Scyld Beowulf Clustering for High Performance Computing

The Scyld Beowulf Cluster Operating System is a Linux-based software solution for scalable performance computing systems. It solves many of the problems associated with first generation Beowulf clusters while simultaneously reducing the costs of system installation, administration and maintenance. With the Scyld Beowulf Cluster Operating System, the cluster is presented to the user as a single, large scale parallel computer. This white paper describes the technology and benefits of this software solution.

What is a Beowulf Cluster?

Beowulf is a multi-computer architecture which can be used for parallel computations, server consolidation or computer room management. A Beowulf cluster is a computer system conforming to the Beowulf architecture, which consists of one master node and multiple compute nodes. The nodes, typically commodity off-the-shelf servers or server blades, are connected together via a switched Ethernet network or a dedicated high-speed interconnect network such as Infiniband. The nodes usually do not contain any custom hardware components and are trivially reproducible. The master node controls the entire cluster and serves parallel jobs and their required files to the compute nodes. The master node is the cluster’s administration console and its gateway to the outside world. It assumes full control of the compute nodes from software provisioning to monitoring to job execution. Compute nodes do not need keyboards, monitors or even a hard disk. Simply put, Beowulf is a technology of clustering Linux computers together to form a parallel, virtual supercomputer, a beowulf cluster. While Linux-based beowulf clusters provide a cost-effective hardware alternative to the traditional supercomputers for high performance computing applications, the original software implementations for Linux beowulfs were not without their problems.

Issues with First Generation Beowulf Clusters

The original beowulf software environments were implemented as downloadable add-ons to commercially available, royalty-free Linux distributions. These Linux distributions included all of the software needed for a networked workstation: the kernel, various Linux utilities, the GNU software suite and many add-on packages. The downloadable Beowulf add-ons included several programming environments and development libraries as individually installable packages. In this Beowulf scheme, every node in the cluster required a full Linux installation and was responsible for running
its own copy of the kernel. This requirement combined with the lack of a uniform, commercially supported distribution created many administrative headaches for the beowulf system administrators. For this reason, early Beowulf systems were deployed by the software application developers themselves and required detailed knowledge to install and use. The Scyld Beowulf Cluster Operating System solves these and many other problems plaguing the first generation beowulf clusters.

Scyld Beowulf

The Scyld Beowulf Cluster Operating System streamlines the process of configuring, administering, running and maintaining a beowulf cluster. It was developed with the goal of providing the operating system software infrastructure for enterprise-class beowulf clusters. Scyld Beowulf features simplified cluster integration and setup, minimal and easy system administration, increased reliability, and seamless cluster scalability. In addition to its technical advances, Scyld Beowulf is a complete Linux distribution that provides a standard, stable, commercially supported platform for deploying advanced cluster applications.

Key Features of Scyld Beowulf

The following list summarizes the key features available with Scyld Beowulf:

- **Easy Installation** - The Scyld Beowulf installation procedure is identical to the standard Red Hat Linux desktop installation with one additional dialog screen for configuring the network interface on the master node.
- **Install Once, Execute Everywhere** - A full installation of Scyld Beowulf is only required on the master node. Compute nodes get all required binaries from the master node during their boot-up procedure and during process migration.
- **Execution Time Process Migration** - Scyld Beowulf stores applications on the master node. At execution time, processes are migrated from the master node to the compute nodes. This approach virtually eliminates the risk of “version skew”.
- **Seamless Cluster Scalability** - Scyld Beowulf seamlessly supports the dynamic addition and deletion of compute nodes to/from the cluster without modification to existing source code and/or configuration files.
- **Administration Tools** - Scyld Beowulf includes simplified tools for performing cluster administration and maintenance. Both graphical user interface (GUI) and command line interface (CLI) tools are supplied.
- **Fully Supported** - Scyld Beowulf is fully supported by Scyld Software.

Scyld Beowulf, Technical Summary

Built using the same source as Red Hat Linux distribution, Scyld Beowulf presents a more uniform system image of the entire cluster to both users and applications through extensions to the kernel. A guiding principle of these extensions is to have little increase in kernel size and complexity and more importantly, negligible impact on individual processor performance. In addition to its enhanced Linux kernel, the Scyld Beowulf distribution includes improved libraries and utilities specifically designed for high-performance computing applications. The latest 29-series release is compatible with Red Hat ES3 libraries and utilities.
Beowulf Cluster Process Migration Technology

Scyld Beowulf is able to provide a single system image through its use of BProc, the Beowulf distributed process management kernel enhancement, which was developed specifically for Scyld Beowulf. All processes are initiated on the master node and are migrated to a designated compute node using BProc’s process migration mechanism. BProc enables the processes running on cluster compute nodes to be visible and manageable on the master node. The parent-child process relationships and UNIX job control information are both maintained with migrated tasks. Because cluster compute nodes are not required to contain resident applications, their hard disks are available for application data and cache. This approach eliminates both the need to have full installations on the compute nodes and the version skew problem common with previous generation cluster architectures.

![Architecture of a Scyld Beowulf cluster](image)

Compute Node Boot Procedure

Scyld Beowulf supports a large variety of booting media: floppy disk, CDROM, Linux BIOS, flash disk, hard-disk and PXE-boot. The preferred booting system for the nodes is PXE-boot. The compute nodes in a Scyld Beowulf cluster boot using a two-stage procedure. Compute nodes begin their boot process using a minimal boot image, after which they contact the master node to obtain their final image. For older hardware without PXE boot capabilities, the initial boot images are approximately 1Mbyte in size (it will fit on a floppy) and contain a minimal Linux kernel with just enough functionality to configure a reliable TCP/IP socket connection between the compute node and the master node. For newer hardware with PXE boot capabilities, the initial stage is replaced by the PXE mechanism. After this initial booting stage, the compute node attempts to communicate with the master to obtain its required runtime files and complete its initialization.
procedure. The compute node uses the Reverse Address Request Protocol (RARP) to contact the master node. This RARP request, which is broadcast on all network interfaces until a reply is received, includes the compute node’s ethernet MAC address. After the master node validates the compute node’s ethernet MAC address, it “accepts” the node into the cluster and replies with the node’s IP address and a fully functional kernel. Finally, the shared, runtime libraries are cached on the compute node from the master and two user daemons (bpslave and sendstats) are started on the compute node. Using BeoPXE server, Scyld Beowulf is capable of provisioning a new node in few seconds - the majority of the boot-up time is spent on the BIOS probing the hardware.

Major Software Components

The following is a list of the major software components included with the Scyld Beowulf Cluster Operating System:

- bproc - the Beowulf process migration technology; an integral part of Scyld Beowulf
- beosetup - a GUI interface for configuring the cluster
- beostatus - a GUI interface for monitoring cluster status
- beostat - a text-based tool for monitoring cluster status
- beoboot - a set of utilities for booting the compute nodes
- beofdisk - a utility for remote partitioning of hard disks on the compute nodes
- beoserv - the beoboot server; it responds to compute nodes and serves the boot image
- bpmaster - the bproc master daemon; it only runs on the master node
- bpslave - the bproc compute daemon; it runs on each of the compute nodes
- bpstat - a bproc client; it maintains status information for all nodes in the cluster
- bpctl - a bproc client; a command line mechanism for controlling the nodes
- bpsh - a bproc client; a replacement utility for “rsh” (remote shell)
- bpcp - a bproc client; a mechanism for copying files between nodes
- beompi - an optimized implementation of the Message Passing Interface
- beopvm - the Parallel Virtual Machine; optimized for use with Scyld Beowulf
- mpprun - a parallel job creation package for Scyld Beowulf
- perf - support for platform specific hardware performance counters

Scyld Beowulf vs. original Beowulf Comparison Chart

The following chart compares some of the major features between first-generation beowulf clusters and clusters running the Scyld Beowulf Cluster Operating System.
### Future Enhancements

The following list briefly describes some of the enhancements planned for future releases of Scyld Beowulf. For more detailed information, contact Scyld Software.

- **Master Node Redundacy** - Scyld is planning to update the operating system to support multiple master nodes in the cluster. This feature will allow the transparent switchover to a backup master node in the event that the original master node suffers a failure.

- **Checkpointing** - This Scyld Beowulf feature entails the periodic saving of the state of jobs executing on compute nodes. Interrupted jobs will be automatically resumed on another compute node in the event the original node experiences a disruption of service.

- **Web-based Remote Administration** - Scyld will provide HTML browser based implementations of its BeoSetup and BeoStatus tools for performing cluster administration, maintenance and monitoring.

- **Fully Integrated Diagnostics Package** - This feature will allow administrators to diagnose hardware problems to specific pieces of hardware within a node thus reducing the time required for cluster maintenance activities.

- **Port to Intel’s Itanium Architecture** - Current releases of Scyld Beowulf support both the Intel x86 and the AMD x86_64 architectures. Support for the Intel’s Itanium2 architecture is under development.

### About Scyld Software

Scyld Software develops and supports Linux beowulf cluster operating systems for high performance computing solutions. Scyld is a Penguin Computing company with the division headquarters in Annapolis, Maryland. Contact Scyld Software by phone at (410) 990-9993 or on the web at www.scyld.com.

*Copyright ©2001-2004 Scyld Software. All rights reserved. “Scyld” and “Scyld Beowulf” are registered trademarks of Scyld Software. Beowulf, when used to describe computing machinery, is a descriptive service mark of Linux International. Red Hat is a registered trademark of Red Hat, Inc. All other products or services, service marks, registered trademarks, or registered service marks mentioned in this document are the trademarks, service marks, registered trademarks, or registered service marks of their respective owners.*