

How to Install an OSCAR Cluster  
Quick Start Install Guide  
Software Version 2.0b3-v4.2.1b7  
Documentation Version 2.0b3-v4.2.1b7

<http://oscar.sourceforge.net/>  
[oscar-users@lists.sourceforge.net](mailto:oscar-users@lists.sourceforge.net)

The Open Cluster Group  
<http://www.openclustergroup.org/>

May 24, 2006

# Contents

<b>1</b>	<b>Introduction</b>	<b>4</b>
1.1	Latest Documentation . . . . .	4
1.2	Terminology . . . . .	4
1.3	Supported Distributions . . . . .	5
1.4	Minimum System Requirements . . . . .	5
1.5	Document Organization . . . . .	6
<b>2</b>	<b>Downloading an OSCAR Distribution Package</b>	<b>6</b>
<b>3</b>	<b>Release Notes</b>	<b>7</b>
3.1	Release Features . . . . .	7
3.2	Notes for All Systems . . . . .	7
3.3	SSS-OSCAR Release Notes . . . . .	9
3.4	Red Hat Enterprise Linux 4 & Fedora Core 3 & 4: SELinux Conflict . . . . .	11
3.5	Red Hat Enterprise Linux 4 . . . . .	11
3.6	CentOS 4 ia64 Notes . . . . .	12
3.7	Red Hat Enterprise Linux 3 . . . . .	12
3.8	Mandriva Linux (previously known as "Mandrake Linux") Notes . . . . .	13
3.9	Fedora Core 4 . . . . .	14
<b>4</b>	<b>Quick Cluster Installation Procedure</b>	<b>14</b>
4.1	Server Installation and Configuration . . . . .	14
4.1.1	Install Linux on the server machine . . . . .	14
4.1.2	Disk space and directory considerations . . . . .	15
4.1.3	Download a copy of OSCAR and unpack on the server . . . . .	15
4.1.4	Configure and Install OSCAR . . . . .	15
4.1.5	Configure the ethernet adapter for the cluster . . . . .	15
4.1.6	Copy distribution installation RPMs to <code>/tftpboot/rpm</code> . . . . .	16
4.2	Launching the OSCAR Installer . . . . .	17
4.3	Downloading Additional OSCAR Packages . . . . .	17
4.4	Selecting Packages to Install . . . . .	17
4.5	Configuring OSCAR Packages . . . . .	17
4.5.1	Selecting a Default MPI Implementation . . . . .	17
4.6	Install OSCAR Server Packages . . . . .	18
4.7	Build OSCAR Client Image . . . . .	18
4.8	Define OSCAR Clients . . . . .	19
4.9	Setup Networking . . . . .	20
4.10	Monitor Cluster Deployment . . . . .	21
4.11	Client Installations . . . . .	22
4.11.1	Network boot the client nodes . . . . .	22
4.11.2	Check completion status of nodes . . . . .	22
4.11.3	Reboot the client nodes . . . . .	22
4.12	Complete the Cluster Setup . . . . .	22
4.13	Test Cluster Setup . . . . .	22
4.14	Congratulations! . . . . .	22
4.15	Adding and Deleting client nodes . . . . .	23

4.15.1	Adding OSCAR clients . . . . .	23
4.15.2	Deleting clients . . . . .	23
4.16	Install/Uninstall OSCAR Packages . . . . .	23
4.16.1	Selecting the Right Package . . . . .	23
4.16.2	Single Image Restriction . . . . .	24
4.16.3	Failures to Install and Uninstall . . . . .	24
4.16.4	More Debugging Info . . . . .	24
4.17	Starting over – installing OSCAR again . . . . .	24

## Notice

This is a specialized version of OSCAR which has been packaged to include the SciDAC: Scalable System Software (SSS) components. The SSS-OSCAR release is based upon OSCAR v4.x. The documentation is generally the same except for the distribution support. See Section 3.3 for other SSS-OSCAR specific release notes.

## 1 Introduction

OSCAR version 2.0b3-v4.2.1b7 is a snapshot of the best known methods for building, programming, and using clusters. It consists of a fully integrated and easy to install software bundle designed for high performance cluster computing (HPC). Everything needed to install, build, maintain, and use a modest sized Linux cluster is included in the suite, making it unnecessary to download or even install any individual software packages on your cluster.

OSCAR is the first project by the Open Cluster Group. For more information on the group and its projects, visit its website <http://www.OpenClusterGroup.org/>.

This document provides a step-by-step installation guide for system administrators, as well as a detailed explanation of what is happening as you install. Note that this installation guide is specific to OSCAR version 2.0b3-v4.2.1b7.

### 1.1 Latest Documentation

Please be sure that you have the latest version of this document. It is possible (and probable!) that newer versions of this document were released on the main OSCAR web site after the software was released. You are *strongly* encouraged to check <http://oscar.sourceforge.net/> for the latest version of these instructions before proceeding. Document versions can be compared by checking their version number and date on the cover page.

### 1.2 Terminology

A common term used in this document is *cluster*, which refers to a group of individual computers bundled together using hardware and software in order to make them work as a single machine.

Each individual machine of a cluster is referred to as a *node*. Within the OSCAR cluster to be installed, there are two types of nodes: *server* and *client*. A *server* node is responsible for servicing the requests of *client* nodes. A *client* node is dedicated to computation.

An OSCAR cluster consists of one server node and one or more client nodes, where all the client nodes [currently] must have homogeneous hardware. The software contained within OSCAR does support doing multiple cluster installs from the same server, but that process is outside the scope of this guide.

An *OSCAR package* is a set of files that is used to install a software package in an OSCAR cluster. An OSCAR package can be as simple as a single RPM file, or it can be more complex, perhaps including a mixture of RPM and other auxiliary configuration / installation files. OSCAR packages provide the majority of functionality in OSCAR clusters.

OSCAR packages fall into one of three categories:

- *Core packages* are required for the operation of OSCAR itself (mostly involved with the installer).
- *Included packages* are shipped in the official OSCAR distribution. These are usually authored and/or packaged by OSCAR developers, and have some degree of official testing before release.

- *Third party packages* are not included in the official OSCAR distribution; they are “add-ons” that can be unpacked in the OSCAR tree, and therefore installed using the OSCAR installation framework.

### 1.3 Supported Distributions

OSCAR has been tested to work with several distributions. Table 1 lists each distribution and version and specifies the level of support for each. In order to ensure a successful installation, most users should stick to a distribution that is listed as *Fully supported*.

Distribution and Release	Architecture	Status
Red Hat Enterprise Linux 3	x86	Fully supported
Red Hat Enterprise Linux 3	x86_64	Fully supported
Red Hat Enterprise Linux 3	ia64	Fully supported
Red Hat Enterprise Linux 4	x86	Fully supported
Red Hat Enterprise Linux 4	x86_64	Fully supported
Red Hat Enterprise Linux 4	ia64	Fully supported
Fedora Core 2	x86	Fully supported
Fedora Core 3	x86	Fully supported
Fedora Core 3	x86_64	Fully supported
Fedora Core 4	x86	Fully supported
Mandriva Linux 10.0	x86	Fully supported
Mandriva Linux 10.1	x86	Fully supported

Table 1: OSCAR supported distributions

Clones of supported distributions, especially open source rebuilds of RedHat Enterprise Linux such as CentOS and Scientific Linux, should work but are not officially tested. See the release notes (Section 3) for your distribution for known issues.

### 1.4 Minimum System Requirements

The following is a list of minimum system requirements for the OSCAR server node:

- CPU of i586 or above
- A network interface card that supports a TCP/IP stack
- If your OSCAR server node is going to be the router between a public network and the cluster nodes, you will need a second network interface card that supports a TCP/IP stack
- At least 4GB total free space – 2GB under / and 2GB under /var
- An installed version of Linux, preferably a *Fully supported* distribution from Table 1

The following is a list of minimum system requirements for the OSCAR client nodes:

- CPU of i586 or above
- A disk on each client node, at least 2GB in size (OSCAR will format the disks during the installation)

- A network interface card that supports a TCP/IP stack<sup>1</sup>
- Same Linux distribution and version as the server node
- All clients must have the same architecture (e.g., ia32 vs. ia64)
- Monitors and keyboards may be helpful, but are not required
- Floppy or PXE enabled BIOS

## 1.5 Document Organization

Due to the complicated nature of putting together a high-performance cluster, it is strongly suggested that even experienced administrators read this document through, without skipping any sections, and then use the detailed installation procedure to install your OSCAR cluster. Novice users will be comforted to know that anyone who has installed and used Linux can successfully navigate through the OSCAR cluster install.

The rest of this document is organized as follows. First, Section 2 tells how to obtain an OSCAR version 2.0b3-v4.2.1b7 distribution package. Next, the “Release Notes” section (Section 3) that applies to OSCAR version 2.0b3-v4.2.1b7 contains some requirements and update issues that need to be resolved before the install. Section 4 details the cluster installation procedure (the level of detail lies somewhere between “the install will now update some files” and “the install will now replace the string ‘xyz’ with ‘abc’ in file `some_file`.”)

More information is available on the OSCAR web site and archives of the various OSCAR mailing lists. If you have a question that cannot be answered by this document (including answers to common installation problems), be sure to visit the OSCAR web site:

<http://oscar.sourceforge.net/>

## 2 Downloading an OSCAR Distribution Package

The OSCAR distribution packages can be downloaded from:

<http://oscar.sourceforge.net/>

Note that there are actually three flavors of distribution packages for OSCAR, depending on your bandwidth and installation/development needs:

1. “Regular”: All the OSCAR installation material that most users need to install and operate an OSCAR cluster.
2. “Extra Crispy”: Same as Regular, except that the SRPMs for most of the RPMs in OSCAR are also included. SRPMs are *not* required for installation or normal operation of an OSCAR cluster. This distribution package is significantly larger than Regular, and is not necessary for most users – only those who are interested in source RPMs need the Extra Crispy distribution. The SRPMs can be found under `packages/* /SRPMS/` directories.

---

<sup>1</sup>Beware of certain models of 3COM cards – not all models of 3COM cards are supported by the installation Linux kernel that is shipped with OSCAR. See the OSCAR web site for more information.

3. “Secret Sauce”: This distribution contains *only* the SRPMs for RPMs in OSCAR in the Regular and Extra Crispy distributions (it’s essentially (Extra Crispy - Regular)). It is intended only for those who initially downloaded the Regular distribution and later decided that they wanted the SRPMs as well. The Secret Sauce distribution is intended to be expanded over your Regular installation – it will create `packages/* /SRPMS/` directories and populate them with the relevant `.src.rpm` files.

All three distributions can be downloaded from the main OSCAR web page.

## 3 Release Notes

The following release notes apply to OSCAR version 2.0b3-v4.2.1b7.

### 3.1 Release Features

- x86\_64 support for Fedora Core 3
- x86 support for Fedora Core 4
- Manpath issues with C3 [1346603] fixed
- Bugs with Ganglia setup test [1346598] fixed
- WizardEnv::update\_env() causes Invalid argument on RHEL3u5 [1399567] fixed
- update-rpms picks wrong version [1403944] fixed
- python-elementtree in Scientific Linux 4.2 [1373984] fixed
- Failure to create client image with RHEL3 update 6 [1449837] fixed

### 3.2 Notes for All Systems

- The OSCAR installer GUI provides little protection for user mistakes. If the user executes steps out of order, or provides erroneous input, Bad Things may happen. Users are strongly encouraged to closely follow the instructions provided in this document.
- Each package in OSCAR has its own installation and release notes. See the full Installation Guide for these notes.
- All nodes must have a hostname other than “localhost” that does not contain any underscores (“\_”) or periods (“.”). Some distributions complicate this by putting a line such as the following in `/etc/hosts`:

```
127.0.0.1 localhost.localdomain localhost
yourhostname.yourdomain yourhostname
```

If this occurs the file should be separated as follows:

```
127.0.0.1 localhost.localdomain localhost
192.168.0.1 yourhostname.yourdomain yourhostname
```

- A domain name must be specified for the client nodes when defining them.
- Although OSCAR can be installed on pre-existing server nodes, it is typically easiest to use a machine that has a new, fresh install of a distribution listed in Table 1 *with no updates installed*. If the updates are installed, there may be conflicts in RPM requirements. It is recommended to install Red Hat updates *after* the initial OSCAR installation has completed.
- The “Development Tools” packages for are not default packages in all distributions and are required for installation.
- The following benign warning messages will appear multiple times during the OSCAR installation process:

```
awk: cmd. line:2: fatal: cannot open file `/etc/fstab'
  for reading (No such file or directory)

rsync_stub_dir: no such variable at ...

Use of uninitialized value in pattern match (m//) at
/usr/lib/perl5/site_perl/oda.pm ...
```

It is safe to ignore these messages.

- The OSCAR installer will install the MySQL package on the server node if it is not already installed. A random password will be automatically generated for the oscar user to access the oscar database. This password will be stored in the file `/etc/odapw`. It should not be needed by other users.
- The OSCAR installer GUI currently does not support deleting a node and adding the same node back *in the same session*. If you wish to delete a node and then add it back, you must delete the node, close the OSCAR installer GUI, launch the OSCAR installer GUI again, and then add the node.
- If `ssh` produces warnings when logging into the compute nodes from the OSCAR head node, the C3 tools (e.g., `cexec`) may experience difficulties. For example, if you use `ssh` to login in to the OSCAR head node from a terminal that does not support X windows and then try to run `cexec`, you might see a warning message in the `cexec` output:

```
Warning: No xauth data; using fake authentication data for
X11 forwarding.
```

Although this is only a warning message from `ssh`, `cexec` may interpret it as a fatal error, and not run across all cluster nodes properly (e.g., the **<Install/Uninstall Packages>** button will likely not work properly).

Note that this is actually an `ssh` problem, not a C3 problem. As such, you need to eliminate any warning messages from `ssh` (more specifically, eliminate any output from `stderr`). In the example above, you can tell the C3 tools to use the “-x” switch to `ssh` in order to disable X forwarding:

```
# export C3_RSH='ssh -x'
# cexec uptime
```



The warnings about `xauth` should no longer appear (and the **<Install/Uninstall Packages>** button should work properly).

- The SIS multicast facility (Flamethrower) is “experimentally” supported. If you are having problems with multicast and would like to experiment please check the `oscar-users` and/or `sisuite-users` mailing lists for tips.
- Due to some distribution portability issues, OSCAR currently installs a “compatibility” (`python2-compat-1.0-1`) RPM to resolve the Python2 prerequisite that is slightly different across different Linux distributions. Also see the file `packages/c3/RPMS/NOTE.python2`.
- *FutureWarning* message during APItests on Python2.3 based systems. The following is a warning message about the for the version of TwistedMatrix used by the APItest tool. It is only a warning and can be ignored.

```
Running Installation tests for pvm
/usr/lib/python2.3/site-packages/twisted/internet/defer.py:398:
FutureWarning: hex()/oct() of negative int will return
a signed string in Python 2.4 and up return "<%s at %s>"
% (cname, hex(id(self)))
```

- In some cases, the test window that is opened from the OSCAR wizard may close suddenly when there is a test failure. If this happens, run the test script, `testing/test_cluster`, manually in a shell window to diagnose the problem.
- OSCAR uses the `atftp-server` package instead of the `tftp-server` package which ships with many distributions. The `tftp-server` RPM should be uninstalled, along with any packages which depend on it before the OSCAR wizard is started. Using the standard or “workstation” installation should avoid this issue.
- The `pfilter` package may cause image deployment to freeze midway through, so if you are encountering this problem and have the package installed, you can turn it off prior to imaging your nodes via PXE-boot and/or BootCD. To turn off `pfilter`, run the following on the command prompt as root: `/etc/init.d/pfilter stop`. OSCAR will restart the service during a later step. An alternate solution is to update the kernel on the headnode to the latest version that is available for your installed Linux distribution.
- In some distributions, when using `cshell`, the `/etc/profile.d/less.csh` script expects to find `/etc/sysconfig/i18n`. If it’s missing, as it is on every OSCAR compute node, `ssh` or `login` generates an error. To correct the error, simply use “`cpush /etc/sysconfig/i18n`” after install and updating affected images. Fedora Core 4 and RHEL4U2 should not be affected by this issue.
- When using the **<Install/Uninstall OSCAR Packages>** feature, a RPM cache needs to be first initialized. This will only take a few minutes and only happens the first time this functionality is used.

### 3.3 SSS-OSCAR Release Notes

This is a specialized version of OSCAR which has been packaged to include the SciDAC: Scalable System Software (SSS) components. The SSS-OSCAR release is based upon OSCAR v4.x. Therefore the docu-

mentation is the same, except for the supported distributions. This release, SSS-OSCAR v2.0b3-v4.2.1b7, has been limited to Fedora Core 2 (x86).<sup>2</sup>

For more information on the SciDAC: Scalable Systems Software (SSS) project please visit, <http://www.scidac.org/ScalableSystems>. Information on SSS-OSCAR is available at <http://sss-oscar.sourceforge.net>.

The following SSS-OSCAR specific release notes supersede any conflicts that might occur in latter portions of this documentation.

- This release of SSS-OSCAR v2.0b3-v4.2.1b7 is limited to Fedora Core 2 on x86. (*The following sections which detail standard OSCAR have details for the general OSCAR v4.x release and are not to be confused with this SSS-OSCAR release.*)
- The current set of packages included in SSS-OSCAR are configured to work together; some additional packages available on the OPD repositories will conflict with SSS-OSCAR included packages. Those known to effect the default SSS-OSCAR configuration are:  
pbs torque maui lam (non-sss-oscar release)
- Some tests have stalled/hung during **<Step3: Install OSCAR Server Packages>** when trying to start NFS. Starting/restarting the 'portmap' service fixes the problem, e.g.,  
service portmap restart
- Due to some differences with standard PBS and the Bamboo & friends tools used with SSS-OSCAR, some of the test scripts are SKIPPED. More specifically, any OSCAR Package test that uses a 'test\_user' script, which makes use of the 'pbs\_test' helper script, will be flagged as SKIPPED. This will be fixed in a future release. (This issue is known to effect LAM/MPI, MPICH & PVM.)
- The following packages were removed from the stock OSCAR package set:  
maui pbs lam  
This was due to either an alternate version supplied with SSS-OSCAR or because of conflicts/errors.
- Occasionally on the first invocation of the 'install\_cluster' script, an error occurs related to the initialization of the OSCAR Database (ODA) that causes the script to stop. If this occurs, simply re-run the script and it should startup properly.
- During **<Step 7: Complete Cluster Setup>** some services that are restarted print usage errors when stopping the service. This is generally not a problem and can be ignored. Example,

```
Stopping Event Manager: cat: /var/run/sss\_em.pid: No such file or
directory
kill: usage: kill [-s sigspec | -n signum | -sigspec] [pid | job]...
or kill -l [sigspec]
done
```

- Warehouse: If the test script fails, try manually restarting Warehouse's client and server services by typing the following command as shown here (in this order):

---

<sup>2</sup>The version string indicates both the SSS version as well as the OSCAR version used for the release. For example, "sss-oscar-1.2b1-v4.1" is SSS beta v1.2b1 and OSCAR stable v4.1.

```
[root@headnode]# /etc/init.d/warehouse_SysMon stop \
    && cexec -p /etc/init.d/warehouse_node stop

[root@headnode]# cexec -p /etc/init.d/warehouse_node start \
    && /etc/init.d/warehouse_SysMon start
```

- If trying to work directly from CVS, the 'make\_dist.pl' script should be helpful in pooling together the necessary files. It creates a tarball that can be used for testing.

### 3.4 Red Hat Enterprise Linux 4 & Fedora Core 3 & 4: SELinux Conflict

- Due to issues with displaying graphs under Ganglia, and installing RPMs in a SIS chroot environment (needed to build OSCAR images), SELinux should be disabled before installing OSCAR. During installation, it can be deactivated on the same screen as the firewall. If it is currently active it can be turned off in /etc/selinux/config by setting SELINUX to "disabled" and rebooting.

When installation fails due to SELinux, it displays a few messages like the following:

```
error: %pre(coreutils-5.2.1-31.1.x86_64) scriptlet failed,
    exit status 255
error: install: %pre scriptlet failed (2), skipping
    coreutils-5.2.1-31.1
```

### 3.5 Red Hat Enterprise Linux 4

- On Red Hat Enterprise Linux 4 Update 2+ and CentOS 4.2+ systems you must uninstall the 'tog-pegasus' and 'tog-pegasus-devel' RPMs from the server, if installed, to avoid file ownership conflicts with 'pvm' on the compute nodes and SIS image. Use the command:

```
# rpm --erase tog-pegasus tog-pegasus-devel
```

This is due to an ordering issue with the RPM installs which both create system-level users which get out of ordering during OSCAR's image build process. See also:

- OSCAR Bug#1368604 "/opt/pvm3 directory ownership on image"
- OSCAR Bug#1371143 "docfix: tog-pegasus / pvm permission issues"

- On ia64 systems using Red Hat Enterprise Linux 4, it may be possible that proc remains mounted on the image after it has been built; this causes subsequent compute node installations (rsync) to fail. To get around this problem, you need to umount the directory manually after image has been created and before installing the compute nodes.

First, find out what is mounted by looking in /proc/mounts with

```
# cat /proc/mounts
```

It should list something like the following:

```
proc /var/lib/systemimager/images/oscarimage/proc proc rw,nodiratime 0 0
none /var/lib/systemimager/images/oscarimage/proc/sys/fs/binfmt_misc
binfmt_misc rw 0 0
```

Then unmount the two directories that are mounted, e.g.,

```
# umount -f /var/lib/systemimager/images/oscarimage/proc/sys/fs/binfmt_misc
# umount -f /var/lib/systemimager/images/oscarimage/proc"
```

Note the ordering of the unmount commands is important.

- On ia64 systems using Red Hat Enterprise Linux 4, the `ia32el.rpm` is needed. This rpm is located on the extra CDs. Make sure you include the RPMs from these disks when you copy the RPMs into `/tftpboot/rpm`.

### 3.6 CentOS 4 ia64 Notes

These items are specific to CentOS 4. The items listed above for Red Hat Enterprise Linux 4 also apply.

- The `ia32el` RPM is missing from CentOS, which prevents the successful building of disk images on these systems. To work around this, we suggest downloading the RPM from this site <ftp://ftp.scientificlinux.org/linux/scientific/41/ia64/SL/RPMS/ia32el-1.2-4.ia64.rpm> and placing it in the `/tftpboot/rpm` directory. This package contains the ia32 execution layer needed for running ia32 applications and is included in most other Red Hat based distributions by default.

### 3.7 Red Hat Enterprise Linux 3

- When creating an OSCAR image on Red Hat Enterprise Linux 3 Update 6 or 7 on x86\_64 hardware, it is necessary to manually choose the `redhat-3asU6-x86_64.rpmlist` otherwise the build process will error out. This is caused by a bug which prevents update-rpms from installing the i386 compat RPMs correctly. Since update-rpms will be deprecated in future (5.0) releases, a workaround is discussed here. The above mentioned rpmlist does not contain the i386 compat RPMs but they are installed after the image has been created by executing the `rhel3u6_fixup` script located in the `OSCARTHOME/scripts` directory. This will install the i386 compat RPMs into the image:

```
OSCARTHOME/scripts/rhel3u6_fixup <image name>
```

- The `efibootmgr` (from package `elilo`) isn't able to deal with the efi variables in `/sys` (as in the new installation kernel). It expects to find them in `/proc`. This means that at the end of the installation `systemconfigurator` isn't able to add the necessary entry to the efi boot menu. The system will not boot and reinstall itself indefinitely.

To fix this issue download the RPM <ftp://ftp.scientificlinux.org/linux/scientific/41/ia64/SL/RPMS/elilo-3.4-10.ia64.rpm>.

Unpack its contents to some subdirectory:

```
mkdir t
cd t
rpm2cpio ../elilo-3.4-10.ia64.rpm | cpio -i -d
```

Replace \$IMAGEDIR/usr/sbin/efibootmgr with the file from the new elilo rpm: cp usr/sbin/efibootmgr \$IMAGEDIR/usr/sbin

Install the node.

### 3.8 Mandriva Linux (previously known as "Mandrake Linux") Notes

The versions of Mandriva Linux officially supported in OSCAR 4.2 are Mandriva Linux 10.0 Official (3 cd set) and Mandriva Linux 10.1 Official, PowerPack edition (6 cd set). While it is possible to install OSCAR 4.2 under others editions of those versions of Mandriva Linux, the above mentioned are those officially supported by the OSCAR development team. Please read carefully the sub-items bellow and follow the installation instructions with attention.

- Common notes

When openssh-server is installed in Mandriva systems (during Step 3) its config file is usually set to "NOT permit root login" by default. Since OSCAR needs to be able to login as root via ssh, you have to edit the file /etc/ssh/sshd\_config whether by setting 'PermitRootLogin yes' or by commenting/deleting this entire line. This must be done before Stage 4 because this configuration file will be copied to the node's image and spread to all nodes during step 6. Remember to reload the openssh server in the master node after this change:

```
service sshd reload
```

- Mandriva 10.0 Official specific notes

During Step 3, <Install OSCAR Server Packages>, the RPM package kernel-enterprise will be selected to be installed on the head node. As a result, this kernel will be used upon subsequent reboot.

While Mandriva Linux 10.0 supports 2.4 kernel, OSCAR only supports 2.6 kernel - this is explicitly specified in oscarsamples/mandrake-10.0-i386.rpmlist.

The tftp-server RPM which came with Mandriva Linux 10.0 expects the pxelinux boot files to reside in /var/lib/tftpboot but OSCAR puts them in /tftpboot. To get around this issue a symbolic link is created from /var/lib/tftpboot to /tftpboot.

- Mandriva 10.1 Official specific notes

As mentioned above, the officially supported edition of Mandriva Linux 10.1 is the PowerPack edition. However, the 3 cd set "Download Edition" should work if the following packages are added to the repository of packages (/tftpboot/rpm):

```
perl-PerlIO-gzip
libecpg3
perl-devel
php-xml
gcc-g77
libf2c0
zlib1-devel
```

Those packages are not present in the distribution cds and you will need to download them from a FTP mirror (see <http://www1.mandrivalinux.com/en/ftp.php3> for a list of mirrors).

The OpenSSH packages that came with Mandriva Linux 10.1 Official (version 3.9p1-3mdk) have a bug concerning public keys under user's home directory shared by NFS. That means OSCAR won't work if we install OpenSSH from this source (version 3.9p1-3mdk). For now, it is necessary to delete the following three packages that were copied with the others under /tftpboot/rpm, get a copy of these same packages provided with Mandriva Linux 10.0 (version 3.6.1p2-12mdk) from a FTP miroir (see link for a list of mirroirs above) and copy them to /tftpboot/rpm, replacing the original packages:

```
openssh
openssh-clients
openssh-server
```

### 3.9 Fedora Core 4

- Under Fedora Core 4, the following output may be ignored during Step 5, <Define OSCAR Clients>:

```
Argument "2.121_02" isn't numeric in subroutine entry at
/usr/lib/perl5/site_perl/MLDBM/Serializer/Data/Dumper.pm line 5
```

## 4 Quick Cluster Installation Procedure

All actions specified below should be performed by the `root` user on the server node unless noted otherwise. Note that if you login as a regular user and use the `su` command to change to the `root` user, you *must* use “`su -`” to get the full `root` environment. Using “`su`” (with no arguments) is not sufficient, and will cause obscure errors during an OSCAR installation.

Note that all the steps below are mandatory unless explicitly marked as optional.

Finally, note that this is specifically the “Quick” OSCAR installation guide. There are a *lot* of details and explanations that are left out. If you run into any problems or have any questions, please consult the complete OSCAR Installation Guide.

### 4.1 Server Installation and Configuration

During this phase, you will prepare the machine to be used as the server node in the OSCAR cluster.

#### 4.1.1 Install Linux on the server machine

If you have a machine you want to use that already has Linux installed, ensure that it meets the minimum requirements as listed in Section 1.4. If it does, you may skip ahead to Section 4.1.2.

It should be noted that OSCAR is only supported on the distributions listed in Table 1 (page 5). As such, use of distributions other than those listed will likely require some porting of OSCAR, as many of the scripts and software within OSCAR are dependent on those distributions.

It is best to *not* install distribution updates after you install Linux; doing so may disrupt some of OSCAR's RPM dependencies. Instead, install OSCAR first, and then install the distribution updates.

### 4.1.2 Disk space and directory considerations

OSCAR has certain requirements for server disk space. Space will be needed to store the Linux RPMs and to store the images. The RPMs will be stored in `/tftpboot/rpm`. 2GB is usually enough to store the RPMs. The images are stored in `/var/lib/systemimager` and will need approximately 2GB per image. Although only one image is required for OSCAR, you may want to create more images in the future.

### 4.1.3 Download a copy of OSCAR and unpack on the server

If you are reading this, you probably already have a copy of an OSCAR distribution package. If not, go to <http://oscar.sourceforge.net/> and download the latest OSCAR Regular or Extra Crispy distribution package (see Section 2, page 6). Ensure that you have the latest documentation (later documentation may be available on the OSCAR web site than in the OSCAR distribution package).

Place the OSCAR distribution package in a directory such as `root`'s home directory on the server node. Although there is no required installation directory (note that you may not use the directory `/usr/local/oscar`, `/opt/oscar`, `/var/lib/oscar`, or `/var/cache/oscar` – they are reserved for use by OSCAR), the rest of these instructions will assume that you downloaded the OSCAR distribution package to `root`'s home directory.

### 4.1.4 Configure and Install OSCAR

Starting with OSCAR 2.3, after unpacking the tarball you will need to configure and install OSCAR. The configure portion sets up the release to be permanently installed on the system in `/opt/oscar` (default location). The `--prefix=ALT-DIR` flag can be used to configure and install OSCAR into an alternate directory.

The first step is to run the `configure` script, which is provided in the top-level directory of the OSCAR release.

```
# cd /root/oscar-2.0b3-v4.2.1b7
# ./configure
```

Once the configure script successfully completes you are ready to actually install OSCAR on the server. At this point no changes have been made to the system beyond the `oscar-2.0b3-v4.2.1b7` directory itself. Running the following will copy the files to the system. The files copied will include the base OSCAR toolkit as well as startup scripts for the `profile.d/` area. The startup scripts add OSCAR to your path and set environment variables like `OSCAR_HOME`.

```
# make install
```

At this point you will be ready to change to either the default `/opt/oscar` directory or whatever path was use with the `--prefix` flag to perform the installation steps discussed in Section 4.2.

In the remainder of this document, the variable `$OSCAR_HOME` will be used in place of the directory you installed OSCAR to – by default this is `/opt/oscar`.

### 4.1.5 Configure the ethernet adapter for the cluster

Assuming you want your server to be connected to both a public network and the private cluster subnet, you will need to have two ethernet adapters installed in the server. This is the preferred OSCAR configuration

because exposing your cluster may be a security risk and certain software used in OSCAR (such as DHCP) may conflict with your external network.

Once both adapters have been physically installed in the server node, you need to configure them.<sup>3</sup> Any network configurator is sufficient; popular applications include `neat`, `netcfg`, or a text editor.

The following major requirements need to be satisfied:

**Hostname.** Most Linux distributions default to the hostname “localhost” (or “localhost.localdomain”). This must be changed in order to successfully install OSCAR – choose another name that does not include any underscores (“\_”). This may involve editing `/etc/hosts` by hand as some distributions hide the lines involving “localhost” in their graphical configuration tools. Do not remove all reference to “localhost” from `/etc/hosts` as this will cause no end of problems.

For example if your distribution automatically generates the `/etc/hosts` file:

```
127.0.0.1 localhost.localdomain localhost
yourhostname.yourdomain yourhostname
```

This file should be separated as follows:

```
127.0.0.1 localhost.localdomain localhost
192.168.0.1 yourhostname.yourdomain yourhostname
```

Additional lines may be needed if more than one network adapter is present.

**Public adapter.** This is the adapter that connects the server node to a public network. Although it is not required to have such an adapter, if you do have one, you must configure it as appropriate for the public network (you may need to consult with your network administrator).

**Private adapter.** This is the adapter connected to the TCP/IP network with the rest of the cluster nodes. This adapter must be configured as follows:

- Use a private IP address
- Use an appropriate netmask
- Ensure that the interface is activated at boot time
- Set the interface control protocol to “none”

#### 4.1.6 Copy distribution installation RPMs to `/tftpboot/rpm`

In this step, you need to copy all the RPMs included with your Linux distribution into the `/tftpboot/rpm` directory. Most distributions have more RPMs than are generally necessary for installation, it is important that the RPMs from all the installation discs are included. When each CD is inserted, Linux usually automatically makes the contents of the CD be available in the `/mnt/cdrom` directory (you may need to execute a command such as “`mount /mnt/cdrom`” if the CD does not mount automatically).

---

<sup>3</sup>Beware of certain models of 3COM network cards. See footnote 1 on page 6.



## 4.2 Launching the OSCAR Installer

Run the `install_cluster` script from OSCAR’s top-level directory, with an argument specifying your server’s private network adapter. For example: `./install_cluster eth0`.

A lot of output will be displayed in the console window where you invoked `install_cluster`. This reflects normal operational output from the various installation commands that OSCAR executes. The output is also saved in the file `oscarinstall.log` for later reference (particularly if something goes wrong during the installation).

After the steps listed above have successfully finished, the OSCAR installation wizard GUI will automatically be launched.

## 4.3 Downloading Additional OSCAR Packages

*Note: This step is optional.*

The first step of the Wizard, “Step 0”, enables you to download additional packages. The OSCAR Package Downloader (OPD) is a unified method of downloading OSCAR packages and inserting them in the OSCAR installation hierarchy so that they will be installed during the main OSCAR installation process. The Wizard uses a GUI frontend to OPD affectionately known as *OPDer*. The addition of this frontend limits the need for direct access to the command-line OPD tool directly.<sup>4</sup> If you would like to add additional repository URLs for testing purposes, you can do this by accessing the **<File>** menu and then choosing **<Additional Repositories...>**. The remainder of this sub-section describes the underlying OPD tool.

The command-line OPD can be launched outside of the GUI Wizard by running the command `opd` from the top-level OSCAR `scripts` directory.

## 4.4 Selecting Packages to Install

*Note: This step is optional.*

If you wish to change the list of packages that are installed, click on the **<Select OSCAR Packages To Install>** button. This step is optional – by default all packages directly included in OSCAR are selected and installed. However, if you downloaded any additional packages, e.g., via OPD/OPDer, they will not be selected for installation by default. Therefore you will need to click this button and select the appropriate OSCAR Packages to install on the cluster.

## 4.5 Configuring OSCAR Packages

*Note: This step is optional.*

Some OSCAR packages allow themselves to be configured. Clicking on the **<Configure Selected OSCAR Packages>** button will bring up a window listing all the packages that can be configured.

### 4.5.1 Selecting a Default MPI Implementation

Although multiple MPI implementations can be installed, only one can be “active” for each user at a time. Specifically, each user’s path needs to be set to refer to a “default” MPI that will be used for all commands.

---

<sup>4</sup>Note, if using OPD directly, the packages must be downloaded before the **<Select OSCAR Packages To Install>** step (see Section 4.4).

The Environment Switcher package provides a convenient mechanism for switching between multiple MPI implementations.

The Environment Switcher package is mentioned now, however, because its configuration panel allows you to select which MPI implementation will be the initial “default” for all users. OSCAR currently includes two MPI implementations: LAM/MPI and MPICH. Using Environment Switcher’s configuration panel, you can select one of these two to be the cluster’s default MPI.

## 4.6 Install OSCAR Server Packages

Press the **<Install OSCAR Server Packages>** button. This will invoke the installation of various RPMs and auxiliary configuration on the server node. Execution may take several minutes; text output and status messages will appear in the shell window.

## 4.7 Build OSCAR Client Image

Before pressing the **<Build OSCAR Client Image>**, ensure that the following conditions on the server are true:

- Ensure that the SSH daemon’s configuration file (`/etc/ssh/sshd_config`) on the headnode has `PermitRootLogin` set to `yes`. After the OSCAR installation, you may set this back to `no` (if you want), but it needs to be `yes` during the install because the config file is copied to the client nodes, and `root` *must* be able to login to the client nodes remotely.
- By the same token, ensure that TCP wrappers settings are not “too tight”. The `/etc/hosts.allow` and `/etc/hosts.deny` files should allow all traffic from the entire private subnet.
- Also, beware of firewall software that restricts traffic in the private subnet.

If these conditions are not met, the installation may fail during this step or later steps.

Press the **<Build OSCAR Client Image>** button. A dialog will be displayed. In most cases, the defaults will be sufficient. You should verify that the disk partition file is the proper type for your client nodes. The sample files have the disk type as the last part of the filename. You may also want to change the post installation action and the IP assignment methods. **It is important to note that if you wish to use automatic reboot, you should make sure the BIOS on each client is set to boot from the local hard drive before attempting a network boot by default. If you have to change the boot order to do a network boot before a disk boot to install your client machines, you should not use automatic reboot.**

**Customizing your image.** The defaults of this panel use the sample disk partition and RPM package files that can be found in the `oscarsamples` directory. You may want to customize these files to make the image suit your particular requirements.

**Disk partitioning.** The disk partition file contains a line for each partition desired, where each line is in the following format:

```
<partition> <size in megabytes> <type> <mount point> <options>
```

**Package lists.** The package list is simply a list of RPM file names (one per line). Be sure to include all prerequisites that any packages you might add. You do not need to specify the version, architecture, or extension of the RPM filename. For example, `bash-2.05-8.i386.rpm` need only be listed as “`bash`”.

**Custom kernels.** If you want to use a customized kernel, you can add it to the image after it is built (after installing the server OSCAR packages, but before building the client image).

**Build the Image.** Once you are satisfied with the input, click the <**Build Image**> button. When the image completes, a popup window will appear indicating whether the build succeeded or failed. If successful, click the <**Close**> button to close the popup, and then press the <**Close**> button on the build image window. You will be back at the main OSCAR wizard menu.

## 4.8 Define OSCAR Clients

Press the <**Define OSCAR Clients**> button. In the dialog box that is displayed, enter the appropriate information. Although the defaults will be sufficient for most cases, you will need to enter a value in the *Number of Hosts* field to specify how many clients you want to create.

1. The *Image Name* field should specify the image name that was used to create the image in the previous step.
2. The *Domain Name* field should be used to specify the client's IP domain name. It should contain the server node's domain (if it has one); if the server does not have a domain name, the default name `oscardomain` will be put in the field (although you may change it). **This field must have a value – it cannot be blank.**
3. The *Base name* field is used to specify the first part of the client name and hostname. It will have an index appended to the end of it. **This name cannot contain an underscore character “\_” or a period “.”.**
4. The *Number of Hosts* field specifies how many clients to create. **This number must be greater than 0.**
5. The *Starting Number* specifies the index to append to the *Base Name* to derive the first client name. It will be incremented for each subsequent client.
6. The *Padding* specifies the number of digits to pad the client names, e.g., 3 digits would yield `oscardomain001`. The default is 0 to have no padding between base name and number (index).
7. The *Starting IP* specifies the IP address of the first client. It will be incremented for each subsequent client.

**IMPORTANT NOTE:** Be sure that the resulting range of IP addresses does *not* include typical broadcast addresses such as `X.Y.Z.255`! If you have more hosts than will fit in a single address range, see the note at the end of this section about how to make multiple IP address ranges.

8. The *Subnet Mask* specifies the IP netmask for all clients.
9. The *Default Gateway* specifies the default route for all clients.

Note that this step can be executed multiple times. The GUI panel that is presented has limited flexibility in IP address numbering – the starting IP address will only increment the least significant byte by one for each successive client. Hence, if you need to define more than 254 clients (beware of broadcast addresses!), you will need to run this step multiple times and change the starting IP address. There is no need to close

the panel and return to the main OSCAR menu before executing it again; simply edit the information and click on the **<Addclients>** button as many times as is required.

Additionally, you can run this step multiple times to use more structured IP addressing schemes. With a larger cluster, for example, it may be desirable to assign IP addresses based on the top-level switch that they are connected to. For example, the 32 clients connected to switch 1 should have an address of the form 192.168.1.x. The next 32 clients will be connected to switch 2, and should therefore have an address of the form 192.168.2.x. And so on.

## 4.9 Setup Networking

In order to collect the MAC addresses, press the **<Setup Networking>** button. The OSCAR network utility dialog box will be displayed. To use this tool, you will need to know how to network boot your client nodes, or have a file that lists all the MACs from your cluster.

If you need to collect the MACs in your cluster, start the collection by pressing the **<Collect MAC Address>** button and then network boot the first client. As the clients boot up, their MAC addresses will show up in the left hand window. You have multiple options for assigning MACs to nodes; you can either:

- manually select MAC address and the appropriate client in the right side window. Click **<Assign MAC to Node>** to associate that MAC address with that node.
- click **<Assign all MACs>** button to assign all the MACs in the left hand window to all the open nodes in the right hand window.

Some notes that are relevant to collecting MAC addresses from the network:

- The **<Dynamic DHCP Update>** checkbox at the bottom right of the window controls refreshing the DHCP server. If it is selected (the default), the DHCP server configuration will be refreshed each time a MAC is assigned to a node. Note that if the DHCP reconfiguration takes place quick enough, you may not need to reboot the nodes a second time (i.e., if the DHCP server answers the request quick enough, the node may start downloading its image immediately).

If this option is off, you will need to click the **<Configure DHCP Server>** (at least once) to give it the associations between MACs and IP addresses.

- You *must* click on the **<Stop Collecting MACs>** before closing the MAC Address Collection window!

This menu also allows you to choose your remote boot method. You should do one of two things after collecting your MAC addresses but before leaving the menu:

- The **<Build Autoinstall CD>** button will build an iso image for a bootable CD and gives a simple example for using the `cdrecord` utility to burn the CD. This option is useful for client nodes that do not support PXE booting. In a terminal, execute the command `cdrecord -scanbus` to list the drives `cdrecord` is aware of and their dev number. Use this trio of numbers in place of `dev=1,0,0` when you execute the command `cdrecord -v speed=2 dev=1,0,0 /tmp/oscar_bootcd.iso`.
- The **<Setup Network Boot>** button will configure the server to answer PXE boot requests if your client hardware supports it.

If your network switch supports multicasting, there is a new feature in OSCAR 3.0 which uses multi-cast to push files to the clients. To enable this feature simply click on the **<Enable Multicasting>** checkbox.

Once this feature is enabled, rsync will not be used for file distribution but instead by a program called Flamethrower which is bundled with SystemImager.

Since this new feature is still in its early stage, we recommend only the adventurous to try it as it may not work on all networking gear. Some have reported favorable results with the Force10 and HP 4000M network switches.

In case you cannot get this feature working and want to switch back to using rsync, simply go back to the **<Setup Networking>** menu, make sure that the **<Enable Multicasting>** checkbox is disabled (should be disabled by default) and then click on **<Configure DHCP Server>** - this should revert back to the default settings.

## 4.10 Monitor Cluster Deployment

*Note: This step is optional.*

During the client installation phase it is possible to click on the **<Monitor Cluster Deployment>** button to bring up the SystemImager monitor GUI. This window will display real-time status of nodes as they request images from the server machine and track their progress as they install.

MAC address	IP address	Hostname	RAM (MB)	RAM used	Time (min)	Status	Image	CPU
00.E0.81.30.DD.46	192.168.10.104	nec05	7971	1%	8	done	slesimg	Dual Core AMD Opteron(tm) Processor 275 (2210.211 MHz)
00.E0.81.30.DD.4A	192.168.10.101	nec02	7971	1%	8	done	slesimg	Dual Core AMD Opteron(tm) Processor 275 (2210.215 MHz)
00.E0.81.31.5F.68	192.168.10.108	nec09	7971	1%	8	done	slesimg	Dual Core AMD Opteron(tm) Processor 275 (2210.213 MHz)
00.E0.81.31.5F.9A	192.168.10.100	nec01	7971	1%	8	done	slesimg	Dual Core AMD Opteron(tm) Processor 275 (2210.214 MHz)
00.E0.81.31.5F.C2	192.168.10.103	nec04	7971	1%	8	done	slesimg	Dual Core AMD Opteron(tm) Processor 275 (2210.214 MHz)
00.E0.81.31.5F.CC	192.168.10.109	nec10	7971	1%	8	done	slesimg	Dual Core AMD Opteron(tm) Processor 275 (2210.211 MHz)
00.E0.81.31.68.E6	192.168.10.111	nec12	7971	1%	9	done	slesimg	Dual Core AMD Opteron(tm) Processor 275 (2210.214 MHz)
00.E0.81.31.69.08	192.168.10.118	nec19	5963	1%	3	0.93%	el4img	Dual Core AMD Opteron(tm) Processor 275 (2210.228 MHz)
00.E0.81.31.69.2A	192.168.10.115	nec16	3943	2%	7	77.32%	slesimg	Dual Core AMD Opteron(tm) Processor 275 (2210.213 MHz)
00.E0.81.31.69.2C	192.168.10.116	nec17	7971	1%	5	54.11%	el4img	Dual Core AMD Opteron(tm) Processor 275 (2210.213 MHz)
00.E0.81.31.69.7A	192.168.10.113	nec14	7971	1%	8	done	slesimg	Dual Core AMD Opteron(tm) Processor 275 (2210.210 MHz)
00.E0.81.31.69.AA	192.168.10.112	nec13	7971	1%	8	done	slesimg	Dual Core AMD Opteron(tm) Processor 275 (2210.215 MHz)
00.E0.81.31.69.C6	192.168.10.102	nec03	5963	1%	8	done	slesimg	Dual Core AMD Opteron(tm) Processor 275 (2210.222 MHz)
00.E0.81.31.74.52	192.168.10.107	nec08	7971	1%	8	done	slesimg	Dual Core AMD Opteron(tm) Processor 275 (2210.211 MHz)
00.E0.81.31.74.F8	192.168.10.106	nec07	7971	1%	8	done	slesimg	Dual Core AMD Opteron(tm) Processor 275 (2210.215 MHz)
00.E0.81.31.75.96	192.168.10.105	nec06	7971	1%	16	done	slesimg	Dual Core AMD Opteron(tm) Processor 275 (2210.217 MHz)
00.E0.81.31.8A.92	192.168.10.110	nec11	7971	1%	8	done	slesimg	Dual Core AMD Opteron(tm) Processor 275 (2210.214 MHz)
00.E0.81.31.8B.36	192.168.10.117	nec18	7971	1%	4	10.28%	el4img	Dual Core AMD Opteron(tm) Processor 275 (2210.204 MHz)
00.E0.81.31.8B.BA	192.168.10.114	nec15	7971	1%	8	done	slesimg	Dual Core AMD Opteron(tm) Processor 275 (2210.191 MHz)

Figure 1: Monitor Cluster Deployment

## 4.11 Client Installations

During this phase, you will network boot your client nodes and they will automatically be installed and configured.

### 4.11.1 Network boot the client nodes

Network boot all of your clients. As each machine boots, it will automatically start downloading and installing the OSCAR image from the server node.

### 4.11.2 Check completion status of nodes

After several minutes, the clients should complete the installation. You can use the **<Monitor Cluster Deployment>** functionality to monitor the progress. Depending on the Post Installation Action you selected when building the image, the clients will either halt, reboot, or beep incessantly when the installation is completed.

### 4.11.3 Reboot the client nodes

After confirming that a client has completed its installation, you should reboot the node from its hard drive. If you chose to have your clients reboot after installation, they will do this on their own. If the clients are not set to reboot, you must manually reboot them. The filesystems will have been unmounted so it is safe to simply reset or power cycle them.

**Note: If you had to change the BIOS boot order on the client to do a network boot before booting from the local disk, you will need to reset the order to prevent the node from trying to do another network install.**

## 4.12 Complete the Cluster Setup

**Ensure that all client nodes have fully booted before proceeding with this step.**

Press the **<Complete Cluster Setup>** button. This will run the final installation configurations scripts from each OSCAR software package, and perform various cleanup and re-initialization functions. This step can be repeated should networking problems or other types of errors prevent it from being successful the first time.

## 4.13 Test Cluster Setup

A simplistic test suite is provided in OSCAR to ensure that the key cluster components (OpenSSH, Torque, MPI, PVM, etc.) are functioning properly.

Press the **<Test Cluster Setup>** button. This will open a separate window to run the tests in. The cluster's basic services are checked and then a set of `root` and user level tests are run.

## 4.14 Congratulations!

Your cluster setup is now complete. Your cluster nodes should be ready for work.

It is *strongly* recommended that you read the package-specific installation notes in the main Installation Guide. It contains vital system administrator-level information on several of the individual packages that were installed as part of OSCAR.

## 4.15 Adding and Deleting client nodes

This section describes the steps need when it becomes necessary to add or delete client nodes. If you have already built your cluster successfully and would like to add or delete a client node, execute the following from the top-level OSCAR directory:

```
# ./install_cluster <device>
```

### 4.15.1 Adding OSCAR clients

Press the button of the wizard entitled **<Add OSCAR Clients>**. These steps should seem familiar – they are same as the initial install steps. Refer to Sections 4.8, 4.9, and 4.12.

Note that when adding nodes, in the Defining OSCAR Clients step, you will typically need to change the following fields to suit your particular configuration:

- Number of hosts
- Starting number
- Starting IP

### 4.15.2 Deleting clients

Press the button of the wizard entitled **<Delete OSCAR Clients>**. Select the node(s) that you wish to delete and press the button **<Delete clients>**, then press **<Close>**.

## 4.16 Install/Uninstall OSCAR Packages

The package installation system is designed to help simplify adding and removing OSCAR packages after the initial system installation. The system is fairly straightforward for the user as packages are selected for the appropriate action from the wizard.

### 4.16.1 Selecting the Right Package

Because of the way OSCAR handles packages, two packages of the same name cannot currently coexist nicely on the system. Packages can be in one of two spots:

1. `$OSCAR_HOME/packages` – OSCAR installation directory
2. `/var/lib/oscar/package` – OPD download area

If there are multiple packages on the system of the same name, the package located in the OPD download area is the package that the system recognizes. It is possible that a package placed in `/var/lib/oscar/package` via a download or manually could get loaded into the database, while the original package that is located in `$OSCAR_HOME/packages` is the one that is actually installed. That is why this system does not support upgrades. That is planned for a future OSCAR release. This can be avoided by uninstalling the original package first before you download anything and then re-running the wizard. When the wizard first runs (`./install_cluster ethX`), the XML files are re-read for all packages and the database re-initialized.

### 4.16.2 Single Image Restriction

Currently, if more than one image is detected on your system, the package install/uninstall system will not run. Packages are installed and uninstalled to the clients (compute nodes), the server, and an image (singular). This release does not support the mapping of nodes to an image for package install/uninstall through the wizard. This single image restriction is instituted to help reduce the risk of error with this initial release, i.e., "keep it simpler". The system looks for images in `/var/lib/systemimager/images` and no other place.

### 4.16.3 Failures to Install and Uninstall

A package is not installed or uninstalled until it successfully installs or uninstalls on the compute nodes, the server, and a single image. We will run through an example problem as the best way of describing how to debug the system. As a note, the system does sanity checking to try to avoid the following situation.

For example, during the install process the clients may have been successfully installed, but the server installation died half way through for some unknown reason, and the install program exits with an error. If this happens, a second attempt to install will probably fail even if the server's problem is resolved.

The reason for this is that something happened on the clients and it succeeded, probably RPM's were installed. So the second time you go to install the package, the `rpm -Uvh` command on the nodes will fail.

So, what do you do?

Really, you have two options. The first is to look back through the log to see what commands were run, and undo those commands by hand. The second would be to manually push out the uninstall scripts to the compute nodes, and run them outside of the wizard environment. All scripts are re-runnable in OSCAR, and for several good reasons.

These options are not good ones, and we recognize this fact. We are working on better tools for general use. Generally, it is a difficult problem to judge error codes on remote systems and try to guess the appropriate actions to take. We have developed some additional software to start solving this problem, but there is still work that needs to be done in this area.

On the uninstall side of things, the picture is much simpler. The only action taken by the system is to run the package's uninstall script and judge the return code. So an uninstall failure is probably a faulty uninstall script.

Note if you hit the **<Cancel>** button or close down the window, some unexpected results may occur (post installation scripts will still run) - this is perfectly normal - please refer to the release notes in Section 3 for more info.

### 4.16.4 More Debugging Info

Some amount of trouble was taken to insure that decent debugging output was printed to the log – making best use of this output is your best bet in an error case.

## 4.17 Starting over – installing OSCAR again

If you feel that you want to start the cluster installation process over from scratch in order to recover from irresolvable errors, you can do so with the `start_over` script located in the `scripts` subdirectory.

It is important to note that `start_over` is *not* an uninstaller. That is, `start_over` does *not* guarantee to return the head node to the state that it was in before OSCAR was installed. It does a "best attempt" to do so, but the only guarantee that it provides is that the head node will be suitable for OSCAR re-installation. For example, the RedHat 7.x series ships with a LAM/MPI RPM. The OSCAR install process removes this



RedHat-default RPM and installs a custom OSCAR-ized LAM/MPI RPM. The `start_over` script only removes the OSCAR-ized LAM/MPI RPM – it does not re-install the RedHat-default LAM/MPI RPM.

Another important fact to note is that because of the environment manipulation that was performed via `switcher` from the previous OSCAR install, *it is necessary to re-install OSCAR from a shell that was not tainted by the previous OSCAR installation*. Specifically, the `start_over` script can remove most files and packages that were installed by OSCAR, but it cannot chase down and patch up any currently-running user environments that were tainted by the OSCAR environment manipulation packages.

Ensuring to have an untainted environment can be done in one of two ways:

1. After running `start_over`, completely logout and log back in again before re-installing. *Simply launching a new shell may not be sufficient* (e.g., if the parent environment was tainted by the previous OSCAR install). This will completely erase the previous OSCAR installation's effect on the environment in all user shells, and establish a set of new, untainted user environments.
2. Use a shell that was established *before* the previous OSCAR installation was established. Although perhaps not entirely intuitive, this may include the shell was that initially used to install the previous OSCAR installation.

Note that the logout/login method is *strongly* encouraged, as it may be difficult to otherwise absolutely guarantee that a given shell/window has an untainted environment.