

ClearCase and MultiSite Release Notes: Addendum for Linux on IBM Mainframe Systems

Release 4.2, November 2001
Part No. 800-025396-000

This document supplements the *ClearCase and MultiSite Release Notes* for release 4.2 of Rational ClearCase with information specific to installing and running ClearCase on IBM mainframe systems (S/390 and zSeries) running Linux. Refer to the full set of documentation for Release 4.2, in addition to this addendum. The Release 4.2 documentation has not been updated to include specific information for Linux on IBM mainframes. However, you may find the portions of the documentation that contain information for other Linux platforms to be useful.

This addendum describes important information in this release. Read it *before* you attempt to install Rational ClearCase or Rational ClearCase MultiSite.

For more information about differences in features and functionality by platform, see the *ClearCase Platform-Specific Guide*. The platform guide is available in online format only and is part of the ClearCase Help system. It's listed with other topics on the **Contents** tab of ClearCase **Help Topics**.

NOTE: The term CPF stands for ClearCase Product Family and refers to the products ClearCase, Attache, and MultiSite. ClearGuide is not included in Release 4.2. The term 4.x refers to 4.0, 4.1, and 4.2.

This document is divided into these parts:

- *Release Notes*
- *Architecture-Specific Hardware and Software Requirements*
- *General Issues for IBM Mainframe Systems*
- *Installation and Configuration Issues for Linux on IBM Mainframe Systems*
- *Functionality Specific to Linux*

1.1 Release Notes

Supported Architectures

Rational ClearCase and Rational ClearCase MultiSite Release 4.2 for SuSE Linux run on the platforms listed in Table 1.

Table 1 Supported Platforms for ClearCase and MultiSite Release 4.2

Hardware platform	Operating system
IBM S/390 and zSeries ¹	SuSe Linux 7.0 (2.2.16 kernel)

¹The notation zSeries as used in these release notes refers to “Linux for S/390” running on the IBM zSeries hardware in the 31-bit run-time environment. Running ClearCase on “Linux for zSeries” in a 64-bit runtime environment is not supported.

This release of ClearCase does not include support for the following architecture:

- ▶ Linux for zSeries (64-bit runtime environment)

This release has been tested on Linux for S/390 images running under VM and Linux running native. It has not been tested for Linux running under VIF or for Linux running on the IBM zSeries, although these architectures are supported (with the limitations for zSeries noted above).

Supported Platforms for ClearCase Web Servers and Web Interface

Web Browser

The SuSE Linux distribution currently does not support any Web browser. Therefore, accessing ClearCase on another host by means of the Web client running on the IBM mainframe cannot be supported. However, you can access ClearCase by means of Web clients on other platforms. For more information, see *ClearCase and MultiSite Release Notes* for Release 4.2.

Web Server

The Apache Web server supports Linux on IBM mainframes and accessing ClearCase on the IBM mainframe using the Web Server is supported.

Supported File Systems

Table 2 lists the file systems that ClearCase supports for view and VOB storage for IBM mainframes.

Table 2 Supported File Systems by Platform

Platform	Supported file systems
IBM S/390 and zSeries	UFS, EXT2

NFS Support

General Information

For a given platform, we support the NFS implementations supported by the platform.

Non-ClearCase access is not supported for Release 4.2 on IBM mainframes running Linux.

SuSE Distribution

Use of ClearCase with third-party automounters is not supported on any platform. However, in recognition that some users will make use of automounters, the following information is provided.

The automounter distributed by SuSE (**autofs**) has incompatibilities with ClearCase. Use of **autofs** is acceptable as long as ClearCase data (such as views and VOBs) is not accessed through an automounted path. For ClearCase data, two solutions are available:

- Directly NFS-mount the view or VOB storage areas on `/net/host-name` without automounter intervention.
- Install the **amd** automounter, which does not exhibit those behaviors that make **autofs** incompatible. However, if **amd** is installed, the **amd** startup script (`/etc/rc.d/init.d/amd`) must be modified as follows to run on the SuSE Linux distribution:
 - > Comment out the line `/etc/rc.d/init.d/functions`.
 - > Rename all references to the function **daemon** in the start case to **startproc**.

1.2 Architecture-Specific Hardware and Software Requirements

The following sections describes architecture-specific software and hardware requirements for running ClearCase and MultiSite. For general requirements, see *ClearCase and MultiSite Release Notes* for Release 4.2.

Disk Space Requirements for the Release Area

The file system of the networkwide *release host* must have sufficient disk space to hold the release area:

IBM S/390 and zSeries 119 MB

Exporting the Release Area

Setting up the export is architecture specific (Table 3). For details, see the standard reference pages for these files and programs.

Table 3 Exporting the ClearCase Product Family Release Area

Architecture	Steps to export ClearCase product family release area
SuSE Linux for IBM S/390 and zSeries	Revise <code>/etc/exports</code> <code>/(rw, no_all_squash)</code>

Architecture Mnemonics and mount Commands

Table 4 provides the architecture mnemonic and sample CD **mount** commands for supported platforms. The architecture mnemonic is used as the name of root of the release area for each platform or set of platforms.

Table 4 Mounting the CD

Architecture	Mnemonic	Sample mount command
SuSe Linux for IBM S/390 and zSeries	linux_390	<code>mount -r /dev/cdrom /mnt/cdrom</code>

The CD drive may not be mountable from a Linux image running under VM. In that case, use a CD drive on another computer accessible over the network and mount that drive from the Linux image under VM.

Disk Space Requirements for Individual Hosts

Table 5 shows the disk space requirement for each kind of installation. All the space must be contained in a single disk partition.

Table 5 Disk Space Requirements for ClearCase Product Family Releases

Architecture	Type of installation	ClearCase	MultiSite	Attache helper
SuSe Linux for IBM S/390 and zSeries	Full copy	111 MB	4 MB	2 MB
	Standard	31 MB	850 K	2 MB

1.3 General Issues for IBM Mainframe Systems

This section discusses general issues related to Rational ClearCase Release 4.2 on IBM mainframes running Linux.

The following are known limitations:

- **Symmetric Multiprocessing.** The Linux kernel as distributed by SuSE contains support for symmetric multiprocessing (SMP). It is possible to build a kernel without this support, but the SMP kernel is the only one supported for ClearCase.
- **The /etc/exports File.** The /etc/exports file contains entries of file systems that are to be made available for sharing with other computers. If these entries are to be used for ClearCase data, the **no_all_squash** attribute must be added to the option list. For example:

```
/ (rw,no_all_squash)
```

This attribute prevents remapping of the uid and gid of files on the root directory to the anonymous user, which is incompatible with ClearCase.

- **MVFS Support.** Because of design features of the Linux 2.2.x kernel series on which Linux for IBM mainframes is based, you must patch and rebuild the kernel before you can run the ClearCase MVFS (multiversion file system). If the patch is not installed, the MVFS fails to load, and a diagnostic message is printed to the system log in `/var/log/messages`.
- **Non-ClearCase Access Using NFS Is Not Supported.** Because of limitations in the interfaces between NFS and the MVFS, non-ClearCase access using NFS is not supported.

1.4 Installation and Configuration Issues for Linux on IBM Mainframe Systems

This section discusses installation and configuration issues relevant to installing Rational ClearCase Release 4.2 on IBM mainframes running Linux.

Before Rebuilding the Kernel

You must patch and rebuild the kernel before you can run the ClearCase MVFS (multiversion file system). Rebuilding a Linux kernel is not always straightforward because variety of hardware and the variety of configurable options are available under Linux. If you are not familiar with the process of building and rebooting the Linux kernel, we recommend that you try this before you install ClearCase. For more information on configuring and compiling the Linux kernel, see the following Web sites:

- www10.software.ibm.com/developerworks/opensource/linux390/index.shtml
- www.linuxdoc.org/HOWTO/Kernel-HOWTO.html

Useful information is also available in the Linux release notes, which you can find in the README file in the top-level kernel source directory.

Installation Tasks

The tasks presented in the following sections must be performed before you can run the MVFS on IBM mainframes running Linux:

1. Installing the Linux Kernel Sources
2. Installing Rational ClearCase or ClearCase MultiSite
3. Additional ClearCase Installation Steps for Linux on IBM Mainframes
4. (Optional) Manually Applying the Linux Kernel Patch for IBM Mainframes
5. Configuring the Linux Kernel After Applying the Kernel Patch

Installing the Linux Kernel Sources

The Release 4.2 installation process expects that the Linux kernel sources are loaded on your system. If installed, these sources are located in `/usr/src/linux`.

If you do not have the Linux sources installed, follow these instructions:

1. Log on as **root**.
2. Insert the Linux distribution in your CD drive. If necessary, mount the drive. For Linux (kernel version 2.2.16), insert Disk 1.
3. Change to the `/full-names/s390` directory on the CD (or to the directory where you downloaded the SRPMs).
4. Enter the following command:

```
# rpm -ivh lx_suse-2.2.16.SuSE.s390.rpm
```

Installing Release 4.2

Before Beginning the Installation

Before installing Rational Clearcase, go to the `/boot` directory and check that there is a link from `System.map` to `System.map-2.2.16`. If there is no link, create one with the following command:

```
# ln -s System.map-2.2.16 System.map
```

Follow the Rational ClearCase Installation Procedure

Then, follow the instructions in the *ClearCase Product Family Installation Notes* to install the Release 4.2 software.

Additional Installation Steps for Linux on IBM Mainframes

This section presents installation steps specific to Linux systems. These steps are presented after the standard installation selections.

1. Do you want to patch your Linux sources (yes, no, quit, help) [yes]:
 - > Answer **yes** to apply the patch (recommended if the patch has never been applied).
The product install process attempts to apply the kernel patch to the Linux source directory tree.
 - > Answer **no** to skip applying the patch.
Answer **no** only if this patch was previously applied.
2. Do you want to relink the MVFS when install is complete (yes, no, quit, help) [no]:

- > Answer **no** to leave the MVFS as is. The MVFS is able to run if the symbols used by the MVFS in the release area match the symbols in your running kernel. This is the normal case.
 - > Answer **yes** to relink the MVFS module using those kernel sources, and set up the relinked module for use. Normally, this should not be necessary.
3. You are prompted for the following information only if you have answered yes to either or both of the questions presented in Step #1 and Step #2.

Enter the path to your top-level kernel source directory.
Default is [/usr/src/linux]:

If your Linux kernel sources have been installed in a different directory, specify the directory here.

Let the install process complete. The following sections outline the subsequent steps and issues associated with how you answered the patch question in step 1 above.

If You Answered **yes** to the Kernel Patch Question in Step #1

If the kernel patch was not initially present, it is now present, and the first part of the product installation is complete. Follow the next steps to finish the product installation.

1. Rebuild the kernel and reboot the system.

For instructions, see the section *Configuring the Linux Kernel After Applying the Kernel Patch* on page 9. After the ebuild and reboot are complete, continue to Step #2 below.

2. Reinstall ClearCase.

With the kernel rebuilt and the system rebooted, you must reinstall ClearCase to complete the product installation.

When reinstalling ClearCase, you must answer the following questions as specified below to ensure that ClearCase includes a working version of MVFS:

- a. Do you want to patch your Linux sources (yes, no, quit, help) [yes]:

Answer **no**. You patched the Linux sources before the kernel was rebuilt. It's not necessary to patch them again.

- b. Do you want to relink the MVFS when install is complete (yes, no, quit, help) [no]:

Answer **no**.

After the reinstall is complete you are finished. Clearcase and MVFS should both be running on your system.

If the kernel patch was initially present, after you answer the standard install questions, the install process displays a message indicating that the patch cannot be installed because it is already present. You are then asked if you want to continue the ClearCase product installation. Answer **yes**.

At this point the product is installed and the previously applied kernel patch is present in the Linux source directory tree.

- If the kernel rebuild, system reboot, and reinstall of ClearCase associated with that kernel patch have also been done, installation is complete. Clearcase and MVFS should both be running on your system
- If the kernel patch has been applied but the kernel rebuild, system reboot, and reinstall of ClearCase has not been performed, you must do that now. For instructions on rebuilding and rebooting, see *Configuring the Linux Kernel After Applying the Kernel Patch* on page 9. For information on the reinstall step, see Step #2 earlier in this section.

If You Answered *no* to the Kernel Patch Question in Step #1

If the kernel patch was previously applied, it is now present in the source directory tree and the product is fully installed.

- If the kernel rebuild, system reboot, and reinstall of ClearCase associated with that kernel patch have also been done, installation is complete. Clearcase and MVFS should both be running on your system.
- If the kernel patch has been applied, but the kernel rebuild, system reboot, and reinstall of ClearCase has not been performed, you must do that now. For instructions on rebuilding and rebooting, see *Configuring the Linux Kernel After Applying the Kernel Patch* on page 9. For information on the reinstall step, see Step #2 earlier in this section.

If the kernel patch was not initially present, the MVFS does not load, and the product install fails. In this case you must apply the patch. For instructions, see *Manually Applying the Linux Kernel Patch for IBM Mainframes* on page 8. You must then rebuild the kernel and reboot the system. For instructions, see *Configuring the Linux Kernel After Applying the Kernel Patch* on page 9.

Manually Applying the Linux Kernel Patch for IBM Mainframes

This section explains how to apply the Linux kernel patch manually. The instructions apply for Linux running on IBM mainframes. A precondition is that Rational ClearCase is installed and the kernel patch has not been applied.

NOTE: You do not need to follow this procedure if you chose to have the Release 4.2 install apply the Linux kernel patch (the recommended method) as described in the section, *Additional Installation Steps for Linux on IBM Mainframes* on page 6.

1. Copy the following patch to `/usr/src/linux` (or to the top-level Linux kernel source directory that you specified in Step #3 of *Additional Installation Steps for Linux on IBM Mainframes*). The patch file is `linux-mvfs-4.2-patch-2.2.16`, located in the directory `/usr/atria/etc/conf/linux_390_2216`.

For example:

```
# cp /usr/atria/etc/conf/linux_390_2216/linux-mvfs-patch-2.2.16 /usr/src/linux/.
```

2. Change to the `/usr/src/linux` directory (or to the top-level Linux kernel source directory that you specified in Step #3 of *Additional Installation Steps for Linux on IBM Mainframes*):

```
# cd /usr/src/linux
```

3. To patch the Linux sources, run the following command:

```
# patch -Np1 < linux-mvfs-4.2-patch-2.2.16
```


Verify that no errors were reported.

After you apply **linux-mvfs-4.2-patch-2.2.16**, you must rebuild the Linux kernel and reboot the system. (For the procedure, see the next section, *Configuring the Linux Kernel After Applying the Kernel Patch* on page 9).

Configuring the Linux Kernel After Applying the Kernel Patch

1. If you have built your own Linux kernel, your system already has a **.config** file. In this case, we recommend that you back up the file because the process described in this step will overwrite it.

- a. Go to the directory **/usr/src/linux**:

```
# cd /usr/src/linux
```

- b. Create a configuration file by issuing the following commands:

```
# make mrproper
```

```
# make menuconfig
```

The command **make menuconfig** brings up a menu of kernel configuration options. Selected items are indicated by an asterisk or an **M**. **NOTE:** Do not select 2 GB memory from the CPU-specific parameters screen. Doing so will cause **mvfs.o** to fail.

- c. If you do not have a configuration file on your system, select the **Load an Alternate Configuration File** menu from the **menuconfig** utility main menu. Take the default that appears.
- d. Return to the main menu and select the **Processor type and features** menu. In the **Processor type and features** menu, select **Symmetric multi-processing support**. This is the only supported configuration.
- e. Return to the main menu and select the **Loadable module support** menu. On the **Loadable module support** menu, ensure that both of the following options are selected:
 - * Enable loadable module support
 - * Kernel module loader
- f. Return to the main menu and select the **Filesystems** menu. In the **Filesystems** menu, select **Kernel automounter support** and then select the **Network File Systems** submenu. In the **Network File Systems** menu, select **NFS filesystem support** and **NFS Server support**.
- g. Exit the **menuconfig** utility. When prompted, enter **yes** to save the new **.config** file.
- h. Run the following command:

```
# make dep
```

2. From your **/usr/src/linux** directory, create a compressed Linux kernel image. Type the following:

```
# make image
```

NOTE: If you have turned on or off any of the modules in addition to the default configuration, it may be necessary to run the following commands:

```
# make modules
# make modules_install
```

3. The **install** command can be used to copy the new kernel and related files and back up the necessary files for recovering from a bad kernel build. Backing up these files is highly recommended. Use the following commands to copy the rebuilt kernel files to the **/boot** directory, so that they can be used the next time you boot the machine.

```
# install -backup -suffix=.save System.map /boot/System.map-2.2.16
# install -backup -suffix=.save .config /boot/image.config
# install -backup -suffix=.save include/linux/autoconf.h /boot/image.autoconf.h
# install -backup -suffix=.save include/linux/version.h /boot/image.version.h
# install -backup -suffix=.save arch/s390/boot/image /boot/image
# install -backup -suffix=.save arch/s390/boot/ipldump.boot /boot/ipldump.boot
# install -backup -suffix=.save arch/s390/boot/ipleckd.boot /boot/ipleckd.boot
# install -backup -suffix=.save arch/s390/boot/iplfba.boot /boot/iplfba.boot
```

NOTE: In the event that you have built a kernel that does not boot, these files are crucial because they can be used to restore the previous kernel. This can be done by booting from a second disk (or the initial ramdisk) and mounting the first disk with the bad kernel. Then copy all the ***.save** files in the **/boot** directory to their original names and boot off the first disk.

4. To complete the kernel build, type the following:

Note the path **/dev/dasdb** is given as an example. The appropriate path depends on the path to the boot device on your system.

```
# cd /boot
# silo -d /dev/dasdb
```

5. Reboot the system:

```
# reboot
```

1.5 Functionality Specific to Linux

This section describes Release 4.2 information specific to the Linux platform, with references to other places in the documentation set where more information can be obtained.

NFS, Automounter, and root ClearCase Access

A **root** user of ClearCase will find that the Linux operating system's NFS implementation can cause some unexpected behavior. For example, using the default **/etc/exports** file, **root** can create a view that cannot be removed by **root**.

If, when using Linux NFS and the automounter, a user references **/net/hostname**, the host is NFS mounted on the mount point specified in the automounter configuration file. This is true even if the user is currently on host. The default rule for NFS mounts—unless explicitly overridden in the **/etc/exports** file—is to map requests from **uid/gid=0** (that is, **root**) to the anonymous uid/gid.

This means that under the default configuration, a file owned by **root** cannot be removed by **root** by means of the `/net/hostname` path.

ClearCase creates files locally, but accesses these files through global path specifications (by default, `/net/hostname/...`). Therefore, if **root** creates a view on the local machine, all the special files that ClearCase creates in the view are owned by **root**. However, if **root** tries to remove the view, the view is accessed through the global path, and the special ClearCase files are not deleted because the ownership and permissions do not allow deletion by the anonymous user.

Workaround: To avoid this problem, the `/etc/exports` entries must include the option `no_root_squash`. This option allows all ClearCase activities to proceed normally. Note that you must exercise caution when using the `no_root_squash` option because it allows any **root** user—on any host that NFS mounts a host's exported file system—to have **root** permissions on that file system. This can be a dangerous security problem in some environments.

Typically, **root** does not create views, and the issue arises only on those ClearCase administrative tasks that require **root** privileges, for example, using the `protectvob` command. In such cases, we recommend that the modifications to `/etc/exports` be made temporarily and restored after the ClearCase administrative tasks are performed.

NOTE: The use of `/net/hostname` in global paths is pervasive in ClearCase, and the inability of ClearCase to access files through these global paths can cause anomalous behavior, often with no warning. Correct configuration of Linux to allow proper ClearCase access is essential for successful ClearCase operation.

Using SuSE Linux as a VOB Server

If you use SuSE Linux as a VOB server, you may receive intermittent error messages in ClearCase because of NFS caching issues in the Linux operating system. In most cases, when you perform the same Clearcase operation again, it succeeds.

The ClearCase configurations listed below are less likely to experience these issues:

- Remote Linux VOB server and a local Windows view server and client using the CCFS protocol, where remote refers to a separate host and local refers to the same host.
- Remote UNIX VOB server and a local Linux view server and client. Supported server platforms are Solaris SPARC, HP-UX, IRIX, AIX, Tru64 UNIX, UnixWare, Solaris Intel, Reliant UNIX, and Red Hat Linux on Intel x86. Refer to Table 1 of *ClearCase and MultiSite Release Notes for Release 4.2* for operating system releases. SuSE Linux on IBM S/390 and zSeries is also a supported server platform.

Linux mount Command Will Not Mount MVFS Directories

The Linux `mount` command shipped with SuSE Linux does not properly mount file systems other than those included in the default distribution. If you try to mount a ClearCase MVFS with the Linux `mount` command, you will cause a kernel fault and must reboot your system.

Workaround: Always use the `cleartool mount` command to mount VOBs.

Increasing the MVFS Cache Size

The chapter *Improving Client Host Performance in Administering ClearCase* discusses changes you can make to improve the performance of ClearCase clients. Use the information in this section in conjunction with the material in that chapter.

The instructions that follow increase the size of the MVFS cache in such a way that the changes are persistent.

Setting Individual Cache Sizes on Linux

On each ClearCase client:

1. Become **root** user.
2. Stop ClearCase by running the following script:

```
# /usr/atria/etc/atria_start stop
```
3. Change to the `/var/adm/atria/mvfs` directory:

```
# cd /var/adm/atria/mvfs
```
4. Edit `mvfs_param.c`. Set the MVFS configuration parameters as described in the chapter *Improving Client Host Performance of Administering ClearCase*.
5. If your kernel sources are somewhere other than `/usr/src/linux`, edit the Makefile in this directory so that `LINUX_KERNEL_DIR` points to the location of your kernel build directory. Alternatively, you can define it in the **make** command line as shown in the next step.
6. Run one of the following **make** commands:

```
# make
```



```
# make LINUX_KERNEL_DIR=/usr/mydir/mykernelsources
```
7. Run **make modules_install**:

```
# make modules_install
```
8. Restart ClearCase by running the following script:

```
# /usr/atria/etc/atria_start
```

Non-ClearCase Access Using NFS Is Not Supported

Because of limitations in the interfaces between NFS and the MVFS, non-ClearCase access using NFS is not supported.

Unable to Perform Certain File Operations on Sockets, Character Devices, and Block Devices When Set to a View

Because of limitations in the interface between the MVFS and Linux, when set to a view, you cannot perform certain file operations—such as the **mv** command—on sockets, character devices, and block devices. For example, if you attempt to rename the character device **ptyx3** using the command

```
# mv ptyx3 ptyx4
```

the following error message appears:

```
mv: cannot move 'ptyx3' to subdirectory of itself 'ptyx4'
```