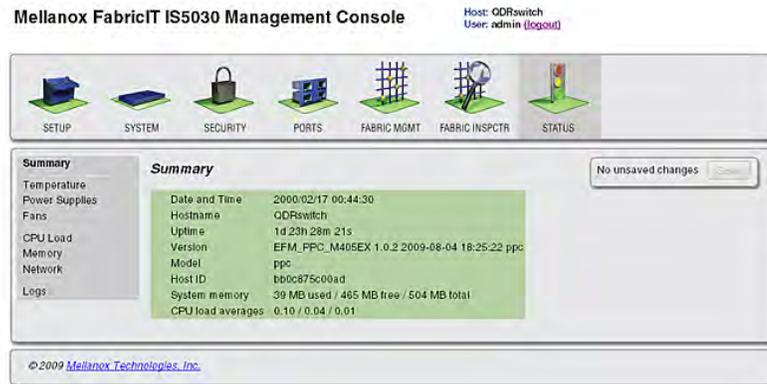
3 Click Ports in the menu at the top.

The Ports page appears, displaying an overview of the switch. Connected ports are displayed in green.
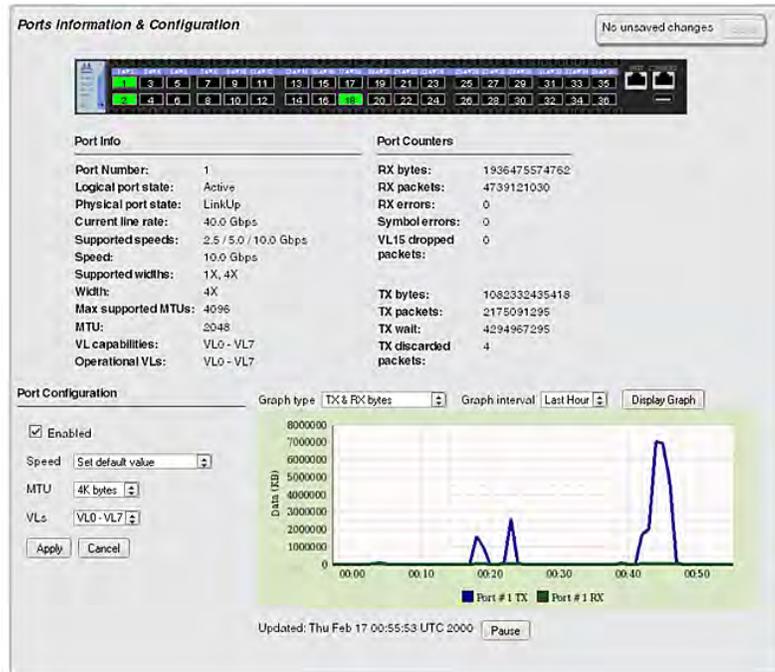
4 Click a port to view information and statistics on it.

> **NOTE** If you have ports with DDR connections that appear to be running at SDR speed (2.5 Gbps instead of 5 Gbps), unplug the cable and then plug it back in. The connection should run at normal DDR speed afterwards. This issue occurs because of a bug in the switch firmware.

Viewing Port Statistics for an InfiniCon InfinIO 9024 Switch

The InfinIO 9024 InfiniBand switch is the recommended switch model for DDR InfiniBand networks.

The InfinIO switch comes with a Web-based application, the InfiniView Device Manager™, that you can use to configure, monitor, and manage the InfiniBand network.

When you suspect that performance is not optimal on your QDR switch, it is a good idea to view statistics for each connected port, to determine if they are running normally. The nominal speed for DDR InfiniBand connections is 5 Gbps.

To access the Device Manager, your browser must be on the same subnet as the InfiniBand switch, which has a default IP address of 10.10.10.252.

To view switch port statistics:

1 Open a Web browser and enter the IP address of the InfiniBand switch, usually 10.10.10.252.

The main page of the Device Manager appears.

2 Click Port Stats in the menu on the left.

The IB Port Statistics page appears.

The page displays the status of each port on the switch.

3 Optional: Click Refresh to update the data on the page.

Index

242