Rocks as Cluster Management

Abstract
Modern high performance computing (HPC) is mainly fulfilled on the cluster environment. Several cluster management software came out to make it easier to build clusters. Rocks is one of them. In this paper I go through the main features of Rocks and analyze its pros and cons by comparing with other clustering toolkits such as OSCAR (Open Source Cluster Application Resources) and Perceus.

1. Introduction
Setting up a cluster requires a lot of system administration works including the installation and configuration of a lot of software on many computers connected with specially designed high bandwidth network. Rocks is a software package that is designed to simplify cluster installation. The primary goal for Rocks is to make cluster installation as easy as possible. Rocks include a collection of open source cluster software for a dedicated, high-performance cluster.

In my survey paper, I examine the installation of Rocks in Section 2, introduce its main features in Section 3, and analyze its pros and cons by comparing other clustering toolkits in Section 4. In Section 5 I discuss my analysis of Rocks features and conclude my survey paper.

Keywords
Cluster, Rocks, Clustering toolkit, OSCAR, Perceus
2. Installation

I went through the Rocks installation a few years ago and it was pretty straightforward. I do not install Rocks this time because the installation steps have not changed since then. Instead, I examine the installation instructions posted at the Rocks’ website. Rocks installation consists of two steps: 1) Install and configure frontend 2) Install compute nodes.

Rocks requires the following minimum hardware to setup a cluster.

- **Frontend Node**
  - **Disk Capacity:** 30 GB
  - **Memory Capacity:** 1 GB
  - **Ethernet:** 2 physical ports (e.g., "eth0" and "eth1")
  - **BIOS Boot Order:** CD, Hard Disk

- **Computed Node**
  - **Disk Capacity:** 30 GB
  - **Memory Capacity:** 1 GB
  - **Ethernet:** 1 physical port (e.g., "eth0")
  - **BIOS Boot Order:** CD, PXE (Network Boot), Hard Disk

### Install and Configure frontend

Users need to prepare all the required and preferred roll CDs for the installation of frontend. Once the installation starts, the previous existing system is wiped out and a new operating system (RedHat Enterprise Linux) is installed. The new operating system is installed and all the selected packages in Rocks are installed to setup the preferred roll’s features with their default configuration. The frontend is ready to serve as a head node in a cluster.

#### Install Compute Nodes

The frontend is ready to find all the connected compute nodes in its subnet by running the “insert-ethers” command on the frontend. When a compute node is booted up with PXEboot (Preboot Execution Environment), its MAC address shows up on the frontend and a new IP is assigned to the compute node. Once the compute node is fully connected to the frontend, it requests a kickstart file from the frontend and it starts to install all the software in kickstart file. The rest of compute nodes will be installed in the same manner.

3. Main Features

Default installation of Rocks is very simple and goes smoothly. Rocks assume that users deal with all the software problems that may happen on a node by reinstalling the system on that node rather than trying to figure out the issues and to fix them. Rocks supports various HPC solutions such as Area 51, Bio, Ganglia, PVFS2,
Sun Grid Engine (SGE), Grid, and so on by providing its own independent roll with CD. The Area 51 roll includes tripwire and chkrootkit to have security enhancement in Rocks. The Grid roll has a full complement of grid software, such as Globus toolkit, Condor, and MPICH-G2. Especially Rocks is well designed to support the grid computation.

Setting up a cluster is associated with the following features that Rocks tries to satisfy all.

- Various Linux distributions and architectures
- Customizing the head node
  - User account management (synchronize access through the whole system)
  - Monitoring systems
  - Firewall to disallow the direct access to the client nodes
- Customizing the client nodes
  - Updating packages
  - Changing disk partitions

Rocks is tightly bound to RedHat Enterprise Linux (RHEL) distribution and it supports most features that RHEL provides. The RPM package management tool and Kickstart provisioning solution are the features that Rocks inherited from RHEL automatically.

One of main features in Rocks is the web-based management tools which can provide a flexible access to the management interface. Rocks tries to provide recently many other features: new rolls to support more HPC solutions, Virtual LANs (VLANs), and Logical VLAN Bridges. The command lines to manage a cluster are very useful. For example, the “sync” command with some proper options can sync the configuration, DNS, firewall, networks, and users automatically through the whole nodes. As Figure 1 shows the main architecture of Rocks, all the compute nodes are sitting behind the Ethernet Switch so

![Figure 1. Rocks fundamental architecture.](image-url)
that the internal communication among the nodes is secure and can be very fast depending on the network infrastructure.

4. Pros and Cons

I introduce two other clustering toolkits to compare Rocks. OSCAR (Open Source Cluster Application Resources) and Perceus have been developed later than Rocks that started its development in 2000 but they have played an enormous role to improve the HPC world like Rocks. They are well equipped with their own features to compete Rocks.

The OSCAR group started the OSCAR development in 2000 and they released their first official software in 2002. OSCAR provides very flexible cluster deployment solution but it is not designed to fully support for Grid and PVFS.

Perceus is designed to provide the clustering provision solution for the large quantities of machine in a scalable fashion. Perceus uses Virtual Node File System (VNFS) to enable the client nodes to run on either the stateless system or stateful system. The stateless system is the system that the operating system is installed on RAM or via the network file system. The normal system with the operating system installed on the hard disk is the stateful system.

As Table 1. shows, Rocks, OSCAR, and Perceus have their own provisioning solutions and I can not tell which one is the best for the cluster management but I try to analyze their features depending on their solution. Rocks’ flexibility on the re-configuration of the system is low in comparison to the others because Rocks does not allow the users to tweak the systems to fix the problems but to re-install the problematic nodes to remove the issues. It is simple but not many options are available for the users. Rocks is designed to the specific roll of features and it is completely stable for the target features. The usability of Rocks is high enough because it does not require much learning curve to setup a cluster like OSCAR. All three toolkits are not fully developed to support the multiple clusters. It is possible for users to setup the multiple clusters by re-configuring the existing clusters that are built by the three toolkits. The easiness of maintenance of Rocks cluster is not perfect because it does not provide a functional solution to update the running client nodes without shutting down them or re-installing the new system on them. The existing systems would be removed when Rocks is installed but OSCAR can keep the existing head node only. On contrast to them, Perceus can keep all the nodes as they are because its client nodes can be deployed with VNFS. The ability to setup the
heterogeneous system on Rocks is not good because Rocks works with only RHEL. Even with some drawbacks Rocks has the highest market share in the HPC world. I think it is because of its simplicity and stability in the cluster installation and management.

<table>
<thead>
<tr>
<th></th>
<th>Rocks</th>
<th>OSCAR</th>
<th>Perceus</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Provisioning solution</strong></td>
<td>Kickstart</td>
<td>SIS (System Installation Suite)</td>
<td>VNFS</td>
</tr>
<tr>
<td><strong>Flexibility</strong></td>
<td>Low</td>
<td>Medium</td>
<td>High</td>
</tr>
<tr>
<td><strong>Stability</strong></td>
<td>High</td>
<td>Medium</td>
<td>Medium</td>
</tr>
<tr>
<td><strong>Usability</strong></td>
<td>High</td>
<td>High</td>
<td>Medium</td>
</tr>
<tr>
<td><strong>Multiple clusters</strong></td>
<td>Low</td>
<td>Low</td>
<td>Low</td>
</tr>
<tr>
<td><strong>Maintainability</strong></td>
<td>Medium</td>
<td>High</td>
<td>High</td>
</tr>
<tr>
<td><strong>Re-usability of the existing system</strong></td>
<td>Low</td>
<td>Medium</td>
<td>High</td>
</tr>
<tr>
<td><strong>Heterogeneous system</strong></td>
<td>Low</td>
<td>High</td>
<td>High</td>
</tr>
<tr>
<td><strong>Market share</strong></td>
<td>High</td>
<td>Medium</td>
<td>Low</td>
</tr>
</tbody>
</table>

Table 1. Comparison among Rocks, OSCAR, and Perceus in terms of their features
5. Conclusions
In this paper, I identified analysis of features in Rocks by comparing with OSCAR and Perceus. Rocks is not the perfect clustering toolkit but it has a high market share in the HPC world because it is well designed for the specific clustering solutions. The production cluster does not really care about the flexibility and heterogeneous system support but the stability and availability of the system. Rocks hits the edge and I believe that two other clustering toolkits keep up the good works.

References
Joseph D. Sloan (2004): High Performance LINUX CLUSTERS with OSCAR, Rocks, OpenMosix, and MPI
Gregory F. Pfister: In Search Of CLUSTERS (2nd Edition)
Rocks user manual in http://www.rocksclusters.org/
OSCAR user manual in http://svn.oscar.openclusterorg.org/
Perceus user manual in http://www.perceus.org/
DongInn Kim, Jeffrey Squyres, Andrew Lumsdaine (2005): Revamping the OSCAR Database: A Flexible Approach to Cluster Configuration Data Management
DongInn Kim, Jeffrey Squyres, Andrew Lumsdaine (2006): The Introduction of the OSCAR Database API (ODA)