How to Install an OSCAR Cluster
Quick Start Install Guide
Software Version 1.1-v3.0
Documentation Version 1.1-v3.0

http://oscar.sourceforge.net/

oscar-users@lists.sourceforge.net

The Open Cluster Group
http://www.openclutergroup.org/

July 8, 2005

Contents

1 Introduction **4**
- 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

2 Downloading an OSCAR Distribution Package **6**

3 Release Notes **6**
- 3.1 SSS-OSCAR Release Notes . . . 7
- 3.2 Notes for All Systems . . . 8
- 3.3 Red Hat 8/9.0 Notes . . . 10
- 3.4 Mandrake 9.0 Notes . . . 10
- 3.5 IA64 and Other Bleeding Edge Systems Notes . . . 11

4 Quick Cluster Installation Procedure **11**
- 4.1 Server Installation and Configuration . . . 11
  - 4.1.1 Install Linux on the server machine . . . 12
  - 4.1.2 Disk space and directory considerations . . . 12
  - 4.1.3 Download a copy of OSCAR and unpack on the server . . . 12
  - 4.1.4 Configure and Install OSCAR . . . 12
  - 4.1.5 Configure the ethernet adapter for the cluster . . . 13
  - 4.1.6 Copy distribution installation RPMs to `/tftpboot/rpm` . . . 13
- 4.2 Launching the OSCAR Installer . . . 14
- 4.3 Downloading Additional OSCAR Packages . . . 14
- 4.4 Selecting Packages to Install . . . 14
- 4.5 Configuring OSCAR Packages . . . 14
  - 4.5.1 Selecting a Default MPI Implementation . . . 15
- 4.6 Install OSCAR Server Packages . . . 15
- 4.7 Build OSCAR Client Image . . . 15
- 4.8 Define OSCAR Clients . . . 16
- 4.9 Setup Networking . . . 17
- 4.10 Client Installations . . . 18
  - 4.10.1 Network boot the client nodes . . . 18
  - 4.10.2 Check completion status of nodes . . . 18
  - 4.10.3 Reboot the client nodes . . . 18
- 4.11 Complete the Cluster Setup . . . 18
- 4.12 Test Cluster Setup . . . 18
- 4.13 Congratulations! . . . 19
- 4.14 Adding and Deleting client nodes . . . 19
  - 4.14.1 Adding OSCAR clients . . . 19
  - 4.14.2 Deleting clients . . . 19
- 4.15 Install/Uninstall OSCAR Packages . . . 19

4.15.1 Selecting the Right Package 20
4.15.2 Single Image Restriction 20
4.15.3 Failures to Install and Uninstall 20
4.15.4 More Debugging Info 21
4.16 Starting over – installing OSCAR again 21

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 v3.0. The documentation is generally the same except for the distribution support, which is limited to Red Hat 9. See Section 3.1 for other SSS-OSCAR specific release notes.

1 Introduction

The OSCAR (Open Source Cluster Application Resource) software package is intended to simplify the complex tasks required to install a cluster. While the usual intended use for OSCAR clusters is for high-performance computing (HPC), OSCAR clusters can be used for any cluster-enabled kinds of applications. Note that since OSCAR is aimed towards HPC, several HPC-related packages are installed by default, such as popular MPI implementations, PVM, PBS, etc.

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 1.1-v3.0.

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 no documentation is provided on how to do so.

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	Status
RedHat 9.0	Fully supported
RedHat 8.0	Fully supported
Mandrake 9.0	Fully supported

Table 1: OSCAR supported distributions

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[1]
- 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] 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.

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 1.1-v3.0 distribution package. Next, the "Release Notes" section (Section 3) that applies to OSCAR version 1.1-v3.0 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.

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 1.1-v3.0.

3.1 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 v3.0. Therefore the documentation is the same, except for the supported distributions. This release, SSS-OSCAR v1.1-v3.0, has been limited to Red Hat 9 (x86). [2]

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 v1.1-v3.0is limited to Red Hat 9 on x86. (*The following sections which detail standard OSCAR have details for the general OSCAR v3.0 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
  ```

- If standard manual pages are not available, use the following to extend the MANPATH (this is due to a problem with the modules/env-switcher shipped in oscar-3.0).

  BASH users: `export MANPATH=¨$MANPATH:¨`

  CSH users: `setenv MANPATH ¨$MANPATH:¨`

- 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,

[2] The version string indicates both the SSS version as well as the OSCAR version used for the release. For example, "sss-oscar-0.2a1-v3.0" is SSS alpha *v0.2a1* and OSCAR stable *v3.0*.

```
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):

```
[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.

3.2 Notes for All Systems

- 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 ("_").
- A domain name must be specified for the client nodes when defining them.
- 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`.
- 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.
- 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 RedHat updates *after* the initial OSCAR installation has completed. On the Mandrake systems the security updates must be added prior to the install.
- The following benign warning messages will appear multiple times during the OSCAR installation process:

```
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. You will be prompted to enter a password to be used for the MySQL database.

- 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.

- 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.

- During the <**`Build OSCAR Client Image`**> step, the "Successfully created image" notice will appear even though the status bar looks incomplete. This incomplete status bar can be ignored.

- 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 <**`Cancel`**> button in the <**`Install/Uninstall Package`**> step does not work properly; if any packages are selected to be installed or uninstalled, clicking the <**`Cancel`**> button still triggers the execution of the package installer/uninstaller. This will be fixed in a future release. The same behavior occurs if you close the window via the window manager's "close" functionality.

  Note that if you do not select any additional packages to install/uninstall, nothing will run (as expected).

3.3 Red Hat 8/9.0 Notes

There are a few issues that may crop up when using OSCAR on Red Hat 9.0 and/or Red Hat 8.0 clusters. The following items highlight these issues.

- Deselecting Pfilter causes the image creation to fail. This is due to a dependency with IPtables and when Pfilter is not selected the IPtables RPM is not listed in the node (image) rpmlist. The simple fix is to add "iptables" to the Red Hat 9.0 rpmlist if you are not installing Pfilter on the compute nodes.
- The RPM system has been updated with this Red Hat release. The OSCAR install process will likely display several warnings due to unsigned RPMS. These warnings can be ignored.
- In some OSCAR pre-release testing, RPM would hang during the building of a client image (Section 4.7). This is a documented bug in the version of RPM that ships with Redhat 8 and 9.0; it is not a problem with OSCAR. The procedure that was used to remedy this situation is outlined below (excerpts taken from http://www.rpm.org/hintskinks/repairdb-2003-06/):
  - If RPM hangs at any point (e.g., building the client image) – first ensure that it really has hung and just isn't taking a long, long time to complete. Typical indications that it has genuinely hung include: the disk is not running and load goes down to 0 (or nearly 0) and stays there.
  - Then do a `ps` and find the PID of the `rpm` process:

    ```
    # ps -eadf | grep rpm | grep -v grep
    ...output...
    # kill <PID_of_RPM>
    ```

  - This will probably not kill the process (it's likely to be in a state where it is ignoring signals), but it should be tried anyway – this would allow `rpm` to exit cleanly. If `rpm` does exit cleanly, jump down to the last step in this procedure.
  - If `rpm` does not exit within a short period of time, `kill -9 <PID_of_RPM>`. This guarantees that `rpm` will not exit cleanly, but in this case, it's ok. Now, do the following:
    1. Save a copy of the RPM database (just to be safe):

       ```
       # cd /var/lib
       # tar zcvf /tmp/rpmdb.tar.gz rpm
       ```

    2. Delete any existing RPM database locks:

       ```
       # cd /var/lib/rpm
       # rm -f __db*
       ```

    3. Rebuild the RPM database:

       ```
       # rpm -vv --rebuilddb
       ```

  - Now re-run the OSCAR step that hung. If RPM hangs again, repeat these steps to un-hang it. Testing has shown that it may be necessary to repeat these steps multiple times in order to get a successful RPM run.

3.4 Mandrake 9.0 Notes

The following may need to be run before attempting to install OSCAR on a Mandrake 9.0 cluster. If there are problems during the server preparation related to C3 or Python2 this may solve the issue.

- Install the Python 2 compatibility RPM. This RPM can be found in the OSCAR distribution package, under the `packages/c3/RPMS` directory:

```
# cd oscar-2.3/packages/c3/RPMS
# rpm -Uvh python2-compat-1.0-1.noarch.rpm
```

- The OSCAR wizard makes use of `xterm` so this package must be installed. You can check the RPM database for this package by typing `rpm -q xterm`. If `xterm` is not available, you must install the `xterm-165-3mdk.i586.rpm` RPM (you may need Mandrake CD's to run the `urpmi` command as mentioned above):

```
# urpmi xterm
```

- Lastly, `root`'s default shell configuration files hardcode the value for the `PATH` environment variable, regardless of what is added via `profile.d/` startup scripts. This effects various OSCAR installed components, which are installed into locations such as `/opt`. As such, the OSCAR installer appends the global path, e.g., `$PATH` or `${PATH}` depending on the shell, to the end of the path so OSCAR installed applications are in the search environment. This potentially makes changes to the following files:
  - `/root/.bashrc`
  - `/root/.cshrc`
  - `/root/.tcshrc`

3.5 IA64 and Other Bleeding Edge Systems Notes

IA64 support was removed for the 1.1-v3.0 release of OSCAR due to timing problems and the significant differences between current IA32 distributions and the freely available IA64 distributions. IA64 support is fully expected to return in the next release.

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) may not be sufficient, and may 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-1.1-v3.0
# ./configure
```

Once the configure script sucessfully 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-1.1-v3.0` 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.[3] 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 ("_").

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 the RPMs included with your Linux distribution into the `/tftpboot/rpm` directory. 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).

[3] Beware of certain models of 3COM network cards. See footnote 1 on page 5.

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 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.[4] 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.[5] 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] Ganglia is a notable package that is not included in the main OSCAR distribution. The process mentioned in this section will enable you to download Ganglia and install it with the rest of the OSCAR components.

[5] Note, if using OPD directly, the packages must be downloaded before the <**`Select OSCAR Packages To Install`**> step (see Section 4.4) .

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. 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 "_".

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 yeild 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.

   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!

You may also configure your remote boot method from this panel. The <**Build Autoinstall Floppy**> button will build a boot floppy for client nodes that do not support PXE booting. 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 multicast 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.

Incase 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 Client Installations

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

4.10.1 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.

4.10.2 Check completion status of nodes

After several minutes, the clients should complete the installation. You can watch the client consoles 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.

4.10.3 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, 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.

4.11 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.

4.12 Test Cluster Setup

A simplistic test suite is provided in OSCAR to ensure that the key cluster components (OpenSSH, PBS, 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.

4.13 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.

4.14 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>
```

4.14.1 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 4.11.

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

4.14.2 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**>.

4.15 Install/Uninstall OSCAR Packages

The package installation system is a new feature in OSCAR 3.0 and is designed to help simplify adding and removing OSCAR packages after the initial system installaion. The system is fairly straightforward for the user as packages are selected for the appropriate action from the wizard. This is the first release to support package install and uninstall from the Wizard therefore some general background is provided to assist the user if problems are encountered.

4.15.1 Selecting the Right Package

Because of the way OSCAR handles packages, two packages of the same name cannot currently c-exist 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.

4.15.2 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.

4.15.3 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. A `-f` flag is avoided.

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 serveral 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.

4.15.4 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.

4.16 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.