

OSCAR: A packaged Cluster software stack for High Performance Computing

The Open Cluster Group

www.OpenClusterGroup.org

February 22, 2001

Abstract

OSCAR is a fully integrated easy to install bundle of software designed to make it easy to build and use a cluster for high performance computing. Everything you need to build, maintain, and use a modest sized Linux cluster is included in OSCAR. In this note, we introduce OSCAR and provide the background information you need to use it.

Introduction

OSCAR is a package of RPM's, perl-scripts, libraries, tools, and whatever else is needed to build and use a modest-sized Linux cluster. With OSCAR, you don't need to roam the web to find what you need. Just download a single package, install it, and you're ready to start using your cluster.

The acronym, OSCAR, is a bit contrived, but it goes a long way towards explaining what the goal of the OSCAR project is:

Open Source Cluster Application Resources

First and foremost, OSCAR is an Open Source project. Every component within OSCAR is available under one of the well known Open-Source licenses (e.g. GPL). The goal of OSCAR is making clusters easy to build, easy to maintain and easy to use. In other words, OSCAR contains the resources you need to apply cluster computing to your High Performance Computing problems.

Now that you know what OSCAR is, lets be absolutely clear about what OSCAR is not. OSCAR is not a new standard. It is not an attempt to cram a particular approach to clustering down the collective throats of the high performance computing community.

Rather, OSCAR is a snapshot of current, *best-known-practices* in cluster computing. The creators of OSCAR – the Open Cluster Working Group – have studied what works at cluster-computing sites around the world, and gathered it into a single integrated package. Our hope is that with OSCAR it will be easy for people to replicate successful clustering techniques at their own sites.

OSCAR is not unique. The Beowulf and extreme-Linux projects have essentially implemented the same idea. What distinguishes OSCAR is its continuity. The Open Cluster Group will continue to update the package and make sure it represents a current snapshot of best-known-practices for building and using clusters.

Another unique feature of OSCAR is its origins from a collaboration of hardware vendors, software vendors, and national-labs. The national labs are a great source of experience and open source technology. Software vendors bring software expertise, but also channels for fully supported instantiations of OSCAR. Finally, the hardware vendors bring exposure to OSCAR and will be able to propagate it widely.

In these notes, we will describe what a typical OSCAR cluster looks like. We will then describe at a very high level each of the components in the current release of OSCAR.

OSCAR Cluster: hardware considerations

What does an OSCAR cluster look like? We expect clusters built with OSCAR to vary considerably from one site to another based on local policies or the needs of the cluster's users.

To help clarify the discussion that follows, we need to define a canonical OSCAR cluster. An OSCAR cluster consists of Servers, a gateway, nodes, and a network. Lets look at each one of these.

Servers: As the name implies, *servers* are computers that provide services to the cluster. This includes the NFS file server, the PBS server and any other server-functions required by OSCAR.

Gateway: The gateway has at least two network connections. One goes to the cluster's internal network. The second connection goes to an external local area network. By using a gateway, a cluster administrator can choose to relax security constraints inside the cluster. Since nodes can only leave the cluster through the gateway, you can put full security measures on the gateway's external LAN connection and protect the cluster behind it. This is an optional component of an OSCAR cluster, though we find it hard to believe a production cluster could meet the needs of an organization without a gateway.

Nodes: The nodes are the heart of a working cluster – the computers that do the actual computing. There must be at least one node, though more typically there will be somewhere between 4 and 100 nodes. The nodes must have local disk storage. Note that nothing in OSCAR precludes its use for huge clusters with many hundreds of nodes, but the package has not been tuned with arbitrarily large numbers of nodes in mind. The nodes can be different from each other (resulting in a heterogeneous cluster), but in most cases, we expect the nodes will be similar (i.e. the same CPU architecture and comparable memory sizes and speeds).

Network: Every cluster must have at least one network: i.e. some way of connecting its computers together. OSCAR currently requires that an Ethernet network connects the computers comprising the cluster. There may be an additional network to provide high performance communication, but this network will not be used to install cluster software.

While it isn't required in principle, we combine all servers within OSCAR onto a single node. It could be any node, but to keep things as simple as possible, we use one computer to be both our gateway and our server node.

OSCAR Software Components

OSCAR is a collection of integrated software components that are used to build, maintain and use a cluster. This big job can be broken down into a number of core functionalities:

- Installing Linux on each node.
- Building a database of the cluster and then semi-automatically installing OSCAR.
- Security
- Cluster management
- Setting up the libraries and tools needed to build programs to run on the cluster
- Workload management tools for multi-user clusters – batch queues, scheduling, and job monitoring.
- Packaging and documentation

In the following sub-sections, we describe the OSCAR components that implement each of these functionalities. We restrict ourselves to a high level overview of each package since more detailed documentation is included in the OSCAR distribution.

Linux installation: LUI

LUI [1] the *Linux Utility for cluster Install*, is a Linux installation tool sponsored by IBM and released under the GPL. It is used to install heterogeneous Linux clusters over a network.

We also use LUI to build a database describing the cluster itself. This database includes the node names, network information, cluster configuration data, and anything else required to install the other components of OSCAR.

Security: OpenSSH

A cluster usually sits in a "back-room" with users coming in over a LAN connection to the Gateway node. When everything is sitting behind a firewall and the level of trust between users is high, additional security measures may not be required. As soon as the cluster is connected to larger networks, however, all traffic to and from the cluster should be encrypted. If users need security from other users, then ssh can be used for inter-cluster instructions as well.

The most common way to allow secure connections in a Linux environment is with OpenSSH [7]. OpenSSH is a collection of packages that handle secure connections, server-side SSH services, secure-key generation and any other functions required to support secure connections between computers. We provide OpenSSH as an installation option within OSCAR. When running an isolated cluster, users may opt to use rsh, though we expect most sites running a production cluster will require OpenSSH.

Cluster management: C3

Each computer in a cluster runs its own copy of the operating system. In many ways, this is a "good thing": since it allows you to build a high performance supercomputer using "off the shelf" operating systems designed for the mass-market. The problem is that for certain operations, you want to view the cluster as a single computer. Files need to be moved around the computer, processes started on groups of nodes, and other system-wide operations we take for granted on a single computer.

We call these operations "cluster management". The cluster management package used in OSCAR is C3 [2] from Oak Ridge National Laboratory.

Programming environments: MPI and PVM

Most users of clusters write the software that runs on the cluster. There are many different ways to write software for clusters. The most common approach is to use a message passing library. We have included both MPI [4] (Message Passing Interface) and PVM [5] (Parallel Virtual Machine). At this time, compilers or math libraries installed by OSCAR come from the Linux distribution.

Workload Management: PBS

When multiple people share a cluster, some type of workload management is needed. Actually, even one person running a large mixture of jobs may need help managing the work. A workload management system ensures that every person (or job) gets their fair share of the cluster, and all resources get used efficiently.

There are three distinct components to workload management: resource management, job management, and job scheduling. For our workload management software, we use PBS [3], the Portable Batch System from Veridian. Once you queue jobs to PBS, PBS monitors the state of the cluster, and handles starting (and stopping) jobs, and delivery of output. Effectively scheduling jobs on the cluster requires a job scheduler.

For now, we are using the default scheduler that is included with PBS. In the future, however, we will probably include the well-known Maui scheduler [6].

OSCAR 1.0: Release Notes

In release 1.0 of OSCAR, we include the following specific packages:

- LUI - 1.9
- PBS - 2.2p11

- OpenSSL - 0.9.5a
- OpenSSH - 2.1.1p1
- MPICH - 1.2.1 (from MCS Software based upon MPICH 1.2.0)
- PVM - 3.4.3
- C3 - 2.6

In each case, we include the full package including source code and associated documentation. Information about how to use each of the above packages can be found in the distribution directory for the package in question.

In addition to the above packages, OSCAR 1.0 includes an installation wizard based closely on the work by the LUI team on a GUI for LUI. After loading the OSCAR distribution onto the server node, a shell script called *install_cluster* is run. This will bring up the installation wizard, which will guide you through each step of building an OSCAR cluster.

OSCAR has been validated for RedHat 6.2. It should be straightforward to use OSCAR with other Linux distributions, but we have not tested this yet.

Conclusion and Future Plans

We have big plans for OSCAR. There are other components we are looking into adding to OSCAR:

- We are exploring various GUIs to provide an easy to use single point configuration.
- There are still too many steps in building an OSCAR cluster that are manual. We are going to continuously strive to automate OSCAR.
- We plan to add the Maui Scheduler.
- We need to expand the OSCAR procedures to make it easier to deviate from our canonical cluster. For example, a user should be able to easily omit installation of packages they don't plan to use (for example, a single-user cluster may not need PBS).

In addition to the above specific plans, we are going to explore ways to extend OSCAR to a broader range of clusters. For example, it might be nice to use OSCAR for building high availability clusters. Another possibility is to extend OSCAR to handle diskless nodes.

In addition to changes in OSCAR itself, we have big plans for the project overall. We want to see OSCAR and the Open Cluster Group grow into a major force in cluster computing. We want to see OSCAR become a starting point that companies will use to build supported cluster software stacks. We want academics interested in tools-research to use OSCAR as the core software they build their tools on top of.

Computer scientists spend way too much time "re-inventing the wheel". We hope that OSCAR can help put an end to this re-inventing - at least in terms of the basic components of an open source cluster.

Acknowledgements

The Open Cluster Group includes representatives from Dell Computers, IBM, Intel Corporation, the National Center for Supercomputing Applications, Oak Ridge National Laboratory, MSC-software, SGI, and Veridian Corporation. Key contributors to the project are:

- Gabriel Broner and John Hesterberg of SGI
- Rich Ferri and Michael Chase-Salerno of IBM
- David Lombard of MSC-software
- Tim Mattson of Intel
- Jenwei Hsieh-Tau Leng and Yung-Chin Fang of Dell
- Bill Nitzberg and Bhroam Mann of Veridian
- Rob Pennington, Jeremy Enos, and Neil Gorsuch of NCSA
- Stephen Scott, Brian Luethke, and Mike Brimm of ORNL

Of these people Rich Ferri, Michael Chase-Salerno, David Lombard, Tim Mattson, Bhroam Mann, Jeremy Enos, Neil Gorsuch, Stephen Scott and Mike Brimm did the actual hard work of finding and packaging the software inside OSCAR.

PBS includes software developed by NASA Ames Research Center, Lawrence Livermore National Laboratory, and Veridian.

References

- [1] LUI, <http://oss.software.ibm.com/lui>
- [2] C3, <http://www.epm.ornl.gov/torc/C3/>
- [3] www.OpenPBS.org. The Portable Batch System Software (PBS v2.2p13), Veridian, PBS Products Dept., Mountain View, CA, March 2000.
- [4] MPI, <http://www-unix.mcs.anl.gov/mpi/>
- [5] PVM, <http://www.epm.ornl.gov/pvm/>
- [6] Maui scheduler, <http://mauischeduler.sourceforge.net/>.
- [7] OpenSSH, <http://www.openssh.com/>

Glossary

The definitions in this glossary are specialized to their use in OSCAR

Boot kernel: The kernel that is sent over the network to initially boot the node, mount the remote root file system, and prepare the hard-drive for installation.

Linux Kernel: The kernel that is permanently installed on the hard-drive of the node, and is used to boot the node after installation.

MAC Address: A 6 byte hex address that is uniquely assigned to each ethernet adapter. The MAC address is used to identify individual nodes during the boot process.

RAM disk: The RAM disk is used to configure adapters that are not directly supported in the Linux kernel. Refer to the Linux `mkinitrd` command for further details.

RPM: RedHat Package Manager. RPM is the most widely used Linux method of packaging software so that it checks for pre-reqs and co-reqs during installation of the package.