

Table of Contents

<u>OSCAR Install Guide.....</u>	<u>1</u>
<u>Chapter 1: Introduction.....</u>	<u>3</u>
<u>1.1 Overview.....</u>	<u>3</u>
<u>1.2 Terminology.....</u>	<u>3</u>
<u>1.3: Overview of System Installation Suite (SIS).....</u>	<u>3</u>
<u>1.4 Supported Distributions.....</u>	<u>4</u>
<u>1.5 Minimum System Requirements.....</u>	<u>5</u>
<u>1.6 Document Organization.....</u>	<u>5</u>
<u>Chapter 2: Release Notes.....</u>	<u>7</u>
<u>2.1 Release Features.....</u>	<u>7</u>
<u>2.2 General Installation Notes.....</u>	<u>7</u>
<u>2.3 Networking Notes.....</u>	<u>7</u>
<u>2.4 SELinux Conflict.....</u>	<u>8</u>
<u>2.5 Distribution Specific Notes.....</u>	<u>8</u>
<u>2.5.1 RHEL 5.....</u>	<u>8</u>
<u>2.6 pfilter Notes.....</u>	<u>9</u>
<u>Chapter 3: Preparing to Install OSCAR.....</u>	<u>10</u>
<u>3.1 Installing Linux on the Server Node.....</u>	<u>10</u>
<u>3.2 Disk space and directory considerations.....</u>	<u>10</u>
<u>3.3 Configuration for the Usage of the On-line OSCAR Repositories.....</u>	<u>10</u>
<u>3.3.1 On CentOS/RHEL Based Systems.....</u>	<u>10</u>
<u>3.3.2 On Debian-5 and Ubuntu Based Systems.....</u>	<u>11</u>
<u>Chapter 4: Network Setup.....</u>	<u>12</u>
<u>4.1 Configure the ethernet adapter for the cluster.....</u>	<u>12</u>
<u>Chapter 5: Detailed Cluster Installation Procedure.....</u>	<u>14</u>
<u>5.1 Launching the OSCAR Installer.....</u>	<u>14</u>
<u>5.2 Manage OSCAR Repositories.....</u>	<u>15</u>
<u>5.3 Selecting Packages to Install.....</u>	<u>15</u>
<u>5.4 Configuring OSCAR Packages.....</u>	<u>16</u>
<u>5.4.1 Selecting a Default MPI Implementation.....</u>	<u>16</u>
<u>5.5 Install OSCAR Server Packages.....</u>	<u>17</u>
<u>5.6 Build OSCAR Client Image.....</u>	<u>17</u>
<u>5.7 Define OSCAR Clients.....</u>	<u>19</u>
<u>5.8 Setup Networking.....</u>	<u>20</u>
<u>5.8.1 Collect Client Node MAC Addresses.....</u>	<u>20</u>
<u>5.8.2 Select Installation Mode.....</u>	<u>21</u>
<u>5.8.3 Setup Boot Environment.....</u>	<u>21</u>
<u>5.8.4 Use Your Own Kernel (UYOK).....</u>	<u>21</u>
<u>5.9 Monitor Cluster Deployment.....</u>	<u>22</u>
<u>5.10 Client Installations.....</u>	<u>22</u>
<u>5.10.1 Network boot the client nodes.....</u>	<u>23</u>
<u>5.10.2 Check completion status of nodes.....</u>	<u>23</u>
<u>5.10.3 Reboot the client nodes.....</u>	<u>23</u>
<u>5.11 Complete the Cluster Setup.....</u>	<u>24</u>
<u>5.12 Test Cluster Setup.....</u>	<u>24</u>

Table of Contents

<u>Chapter 5: Detailed Cluster Installation Procedure</u>	
<u>5.13 Starting Over</u>	24
<u>5.14 Deleting Clients</u>	25
<u>Deleting node images</u>	25
<u>5.14 Reimaging the Cluster</u>	26
<u>Appendices</u>	27
<u>Appendix A: Network Booting Client Nodes</u>	28
<u>Appendix B: What Happens During Client Installation</u>	29
<u>Appendix C: Tips and Troubleshooting</u>	30

OSCAR Install Guide

1. Introduction

1. Overview
2. Terminology
3. System Installation Suite
4. Supported Distributions
5. System Requirements
6. Document Organization

2. Release Notes

1. Release Features
2. General Notes
3. Networking Notes
4. SELinux Note
5. Distribution Specific Notes
6. pFilter Notes

3. Preparing to Install OSCAR

1. Installing Linux
2. Disk Space
3. Configuration for the Usage of the On-line OSCAR Repositories

4. Network Setup

1. Configuring the NIC

5. Cluster Installation

1. Launching the Wizard
2. Downloading OPD Packages
3. Selecting Packages
4. Configuring Packages
5. Install Server Packages
6. Build Client Image
7. Define Clients
8. Setup Networking
9. Monitor Cluster Deployment
10. Client Installations
11. Complete Setup
12. Test Setup
13. Starting Over
14. Deleting Clients

6. Appendices

1. Network Booting
2. SIS Boot Description
3. Tips & Troubleshooting

Error: Macro TOC(None) failed

'NoneType' object has no attribute 'endswith'

[back to Table of Contents](#)

Chapter 1: Introduction

1.1 Overview

OSCAR version 6.1 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 computing (HPC) cluster. Everything needed to install, build, maintain, and use a Linux cluster is included in the suite.

OSCAR is the primary project of 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 installation progresses.

Please be sure that you have the latest version of this document. The PDF version of this document which is included in the distribution is a snapshot of the OSCAR Trac wiki <http://svn.oscar.openclustergroup.org/trac/oscar/wiki> which may have been updated since this version was released.

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.

It is possible to create clusters with only one type of processor and operating system (called a homogeneous cluster) or with more than one type of processor or operating system (called a heterogeneous cluster).

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 binary file, or it can be more complex, perhaps including a mixture of binary 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: Overview of System Installation Suite (SIS)

The System Installation Suite (SIS) is a cluster installation tool developed by the collaboration of the IBM Linux Technology Center and the [SystemImager](#) team. SIS was chosen to be the installation mechanism for OSCAR for multiple reasons:

- SIS is a high-quality, third party, open source product that works well in production environments
- SIS does not require the client nodes to already have Linux installed
- SIS maintains a database containing installation and configuration information about each node in the cluster

In order to understand some of the steps in the upcoming install, you will need knowledge of the main concepts used within SIS. The first concept is that of an image. In SIS, an image is defined for use by the cluster nodes. This image is a copy of the operating system files stored on the server. The client nodes install by replicating this image to their local disk partitions. Another important concept from SIS is the client definition. A SIS client is defined for each of your cluster nodes. These client definitions keep track of the pertinent information about each client. The server node is responsible for creating the cluster information database and for servicing client installation requests. The information that is stored for each client includes:

- IP information such as hostname, IP address, route.
- Image name.

Each of these pieces of information will be discussed further as part of the detailed install procedure.

For additional information on the concepts in SIS and how to use it, you should refer to the SIS(1) man page. In addition, you can visit the SIS web site at <http://wiki.systemimager.org/> for recent updates.

1.4 Supported Distributions

This version of OSCAR has been tested to work with a single Linux distribution. It is, however, known to have certain compatibility issues with some of the distributions. The suggested workarounds in such cases have been discussed in this document. 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.

Table 1: OSCAR Supported Distributions

Distribution and Release	Architecture	Status	Known Issues
Red Hat Enterprise Linux 5 / CentOS 5	x86	Fully supported	None
Red Hat Enterprise Linux 5 / CentOS 5	x86_64	Fully supported	None
Debian 5	x86_64	Fully supported	Not all OSCAR packages are supported
Ubuntu 10.04 (LTS)	x86_64	Fully supported	Not all OSCAR packages are supported

Clones of supported distributions, especially open source rebuilds of Red Hat 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.

1.5 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
- CD-ROM drive or PXE enabled BIOS on the client nodes

Monitors and keyboards are helpful, but are not required.

Given the wide variety of software and hardware combinations possible in a compute clusters, it is impossible for us to keep an accurate list of hardware which OSCAR supports. If the hardware works on a stock install of the host operating system, there are fair odds that it will not cause any problems with OSCAR.

1.6 Document Organization

Due to the complicated nature of putting together a high-performance cluster, it is strongly suggested that even experienced system 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 install OSCAR on your system. Next, the [Release Notes section \(Section 3\)](#) that applies to this OSCAR version contains some requirements and update issues that need to be resolved before the install. Section 4 provides an overview for the System Installation Suite software package used in OSCAR to perform the bulk of the cluster installation. Section 5 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.") Finally, Section 6 contains system administration notes about several of the individual packages that are installed by OSCAR. This section is a *must read* for all OSCAR system administrators.

Appendix A covers the topic of network booting client nodes, which is so important that it deserved its own section. Appendix B provides curious users an overview of what really happens during a client install. Appendix C discusses how to install an OSCAR cluster without a DHCP server. Appendix D covers a primer of some security aspects of a Linux cluster. Although not intended to be a comprehensive description of cluster security, it is a good overview for those who know relatively little about system administration and security. Finally, Appendix E is a screen-by-screen walk through of a typical OSCAR installation.

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](#) or the [mailing list archives](#). The OSCAR mailing lists only

accept posts from members of the lists so please join the list before posting a message. This helps to keep our lists spam free.

Error: Macro TOC(None) failed

'NoneType' object has no attribute 'endswith'

[back to Table of Contents](#)

Chapter 2: Release Notes

2.1 Release Features

The major new features for OSACR 6.x are:

- OSCAR is not installed in /opt anymore but directly on the system (for instance, binaries are in /usr/sbin).
- Full support of on-line repositories.
- New bootstrapping mechanism.
- Experimental support of Debian based systems.
- Better error handling.
- Source code reorganization, based on OPGs classification.

For more details, please refer to the [ChangeLog?](http://svn.oscar.openclustergroup.org/trac/oscar/browser/trunk/ChangeLog)
<http://svn.oscar.openclustergroup.org/trac/oscar/browser/trunk/ChangeLog>

2.2 General Installation Notes

- 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 Section 6 for additional release notes.
- 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 updates after the initial OSCAR installation has completed.
- The "Development Tools" packages are not default packages in all distributions and are required for installation.
- 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 is currently fairly dependent on what language is enabled on the head node. If you are running a non-english distribution, please execute the following command at your shell prompt before running the `install_cluster` script.

```
export LC_ALL=C
```

2.3 Networking 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.
- 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).

2.4 SELinux Conflict

According to the NSA <http://www.nsa.gov/selinux/info/faq.cfm>, the purpose of SE Linux is as follows:

- The Security-enhanced Linux kernel enforces mandatory access control policies that confine user programs and system servers to the minimum amount of privilege they require to do their jobs. When confined in this way, the ability of these user programs and system daemons to cause harm when compromised (via buffer overflows or misconfigurations, for example) is reduced or eliminated.
- Due to issues with displaying graphs under Ganglia, and installing RPMs in a 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 using the selinux OSCAR package (make sure you manually select the selinux OSCAR package to do so).

2.5 Distribution Specific Notes

- This section discuss issues that may be encountered when installing OSCAR on specific Linux distribution versions/architectures.

2.5.1 RHEL 5

RHEL 5 users needs to create a local repository for RHEL 5 RPMs, right after the installation of the oscar RPM. For that, copy all RPMs from the installation CDs or DVD in */ftpboot/distro/redhat-el-5-i386* (**always replace i386 by x86_64 if you are using a x86_64 machine**). Note that RPMs may be in different directories on the CDs/DVD and you really need all of them. Then execute the following command as root:

- *yum update && yum -y install packman createrepo*
- *cd /tftpboot/distro/redhat-el-5-i386 && packman --prepare-repo /tftpboot/distro/redhat-el-5-i386.*

2.6 pfilter Notes

- pfilter is the firewall which is bundled with OSCAR. Besides its normal function as a firewall, it also provides NAT access for the compute nodes in your cluster
- pfilter is currently unmaintained, and thus is not anymore included by default.

Error: Macro TOC(None) failed

'NoneType' object has no attribute 'endswith'

[back to the Table of Content](#)

Chapter 3: Preparing to Install OSCAR

3.1 Installing Linux on the Server Node

To install OSCAR, your server node must have a linux distribution installed. It should be noted that OSCAR is only supported on the distributions listed in Table 1 (page 8). 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.

When installing Linux, it is not necessary to perform a *custom* install since OSCAR will usually install all the software on which it depends. The main Linux installation requirement is that some X windowing environment such as GNOME or KDE must be installed. Typically, a ?Workstation? install yields a sufficient installation for OSCAR to install successfully.

OSCAR-6.0.x assumes the server node has access to internet in order to be able to access on-line repositories. So, please, check that your server node has an active internet connection.

3.2 Disk space and directory considerations

OSCAR has certain requirements for server disk space. Space will be needed to store the Linux binary packages and to store the images. 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. If you are installing a new server, it is suggested that you allow for 4GB in both the `/` and `/var` filesystems when partitioning the disk on your server.

If you are using an existing server, you will need to verify that you have enough space on the disk partitions. Again 4GB of free space is recommended under each of `/` and `/var`.

You can check the amount of free space on your drive's partitions by issuing the command `df -h` in a terminal. The result for each file system is located below the Available column heading.

The same procedure should be repeated for the `/var/lib/systemimager` subdirectory, which will later contain the images used for the compute nodes.

3.3 Configuration for the Usage of the On-line OSCAR Repositories

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.

3.3.1 On CentOS/RHEL Based Systems

1. As root, add the following file into your `/etc/yum.repos.d` directory:

```
# CentOS-OSCAR.repo
#
# If the mirrorlist= does not work for you, as a fall back you can try the
```

```
# remarked out baseurl= line instead.
#
#

[oscar]
name=CentOS-$releasever - OSCAR
baseurl=http://bison.csm.ornl.gov/repos/rhel-5-<arch>
gpgcheck=0
```

where **<arch>** is **i386** or **x86_64**, depending on the architecture of your server node.

1. Make sure that your system is up-to-date, executing as root **yum update**
2. To install the OSCAR RPM, execute as root **yum install oscar-base**
3. Check the content of the `/etc/oscar/oscar.conf` file; make sure it matches your configuration (for instance check the OSCAR interface, i.e., the network interface used to manage your cluster, is correctly set).
4. Execute as root **oscar-config --setup-distro <distro>-<version>-<arch>** (for instance **oscar-config --setup-distro centos-5-x86_64**). To get the full list of supported Linux distributions and get the exact syntax of the distribution identifier, please execute the **oscar-config --supported-distros** command.

3.3.2 On Debian-5 and Ubuntu Based Systems

1. As root, add the following line into your `/etc/apt/sources.list`:
 - ◆ On x86_64 systems: **deb http://bison.csm.ornl.gov/repos/debian-5-x86_64/ lenny /**
 - ◆ i386 systems are not at the moment supported by the OSCAR tea
2. Execute as root **aptitude update**
3. Make sure that your system is up-to-date
4. To install the OSCAR Debian package, execute as root **apt-get install oscar**
5. Check the content of the `/etc/oscar/oscar.conf` file; make sure it matches your configuration (for instance check the OSCAR interface, i.e., the network interface used to manage your cluster, is correctly set).
6. Execute as root **oscar-config --setup-distro <distro>-<version>-<arch>** (for instance **oscar-config --setup-distro debian-4-x86_64**). To get the full list of supported Linux distributions and get the exact syntax of the distribution identifier, please execute the **oscar-config --supported-distros** command.

Error: Macro TOC(None) failed

'NoneType' object has no attribute 'endswith'

[back to Table of Content](#)

Chapter 4: Network Setup

4.1 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.² 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 yourhos
```

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

There are three private IP address ranges: 10.0.0.0 to 10.255.255.255; 172.16.0.0 to 172.31.255.255; and 192.168.0.0 to 192.168.255.255. Additional information on

private intranets is available in RFC 1918. You should not use the IP addresses 10.0.0.0 or 172.16.0.0 or 192.168.0.0 for the server. If you use one of these addresses, the network installs of the client nodes will fail.

- Use an appropriate netmask

A class C netmask of 255.255.255.0 should be sufficient for most OSCAR clusters.

- Ensure that the interface is activated at boot time
- Set the interface control protocol to "none"

Now reboot the server node to ensure that all the changes are propagated to the appropriate configuration files. To confirm that all ethernet adapters are in the "up" state, once the machine has rebooted, open another terminal window and enter the following command:

```
# ifconfig -a
```

You should see UP as the first word on the third line of output for each adapter. If not, there is a problem that you need to resolve before continuing. Typically, the problem is that the wrong module is specified for the given device. Try using the network configuration utility again to resolve the problem.

Error: Macro TOC(None) failed

'NoneType' object has no attribute 'endswith'

[back to Table of Contents](#)

Chapter 5: Detailed Cluster Installation Procedure

5.1 Launching the OSCAR Installer

Change directory to the top-level OSCAR directory and start the OSCAR install wizard. If you placed the source in the location suggested in the earlier example, the commands to start the installer would be:

1. Execute as root **oscar-config --bootstrap**
2. Execute as root **system-sanity** and make sure you address all the reported issues
3. Execute as root **oscar_wizard install**

The **oscar-config --bootstrap** command will execute some setup / configuration steps, including (but not limited to):

- installs prerequisite packages on the server,
- installs all OSCAR server binary packages,
- updates */etc/hosts* with OSCAR aliases,
- updates */etc/exports*,
- start/restarts system services needed by OSCAR.

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

The wizard, as shown in Figure 1, is provided to guide you through the rest of the cluster installation. To use the wizard, you will complete a series of steps, with each step being initiated by the pressing of a button on the wizard. Do not go on to the next step until the instructions say to do so, as there are times when you may need to complete an action outside of the wizard before continuing on with the next step. For each step, there is also a <Help> button located directly to the right of the step button. When pressed, the <Help> button displays a message box describing the purpose of the step.

Figure 1: OSCAR Wizard.

In brief, the functions of the various buttons is as follows:

- **Step 0: Manage OSCAR Repositories** [disabled]
 - ◆ This step is currently disabled but it will be available in a future release.
- **Step 1: Select OSCAR Packages To Install**
 - ◆ This step allows the selection of non-core OSCAR Packages to be installed. Typically these are resource manager/scheduling systems, parallel programming libraries as well as other tools that aid in cluster administration. Certain packages may conflict with each other and only allow either one of them to be installed (eg. SGE vs TORQUE/Maui).
- **Step 2: Configure Selected OSCAR Packages** [optional]
 - ◆ Certain packages have configuration options that can be set prior to installation, these settings

can be set during this step.

- **Step 3: Install OSCAR Server Packages**

- ◆ This step would install all the selected packages (in Step 1) on the server (headnode) - this step is repeatable.

- **Step 4: Build OSCAR Client Image**

- ◆ This step allows the user to build an OS image using SystemInstaller. This image will then be pushed to the compute nodes as part of cluster installation.

- **Step 5: Define OSCAR Clients**

- ◆ After image(s) are created, clients to be part of your cluster needs to be defined. The user can select hostnames for your compute nodes, number of nodes, etc.

- **Step 6: Setup Networking**

- ◆ This step allows the user to tie MAC addresses to defined clients (in the previous step) such that when they boot up, they will automatically be imaged. Installation mode is also set in this step - currently available modes are: systemimager-rsync (default), systemimager-multicast, systemimager-bt. After this mode is set, the user should then configure DHCP Server and also select Setup Network Boot.

- **Delete OSCAR Clients** [hopefully unnecessary]

- ◆ This button allows the user to delete OSCAR clients from the cluster. Services will be stopped on the nodes to be deleted and restarted on all the remaining nodes. The cluster will be re-configured without the presence of the deleted node entries in ODA.

- **Monitor Cluster Deployment** [optional]

- ◆ This step brings up the SystemImager monitoring widget si_monitortk which provides very useful information regarding image progress. The user can also invoke the Virtual Console by double-clicking on a node and this will bring up a window with console messages during the installation.

- **Step 7: Complete Cluster Setup**

- ◆ Perform this step after all your cluster nodes have successfully been imaged and rebooted. This step will initiate post-cluster installation fix-ups and get it ready for production.

- **Step 8: Test Cluster Setup** [optional]

- ◆ OSCAR provides some tests for its packages and this step invokes all these testing harness to ensure that your cluster is setup properly and ready for production runs.

5.2 Manage OSCAR Repositories

Note: This step is disabled and it will be available in a future release.

5.3 Selecting Packages to Install

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. When you click on the button, a window similar to the one shown in Figure 3 appears. Each of the packages that the OSCAR installer has found are listed in the main frame. Core packages must be installed and cannot be unselected. Included packages can be unselected if desired.

Figure 3: OSCAR package selection.

Note that this window only shows OSCAR packages -- it does not show individual RPMs. Once you have a selected a set of OSCAR packages to install, click on the <Exit> button to save your selections and return to the main OSCAR window. Note that closing the window yields the same result and there is no way of defaulting to the original settings, so make sure your package list is complete before proceeding to the next step.

5.4 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. Figure 3 shows a sample with only the Environment Switcher package listed. Clicking on any of the packages' <Config> button will bring up a panel for configuring that package. Select whatever options are appropriate for that package, and then click on the <Save> button to save your selections, or the <Cancel> button to cancel all of your selections and leave the original settings. If you have saved your changes but want to go back to the default settings, simply click on the <Default Configuration> button and then the <Save> button to revert to the original settings.

Figure 4: OSCAR package configuration.

This step is optional. If you do not click on the <Configure Selected OSCAR Packages> button, defaults for all packages will be used.

5.4.1 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. The Environment Switcher package provides a convenient mechanism for switching between multiple MPI implementations. Section 6.13 contains more details about this package (page 40).

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.

You can change this default setting later -- see Section 6.13 for more details.

When you close the main Configuration window, the following benign warning may appear in the shell window (it is safe to ignore):

Tag "mpi" does not seem to exist yet. Skipping.

5.5 Install OSCAR Server Packages

This is the first required step of the OSCAR installation.

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.

A popup will appear indicating the success or failure of this step. Click on the <Close> button to dismiss it.

5.6 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. SELinux should also be deactivated on the head node.

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

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.

Building the image may take several minutes; *compared to previous OSCAR releases, there is progression bar (since we use online repository it is difficult to exactly know the current progression of the image creation). To follow the progression, one may follow the output in the console window during the build. It is normal to see some warning messages in the console. You can safely ignore these messages and wait for the final popup window announcing the success or failure of the overall image build.*

A sample dialog is shown in Figure 4.

Figure 4: Build the image.

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>
```

Here is a sample (for a SCSI disk):

```
/dev/sda1 24 ext3 /boot defaults  
/dev/sda5 128 swap  
/dev/sda6 * ext3 / defaults  
nfs_oscar:/home - nfs /home rw
```

An * in the size column causes that partition to grow to fill the entire disk. You can create your own partition files, but make sure that you do not exceed the physical capacity of your client hardware. Also be careful to not specify duplicate filesystems as this will cause problems during the installation. The sample listed above, and some others, are in the *oscarsamples* directory.

The disk partition file is auto-selected based on the type of disk(s) available on your headnode. However, SystemImager has the functionality where it can deploy images and be agnostic with the target client's disks, whether they are *hda*, *sda*, or other. In other words, even if you built your image with *ide.disk*, you can deploy the image to clients with SCSI/SATA hard disks and vice versa.

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".

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.

If the build fails, look through the console output for some indication as to what happened to cause the failure. Common causes include: prerequisite failure, ran out of disk space, and missing package files. Also see the Release Notes for this version of OSCAR in Chapter 3.

5.7 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 oscaromain will be put in the field (although you may change it). This field must have a value -- it cannot be blank. Note that especially for compute nodes on a private network, the domain name does not necessarily matter much. The domain name supplied in this field is used to form the fullyqualified name of each host in the OSCAR cluster. For example: oscarnode1.oscardomain, oscarnode2.oscardomain, etc. If your compute nodes are on a public network, you may want to use the "real" domain name that is part of their fully-qualified domain names.
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 oscarnode001. 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. See Footnote 3 on page 20 for more information on how to pick a starting IP address. Clients will be given IP addresses starting with this IP address, and incrementing by 1 for each successive client. Ensure that the range of [starting ip, (starting ip+num clients)] does not conflict with the IP addresses of any other nodes on your network.
8. The Subnet Mask specifies the IP netmask for all clients. See Footnote 4 on page 20 for more information on how to select a netmask for your cluster.
9. The Default Gateway specifies the default route for all clients.

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.

When finished entering information, press the <Addclients> button. When those clients have been created in the database, a popup will appear indicating the completion status. A sample dialog is shown in Figure 5.

Figure 5: Define the 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.

After all clients have been created, you may press the <Close> button in the build clients dialogue and continue with the next step.

5.8 Setup Networking

The MAC address of a client is a twelve hex-digit hardware address embedded in the client's ethernet adapter. For example, "00:0A:CC:01:02:03", as opposed to the familiar format of IP addresses. These MAC addresses uniquely identify client machines on a network before they are assigned IP addresses. DHCP uses the MAC address to assign IP addresses to the clients.

In order to collect the MAC addresses, press the <Setup Networking> button. The OSCAR network utility dialog box will be displayed. A sample dialog is shown in Figure 6.

Figure 6: Collect client MAC addresses.

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. For instructions on doing network booting, see Appendix A.

5.8.1 Collect Client Node MAC Addresses

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.
- To remove extraneous MAC addresses from the left hand window (e.g., if the collector finds MACs that are not part of your cluster), select the address and click on the <Remove> button. Or click on the <Remove All> button to remove all of them.

- At any time, you may click on the <Export MACs to file...> button to save the MAC address list to a file. If you need to re-run the OSCAR installation, you can later click on <Import MACs from file...> to import this file rather than re-collecting all the MACs.
- When you have collected all of the MAC addresses, click the <Stop Collecting MACs> button.

If you do not have <Dynamic DHCP update> selected, you need to click the <Configure DHCP Server> button to configure the DHCP server.

5.8.2 Select Installation Mode

SystemImager is the tool that OSCAR uses for deploying images to cluster nodes. It is part of a bigger suite of tools called the System Installation Suite (thus the package name in OSCAR is SIS).

SystemImager is responsible for deploying the OS image to your compute nodes over the network. It supports (as of version 3.7.3) three different transports: *systemimager-rsync* (default), *systemimager-multicast* (flamethrower), and *systemimager-bt* (bittorrent).

By default, systemimager uses a program called rsync to push files to your client nodes. To use one of these other installation modes click on the pull down list which by default displays "systemimager-rsync" and choose one of the other options. Then click the "Enable Install Mode" button to configure the server to use that method to install images on the client nodes. Especially with the multicast option, please make sure your router supports your desired transport method.

In case you cannot get your nodes to image using one of these optional install modes and want to switch back to using rsync, simply go back to the <Setup Networking> menu, select <systemimager-rsync> from the pull down list, and click on <Enable Install Mode>.

5.8.3 Setup Boot Environment

This menu also allows you to choose your remote boot method.

You **must** do one of two things in order for your client nodes to be able to get images from the master node:

- 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. See Appendix A for more details.

5.8.4 Use Your Own Kernel (UYOK)

SystemImager ships with its own kernel and ramdisk (initrd.img) used for starting up a minimal system for imaging your nodes. Although the SystemImager developers try their best to keep this kernel up-to-date with new hardware modules support, this is not always possible. Therefore, starting in version 3.6.x, a new functionality called UseYourOwnKernel? (UYOK) was introduced.

Let's say that you have installed a Linux distribution that supports your hardware on the server, UYOK allows you to take the running kernel (from the Linux distribution) and uses that as the SystemImager boot kernel. This, combined with a ramdisk generated on the fly from an architecture specific *initrd_template* package (eg. *systemimager-i386initrd_template*) allows the user to be able to boot and image a node as long as the target OS to be deployed supports the hardware.

If you had to install any custom kernel modules to get your hardware to work after installing the operating system, or if you have trouble getting your nodes to boot after trying the stock systemimager kernel, click "Enable UYOK" button in the "Setup Networking" step and then either select "Build AutoInstall? CD..." or "Setup Network Boot". Then SystemImager will configure itself to use the kernel running on the head node.

Manual Setup for UYOK

If for some reason you wish to set up the UYOK functionality by hand, instead of using the wizard, please do the following. This should not be necessary if the hardware of your client nodes and head node are sufficiently similar.

First use UYOK to generate a kernel/ramdisk pair, execute the following command on the headnode:

```
# si_prepareclient --server servername --no-rsyncd
```

If you specify the `--no-rsyncd` argument, it will not restart `rsyncd`.

The resulting kernel and ramdisk will be stored in `/etc/systemimager/boot`. Now copy these files to `/tftpboot` if you are PXE-booting. Make sure to edit your `/tftpboot/pxelinux.cfg/default` file with a sufficiently large `ramdisk_size` in the kernel append statement, eg.:

```
LABEL systemimager
KERNEL kernel
APPEND vga=extended initrd=initrd.img root=/dev/ram MONITOR_SERVER=192.168.0.2
```

Now SystemImager will use the UYOK boot package (which should recognize your hardware) to boot your nodes and successfully image them.

5.9 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.

5.10 Client Installations

During this phase, you will boot your client nodes and they will automatically be installed and configured. For a detailed explanation of what happens during client installation, see Appendix B.

The recommended method to perform client installations is via network boot. This is the most convenient way if the network interface cards on your client nodes support PXE-boot. If the network cards do not support PXE-boot, you can check the Etherboot Project <http://www.etherboot.org> to see if you can generate a

boot-ROM for your card. Once you have generated a boot floppy, you can use that to network boot your nodes.

If your client nodes do not have a floppy drive, or you cannot generate a working ROM, then it is still possible to boot your client nodes with an autoinstallation CD. Please refer to the documentation on the "Setup Networking" step for information on how to generate this CD.

If the network cards that come with your client nodes support PXE, then change the BIOS settings such that "Network" is always first in boot order. This ensures that your nodes always boot via the network and thus allows the Network Boot Manager to manage the boot order via software. Note that this tool may not work with etherboot generated ROMs.

Once your client nodes complete installation, its next boot action will be automatically changed to "LocalBoot?" meaning it will boot from hard disk. If for whatever reason this does not work, you can manually change this via the netbootmgr widget.

This widget is available via the OSCAR Wizard in Manage mode, or via the command line as "netbootmgr".

5.10.1 Network boot the client nodes

See Appendix A for instructions on network booting clients.

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

5.10.2 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.

If you chose "reboot" as the post-install action for your nodes, you will be notified when the nodes are rebooted via the <Monitor Cluster Deployment> widget. The node entry will turn green and the "Progress" field will say "REBOOTED".

The time required for installation depends on the capabilities of your server, your clients, your network, and the number of simultaneous client installations. Generally, it should complete within several minutes.

5.10.3 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 (the default), 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 the network cards that come with your client nodes do not support PXE, or has etherboot generated ROMs, you may need to reset the boot order in the BIOS to boot from the local disk.

5.11 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.

A popup window will indicate the success or failure of this step. Press the <Close> button to dismiss it.

5.12 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.

A sample dialog is shown in Figure 8. If any of the test fail, then there may be problem with your installation.

Figure 8: Testing your cluster setup

If all the tests pass, then your OSCAR cluster is ready to use. Congratulations!

5.13 Starting Over

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 capability via the **oscar-config** script (the *oscar-config --startover command*).

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.

The start_over script will try to delete all packages installed by OSCAR and drop the OSCAR database. It will also try to delete the packages which depend on the OSCAR binary packages.

Another important fact to note before starting a new OSCAR installation after using the startover capability 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 startover capability 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 starting 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.

5.14 Deleting Clients

If for some reason you need to delete a node from the cluster during installation, perhaps due to mistakes while adding the client nodes or assigning a MAC address to the wrong node, you simply need to click on the "Delete OSCAR Clients" button on the OSCAR wizard main menu. Then when the sub menu appears, select the problematic nodes from the list and click delete. Multiple nodes can be deleted by selecting multiple names from the list.

Deleting node images

It is also sometimes useful to be able to delete one or more of the node images which OSCAR uses to provision the client nodes or to change which image is sent to a node when it joins the cluster.

To delete an OSCAR image, you need to first unassign the image from the client(s) which are currently using that image and then run the command *mksiimage*.

There is currently no way to change which image is assigned to a node from within the OSCAR GUI, so first you will need to delete the client node(s) if you wish to change which image is used on a particular node. It is not necessary if the image simply changes, this procedure is only necessary to completely change a node to use a different image entirely. To do so, invoke the OSCAR Wizard and select "Delete OSCAR Clients...".

mksiimage is a command from [SystemInstaller](#) which is used to manage SIS images on the headnode (image server).

Assuming the name of your image is *oscarimage*, here are the steps you need to do to fully delete an OSCAR image.

First delete the client(s) associated with the image, then execute:

```
# mksiimage --delete --name oscarimage
```

If this command does not work for some reason, you can also use the command *si_rmimage* to delete the image, just pass it the name of the image as argument.

si_rmimage is a command from [SystemImager](#), the system OSCAR uses to deploy images to the compute nodes. [SystemImager](#) images are typically stored in `/var/lib/systemimager/images`.

Note: If you want to use the *si_rmimage* command, execute the following commands to delete all data:

```
# si_rmimage oscarimage -force
```

5.14 Reimaging the Cluster

Often in the process of setting up an OSCAR cluster for the first time, or on new hardware, it is necessary to re-image the cluster. Starting with OSCAR 5.x OSCAR uses the netboot manager to maintain a database of cluster states. Initially all nodes are marked for "Installation" but after it has been imaged successfully the state is marked so that if it is network booted it will boot from the local hard drive. This is a handy feature which allows system administrators to simply leave the cluster nodes set to network boot all the time, which saves a lot of time fiddling around with the BIOS. However this means that if you network boot your cluster again and expect it to reimage itself as it had previously, you will be disappointed.

In order to reimage your cluster it is a simple matter of loading up the net boot manager from your closest xterm with the "netbootmgr" cmd or loading up the management interface described in the OSCAR administration guide. For full details about how the netbootmgr interface works, please refer to the administration guide.

To reimage your cluster, click on the "All" button under the "Selection" menu. Then under the "Next Boot Action" pulldown select "Install" and click on the "Set" button. That should reset the cluster nodes listed at the left hand pane to all say "Install" as their next boot action.

[back to Table of Contents](#)

Appendices

1. [Network Booting](#)
2. [SIS Boot Description](#)
3. [Tips & Troubleshooting](#)

[back to Table of Contents](#)

Appendix A: Network Booting Client Nodes

There are two methods available for network booting your client nodes. The first is to use the Preboot eXecution Environment (PXE) network boot option in the client's BIOS, if available. If the option is not available, you will need to create a network boot CD disk using the SystemImager boot package or use an Etherboot disk. Each method is described below.

1. Network booting using PXE. To use this method, the BIOS and network adapter on each of the client nodes will need to support PXE version 2.0 or later. The PXE specification is available at <http://developer.intel.com/ial/wfm/tools/pxepdk20/>. Earlier versions may work, but experience has shown that versions earlier than 2.0 are unreliable. As BIOS designs vary, there is not a standard procedure for network booting client nodes using PXE. More often than not, the option is presented in one of two ways.
2. The first is that the option can be specified in the BIOS boot order list. If presented in the boot order list, you will need to set the client to have network boot as the first boot device. In addition, when you have completed the client installation, remember to reset the BIOS and remove network boot from the boot list so that the client will boot from its local hard drive and will not attempt to do the installation again. The net boot manager records when the client node has been imaged, and will cause the client to boot from its own hard drive. If you replace the hard drive or require it to be reimaged, use the oscar_wizard Net Boot Manager menu.
3. The second is that the user must watch the output of the client node while booting and press a specified key such as "N" at the appropriate time. In this case, you will need to do so for each client as it boots.
4. Network booting using a SystemImager boot CD. The SystemImager boot package is provided with OSCAR just in case your machines do not have a BIOS network boot option. You can create a boot CD through the OSCAR GUI installation wizard on the <Setup Networking> panel or by using the mkautoinstallCD command. Once you have created the SystemImager boot CD, set your client's BIOS to boot from the CD drive. Insert the CD and boot the machine to start the network boot. Check the output for errors to make sure your network boot CD is working properly. Remember to remove the CD when you reboot the clients after installation.
5. Using an Etherboot disk Etherboot is a software package for creating ROM images. This type of image is what drives the PXE network boot process described above. However, the Etherboot package (<http://www.etherboot.org/>) can also be used to create bootable floppy diskettes that mimic the PXE functionality of many network cards. This is useful for both older systems, and because booting off a diskette is sometimes easier than fiddling around with BIOS settings. A users manual with installation instructions can be found on the project's website (<http://www.etherboot.org/>). This tool is not supported by the OSCAR team directly, but is very handy.

[back to Table of Contents](#)

Appendix B: What Happens During Client Installation

Once the client is network booted, it either boots off the autoinstall CD that you created or uses PXE to network boot, and loads the install kernel. It then broadcasts a BOOTP/DHCP request to obtain the IP address associated with its MAC address. The DHCP server provides the IP information and the client looks for its auto-install script in `/var/lib/systemimager/scripts/`. The script is named `<nodename>.sh` and is a symbolic link to the script for the desired image. The auto-install script is the installation workhorse, and does the following:

1. partitions the disk as specified in the image in `<imagedir>/etc/systemimager/partitionschemes`.
2. mounts the newly created partitions on `/a`.
3. chroots to `/a` and uses `rsync` to bring over all the files in the image.
4. invokes `systemconfigurator` to customize the image to the client's particular hardware and configuration.
5. unmounts `/a`.

Once clone completes, the client will either reboot, halt, or beep as specified when defining the image.

[back to Table of Contents](#)

Appendix C: Tips and Troubleshooting

This is a rough collection of tips and tricks for when things don't work quite the way we might expect them to. If you carefully follow the documentation provided here, these suggestions should not be necessary. However, the suggestions here might be helpful if something does not work and you need a place to start.

Also, please check out the OSCAR website ([Support](#)) for more support tips and tricks.

Edit */etc/ssh/sshd.config* and ensure the `PermitRootLogin` is set to `Yes`. See [Build Client Images](#)

Then run `/etc/init.d/sshd reload`

Edit */etc/selinux/config* and ensure `SELINUX` is set to `disabled`.

Check the */etc/hosts.allow* and */etc/hosts.deny* files. They should allow all traffic from the entire private subnet.

Edit */usr/share/oscar/oscarsamples/ide.disk* and set appropriate size for swap (suggested twice the client's memory size), and alternate filesystem if desired (eg reiserfs).

Note: the systemimager will determine the correct type of drive (hda/sda) so the *ide.disk* file can be used for scsi hardware as well. An enhancement request has been filed to clear up this issue.

Copy your application (rpm) to the repository */tftpboot/distro/your-os-version/* and add the name to */usr/share/oscar/oscarsamples/your-os-version.rpmlist*. Make sure to select the correct file for the creation of the image.

Other methods of installing your applications after OSCAR has been installed will be covered in the [Administration Guide](#).

```
cpush xyz.rpm /usr/src/ & cexec rpm -i /usr/src/xyz.rpm
yume
scrpm --image all -w -- -Uhv xyz.rpm
```