Streamlining Beowulf Cluster Deployment with

NPACI Rocks

For high-performance parallel-processing applications, Beowulf clusters that comprise industry-standard, two-processor and four-processor servers can be a cost-effective alternative to symmetric multiprocessing computer systems and supercomputers. This article discusses NPACI Rocks, an open source cluster computing software stack that can be used to improve the deployment, management, and maintenance of Beowulf clusters.

BY RINKU GUPTA, YUNG-CHIN FANG, AND MUNIRA HUSSAIN

Because they are built from cost-effective, industry-standard components such as Dell® PowerEdge™ servers, Beowulf clusters can provide a price/performance ratio that makes them a compelling alternative to more expensive symmetric multiprocessing (SMP) systems for many high-performance parallel computing systems. However, administration of clusters can become problematic for IT organizations as clusters grow.

Along with the popularity and customer acceptance of Beowulf clusters has come the need for a robust and comprehensive cluster computing software stack to simplify cluster deployment, maintenance, and management. This article introduces NPACI Rocks, an open source, Linux®-based software stack for building and maintaining Beowulf high-performance computing (HPC) clusters.

The NPACI Rocks toolkit was designed in November 2000 by the National Partnership for Advanced Computational Infrastructure (NPACI). The NPACI facilitates collaboration between universities and research institutions to build cutting-edge computational environments for future scientific research. The organization is led by the University of California, San Diego, and the San Diego Supercomputer Center.

NPACI Rocks is designed to make clusters easy to deploy, manage, maintain, and scale. The Rocks package is built on standard and mostly open source components and is available as a free download on the NPACI Rocks Web site.¹

Rocks software components and capabilities

NPACI Rocks provides a collection of integrated software components that can be used to build, maintain, and operate a cluster. Its core functions include the following:

- Installing the Linux operating system (OS)
- Configuring the compute nodes for seamless integration with the cluster
- Constructing a database of cluster-wide information
- Delivering middleware libraries and tools that build programs to run on the cluster
- Monitoring the cluster
- Managing the cluster
- Providing for system integration, packaging, and documentation

¹For more information about Rocks or to download Rocks, visit www.rocksclusters.org/Rocks.

Installing the Linux OS. Rocks is based on the Red Hat® Linux OS. The availability of kickstart tools and the RPM®
(Red Hat Package Manager) under Red Hat Linux served as a great impetus for the selection of Red Hat Linux as the base OS. The OS is installed on the compute nodes using the Preboot Execution Environment (PXE), in which the client nodes perform a network boot to obtain the OS. Rocks recompiles the available Red Hat RPMs and uses them in the Rocks package.

**Configuring the compute nodes for seamless integration with the cluster.** Like other open source and commercial cluster packages, Rocks uses a master node—called a front-end node in the Rocks terminology—for centralized deployment and management of a cluster. The Rocks front-end node helps administrators to specify the cluster-wide private cluster network configuration and networking parameters, including IP addresses assigned to the compute nodes. These parameters are specified during the front-end node installation process. The front-end node then provides the IP addresses to the compute nodes when the compute nodes contact the front-end nodes via network PXE boot.

**Constructing a database of cluster-wide information.** Many services at and above the OS level—for example, job schedulers and the Dynamic Host Configuration Protocol (DHCP) server—require a global knowledge of the cluster to generate service-specific configuration files. The front-end node of a Rocks cluster maintains a dynamic MySQL database back end, which stores the definitions of all the global cluster configurations. This database forms the core of the NPACI Rocks package, and can be used to generate database reports to create service-specific configuration files such as /etc/hosts and /etc/dhcppd.conf.

**Delivering middleware libraries and tools that build programs to run on the cluster.** Rocks delivers code using two types of CD. The Rocks Base CD contains the core Rocks package. Rocks Roll CDs are add-on software packages that augment the cluster with specific capabilities. The Rocks Roll software packages, designed to seamlessly integrate with the base Rocks installation, provide a mechanism to enable administrators and vendors to add software that provides extra functionality to Rocks clusters. Administrators can create their own Rolls, independent from the Rocks Base CD, to contain their domain-specific software and applications.

The following components come packaged in a separate HPC Roll CD: middleware (for example, MPICH,\(^2\) a portable implementation of Message Passing Interface [MPI], and Parallel Virtual Machine [PVM]\(^3\)); cluster performance monitoring software such as Ganglia;\(^4\) and common performance benchmark applications such as Linpack. Other Rolls provided by NPACI include Intel\(^\circledR\) compilers; the Maui\(^\circledR\) scheduler, and Portable Batch System (PBS)/resource manager functionality.

**Monitoring the cluster.** Cluster monitoring is an important task for helping system administrators to proactively understand and troubleshoot issues occurring in a cluster. Rocks is packaged with open source software tools such as Ganglia, which helps provide in-band global monitoring of the entire cluster. Ganglia is an open source tool developed by the University of California at Berkeley and the San Diego Supercomputer Center under National Science Foundation (NSF) NPACI funding.

**Managing the cluster.** Efficient tools for cluster management are an important part of any cluster computing package. NPACI Rocks provides easy-to-use command-line interface (CLI) commands for adding, replacing, and deleting nodes from the cluster. Rocks treats compute nodes as soft-state machines—that is, machines that have no application information or data stored on their hard drives—and uses fresh OS reinstallation as a way to ensure uniformity across all the nodes in the cluster. This OS-reinstallation approach works well for large clusters because the OS is installed rapidly and automatically. As a result, no time is wasted in performing exhaustive checks to debug problems that might exist in a system on which the OS has already been installed. For cluster-wide operations, NPACI Rocks comes with a parallel command (cluster-fork), which is capable of executing query-based operations.

**Rocks cluster installation**

NPACI Rocks cluster installation consists of two parts. Figure 1 shows a typical Rocks cluster layout.

**Front-end node installation**

Front-end node installation requires the administrator to install the Rocks Base CD along with any of the optional Rocks Roll CDs. As of Rocks 3.3.0, the Rocks Base CD along with the HPC Roll CD and the Kernel Roll CD are required to build a functional Rocks-based HPC cluster.

A front-end node typically has two network interface cards (NICs). One of the NICs connects to the external, public network. The other NIC is used to connect to the private network of the cluster. Rocks interactive front-end node installation allows the administrator to configure the external-network NIC. It also allows the administrator to specify the network configuration details (such

---

1 For more information about MPICH, visit www-unix.mcs.anl.gov/mpi/mpich.
2 For more information about PVM, visit www.csm.ornl.gov/pvm.
3 For more information about Ganglia, visit ganglia.sourceforge.net.
4 For more information about Intel compilers, visit www.intel.com.
5 For more information about Maui, visit www.clusterresources.com/products/maui.
6 For more information about PBS, visit www.openpbs.org.
as IP address and subnet mask) of the private network NIC. These
details are used to assign IP addresses to the client nodes of the
cluster in the latter part of the cluster installation process.

Rocks is self-contained, including the necessary tools, scripts,
and services for subsequent installation phases. Rocks front-end node
installation takes place on bare-metal hardware. The Rocks CDs install
the OS on the front-end node during the installation procedure.

Compute node installation
The rest of the Rocks cluster is installed from the front-end node,
using a test-based utility called insert-ethers (see Figure 2). To build
the compute nodes, administrators can launch the insert-ethers util-
ity from the front-end node. The insert-ethers utility is completely
database driven. It allows the administrator to configure each node
as a compute node, to be used for computational purposes; a Parallel
Virtual File System (PVFS) node, to be used only for I/O purposes;
or a combination compute and PVFS node, to be used for both com-
putational and I/O purposes.

The insert-ethers utility is also used to populate the nodes
table in the MySQL database. To this end, the utility continuously parses
the Linux OS log files (such as the /var/log/messages file) to extract
DHCPDISCOVER messages and Media Access Control (MAC) addresses. When the compute node is PXE-booted, the DHCP
request is sent to the front-end node. When the front-end node finds
the MAC address in the parsed files, and if the MAC address is not
present in the database, then the front-end node accepts the new
node and installs it using the Rocks kickstart mechanism. However
if the MAC address is already present in the database, the front-
end node re-installs the compute node by assigning the same IP
address that it previously had. Hence, the insert-ethers utility merely
reimages the compute node by matching previous information pre-
sent in the database. Front-end node scripts in Rocks generate the
kickstart file on the fly depending on the information—such as
architecture and CPU count—provided by the compute node through
the modified Rocks-supplied Red Hat Anaconda installer.

Rocks is designed to make compute node installation easy.
Rocks can also detect NICs (such as Myricom Myrinet® cards)
on the compute nodes and compile the correct drivers for these
NICs during installation. Rocks is based on XML and the Python®
programming language. This design, combined with the MySQL
database back end, helps provide great flexibility for administra-
tors to modify compute node settings such as disk partitioning;
enable selected services such as Remote Shell (RSH); and install
third-party packages. An administrator can customize the target
node environment by modifying select XML files in Rocks and
simply re-creating a new Rocks distribution, which can then be
used for installation.

Rocks cluster management and monitoring
Rocks uses the basic mechanism of OS reinstallation for updating the
nodes in the cluster. If a hardware failure occurs, or if a hard drive
or NIC must be replaced, the compute node should be reinstalled
from the front-end node; the front-end node’s MySQL database is
automatically updated if required. If additional software or update
files need to be installed on the compute nodes, then the administra-
tor should add these to the front-end node and create an updated
Rocks distribution. All nodes should then be reinstalled using the
updated distribution to help maintain consistency across the cluster.
Reinstallation is fast and helps reduce or eliminate administrator
time spent troubleshooting and debugging failed nodes.

The Rocks CLI is also used to replace and remove nodes from
a cluster. An administrator can create a naming scheme that helps
identify nodes by their cabinet and rack locations.

---
9For more information about PVFS, visit www.parallel.com/pvfs.
10For more information about Myrinet, visit www.myrinet.com.
11For more information about Python, visit www.python.org.

Figure 1: A typical Rocks cluster layout

Figure 2: Steps to install a Rocks cluster
For monitoring, Rocks provides a set of Web pages, which are accessible from the front-end node of the cluster. The administrator can view the MySQL database (using a MySQL Web-based administration tool called phpMyAdmin) that the front-end node of the cluster creates and uses. Ganglia is the default software package bundled with Rocks for monitoring purposes. The Web pages provide a graphical user interface (GUI) for use on the Ganglia monitors running on each cluster node. These Ganglia monitors provide a host of information including CPU usage and load, memory usage, and disk usage. Rocks also provides a utility called Cluster Top, which can be viewed from a browser on the front-end node. The Cluster Top utility is a version of the standard top command for the cluster. The Cluster Top Web page presents process information from each node in the cluster.

In addition, administrators can use the cluster-fork parallel command packaged in Rocks. This CLI-based parallel command, when used in conjunction with queries against the MySQL database, is designed to provide comprehensive management functionality.

**HPC packages and middleware in Rocks**

Rocks, when installed with the HPC Roll and the Kernel Roll, installs many HPC cluster packages and automatically compiles typical MPI libraries such as MPICH and MPICH-GM (MPICH for GM environments). Benchmark software such as Linpack and IOzone are also included and installed for ready use.

Rocks 3.3.0 provides several Roll CDs that can be used to add functionality to the cluster. The Intel Compiler Roll installs all relevant Intel compilers. The Condor Roll installs the Condor job scheduler and the relevant packages, and the Sun Grid Engine Roll installs the Sun-sponsored Sun Grid Engine scheduler packages. A list of current Rolls and their intended functionality can be found at the Rocks Web site. In addition, Rocks allows administrators to create custom Rolls that include their own specific packages, thereby enabling organizations to flexibly adapt Rocks for their individual IT environments.

With the acceptance of NPACI Rocks in the cluster community, vendors are in the process of providing commercial, supported versions of NPACI Rocks. Platform Rocks, which is being developed by Platform Computing, is based on NPACI Rocks. Platform Rocks is a hybrid software stack featuring a blend of market-leading open source software technologies and proprietary products.13

Platform Rocks can be obtained from Platform Computing with an annual support subscription, which provides support, maintenance, fixes, and other value-added services. Platform Rocks includes the supported Red Hat OS version, as compared to the recompiled Red Hat OS used by NPACI Rocks.

Platform Rocks has been verified and validated on 8- to 256-node clusters of the latest-generation Dell HPC servers—including the Dell PowerEdge 1850 server, PowerEdge SC1425 server, and PowerEdge 1855 blade server—using both x86 technology and Intel Extended Memory 64 Technology (EM64T), Red Hat operating systems, and Myrinet and InfiniBand interconnects.

Rinku Gupta is a systems engineer and advisor in the Scalable Systems Group at Dell. Her current research interests are middleware libraries, parallel processing, performance, and interconnect benchmarking. Rinku has a B.E. in Computer Engineering from Mumbai University in India and an M.S. in Computer Information Science from The Ohio State University.

Yung-Chin Fang is a senior consultant in the Scalable Systems Group at Dell. He specializes in cyberinfrastructure resource management and high-performance computing. He also participates in open source groups and standards organizations as a Dell representative. Yung-Chin has a B.S. in Computer Science from Tamkang University in Taiwan and an M.S. in Computer Science from Utah State University.

Munira Hussain is a systems engineer in the Scalable Systems Group at Dell. Her current research interests are in the areas of interconnects, IA-64 and EM64T, life sciences applications, and Linux and Microsoft “Windows” software stacks for high-performance computing. She has a B.S in Electrical Engineering and a minor in Computer Science from the University of Illinois at Urbana-Champaign.

---

13 For more information about Condor, visit www.cs.wisc.edu/condor.
14 For more information about the Sun Grid Engine, visit gridengine.sunsource.net.
15 For more information about Platform Rocks, visit www.platform.com.
16 For test results, visit www.rocksclusters.org/rocks.register.
17 For more information about InfiniBand, visit www.topspin.com.