Scyld Beowulf Series 30

Installation Guide

Scyld Software

Scyld Beowulf Series 30: Installation Guide

by Scyld Software

Series 30cz-1 Edition

Published February 2006

Table of Contents

Preface

Congratulations on purchasing the most scalable and configurable Linux clustering software on the market, *Scyld Beowulf*™. This guide describes how to install the *Scyld Beowulf* software and have a scalable cluster running in just a few minutes.

While this document contains all of the information needed to get your system running, additional guides and reference manuals are available. The *Administrator's Guide* and *User's Guide* describe how to configure, use, maintain and update the cluster. The *Programmer's Guide* and *Reference Guide* describe the commands, architecture and programming interface for the system. All of the documentation may be viewed using a browser from the last CD in the set (this CD is an autorun CD, and should bring up a browser on either Windows or Linux).

Feedback

We welcome any reports on errors or difficulties that you may find. We also would like your suggestions on improving this document. Please direct all comments and problems to: support@scyld.com.

When writing your e-mail, please be as specific as possible, especially with errors in the text. Please include the chapter and section information. Also, mention in which version of the manual you found the error. This version is *Series 30cz-1*, published February 2006.

Chapter 1. Scyld Beowulf System Overview

The *Scyld Beowulf* Series 30cz-1 streamlines the processes of configuring, running, and maintaining a Linux cluster using a group of off-the-shelf computers connected through a private network. The front-end "head node" computer in the cluster distributes computing tasks to the other machines known as the "compute nodes", in a parallel architecture.

System Components and Layout

The head node is configured with a full Linux installation. Each machine in the cluster is installed with a Network Interface Controller (NIC) communicating with an internal cluster network. In order for the head node to communicate with an outside network, it needs two NICs, one for the private internal cluster network, and the other for the outside network. It is suggested that the head node be connected to an outside network so multiple users can access the cluster from remote locations as shown in Figure 1-1.

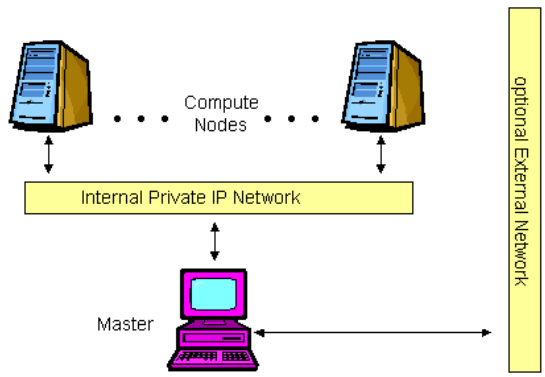

Figure 1-1. Cluster Configuration

In addition to the NIC(s), a booting mechanism is needed. The compute nodes do not boot a full distribution themselves. Instead they boot from the network using PXE boot, or optionally from a CD or floppy disk, or some other supported media that contains a small boot image (see the *Administrator's Guide* for a list of compute node boot options).

For any Beowulf system, hardware selection is based upon the price/performance ratio. Scyld recommends the following components for use with this release of the *Scyld Beowulf* distribution:

Recommended Components

Processors

Intel® Pentium® IV, Intel® Xeon®, AMD Opteron™, single- or multi-core.

Architecture

one, two, or four sockets per motherboard

Physical Memory

1,024 MBytes (1 GByte) or more preferred, minimum 512 MBytes

Operating System

Scyld Beowulf (this release)

Network Interface Controllers (NIC)

Gigabit Ethernet (Fast Ethernet minimum) PCI—X or PCI-Express adapters (with existing Linux driver support) in each node for the internal private IP network. The head node requires an additional NIC for connecting the cluster to the external network. This NIC should be selected based on the network infrastructure (e.g., Fast Ethernet if the external network you are connecting the cluster to is Fast Ethernet). For a list of the latest supported NICs contact Scyld.

Network Switch

All compute nodes, and the head node private network NIC, must be connected to a non-blocking Gigabit Ethernet switch for the internal private network. At a minimum, the network switch must match the speed of the network cards. Note that the switch is a critical component for the correct operation and performance of the cluster. In particular, the switch must be able to handle all of the network traffic over the private interconnect, including cluster management traffic, migrating processes, transferring libraries, and storage traffic, and it must also properly handle DHCP and PXE.

Note: it is sometimes confusing to identify which NIC is connected to the private network. Take care to connect the head node to the private switch through the NIC with the same or higher speed than the NICs in the compute nodes.

Drives

For the head node, we recommend using either SATA or SCSI disks in a RAID 1 (mirrored) configuration. The operating system on the head node requires approximately 3 GB of disk space. We recommend configuring the compute nodes without local disks (diskless).

If the compute nodes do not support PXE boot, a floppy (32-bit architectures) or bootable CD-ROM drive (32- and 64-bit architectures) is required. If local disks are required on the compute nodes, we recommend using them for storing data that can easily be re-created, such as scratch storage or local copies of globally-available data.

If you plan to create boot CDs for your compute nodes, your head node requires a CD-RW or writable DVD drive.

Optional Hardware Components

Gigabit Ethernet with a non-blocking switch serves most users. However, some applications benefit from a lower-latency interconnect. Infiniband is an industry standard interconnect providing low latency messaging support, as well as IP and storage support. Although higher cost than Gigabit Ethernet, Infiniband can be configured as a single universal fabric serving all of the cluster's interconnect needs. More information about Infiniband may be found at the Infiniband Trade Association web site (http://www.infinibandta.org). Scyld supports Infiniband as a supplemental messaging interconnect in addition to Ethernet for cluster control communications.

Assembling the Cluster

The full Scyld Beowulf Scalable Computing Distribution is only installed on the head node. A graphic utility (`BeoSetup`) is available and included on the head node to aid in the cluster configuration process.

Most recent hardware supports network boot (PXE boot), and Scyld recommends the use of PXE boot for booting compute nodes. For nodes that do not support network boot, each compute node requires a floppy disk or CD-ROM drive, with suitable boot media inserted before being powered up.

Software Components

A brief description of the major portions of the *Scyld Beowulf* distribution is given below. For more information, see the

Administrator's Guide and the *User's Guide*.

- *BeoSetup*: A graphic utility for configuring and administering the Scyld Beowulf cluster.
- *BeoStatus*: A graphic utility for monitoring the status of the Scyld Beowulf cluster.
- *Scyld BeoMaster* The *Scyld BeoMaster* software is an integral part of the *Scyld Beowulf* distribution. It allows processes to be started on compute nodes in the cluster and tracked in the process table on the head node. *BeoMaster* also provides process migration mechanisms to help in the creation of remote processes and removes the need for most binaries on the remote nodes.
- *MPICH/LAM*: Message Passing Interface, modified to work with Scyld Beowulf cluster software.
- *Beowulf utilities*: several utilities to control and display nodes and processes in the Scyld Beowulf cluster.

Chapter 2. Quick Start Installation

The Scyld Beowulf distribution is provided as a set of three CD-ROM discs which include the basic Linux operating system distribution as well as Scyld Beowulf cluster software. The first CD in the series (disc 1) is bootable, and is used to initiate the install on the head node. The last disc in the series also contains product documentation which may be read directly from the disk on any running PC or workstation.

This chapter outlines two simple cases: installing on a head node with network-booted compute nodes, and installing on a head node where the compute nodes must be booted from local media. Refer to Chapter 3, Graphical Installation for other scenarios.

Other machines join the cluster as compute nodes, and require no explicit installation. They boot either by obtaining a boot image over the network using PXE, or with boot media (floppy or CD-ROM) that converts them to a Scyld-developed network boot system. If your hardware does not support PXE boot, a bootable floppy or CD-ROM may be created using the BeoSetup utility on the head node.

Install the Head Node

If you need more information on any of the following steps, the installation procedure is fully documented in Chapter 3, Graphical Installation.

1. Boot the front-end (head node) machine from the Scyld Beowulf distribution CD-ROM labeled Disc 1. The graphical installation process starts after 20 seconds.
2. Follow the on-screen instructions to execute the installation. The first few screens set basic elements, including the default language, keyboard and mouse. For most screens you may accept the defaults.
3. If you need to set up hard disk partitions or bootloader defaults differently from the default, please see the detailed instructions in Chapter 3, Graphical Installation.
4. You must configure the networks, as described here and in Chapter 3, Graphical Installation.

   Tip: To proceed with configuring the network, you must know which interface is connected to the public network and which is connected to the private network.

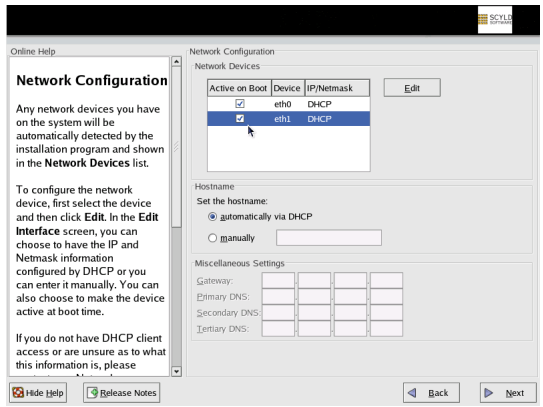

Figure 2-1. Network Interface configuration

For eth0 (or the interface connected to the public network), *DHCP* is selected by default. If your external network is set up to use static IP addresses, select this interface and click **Edit**—your network administrator should provide you with the IP address). Set the *IP Address* and *Netmask*, then click **OK**. If you set a static IP address for the public interface, you must also click *manually* for *Set the hostname* and provide a hostname, gateway and primary DNS IP addresses.

Caution

Note: For eth1 (or the device connected to the private cluster network), *do not* select DHCP. You must select and edit this interface and manually set the IP address (see Figure 2-2). Also check the *Activate on Boot* box to make the specific network device initialized at boot-time.

Note that the head node also functions as a PXE and DHCP server for the cluster. On the Firewall page following, ensure that the interface is set as a trusted interface (check the box under "Allow all traffic to pass for this interface").

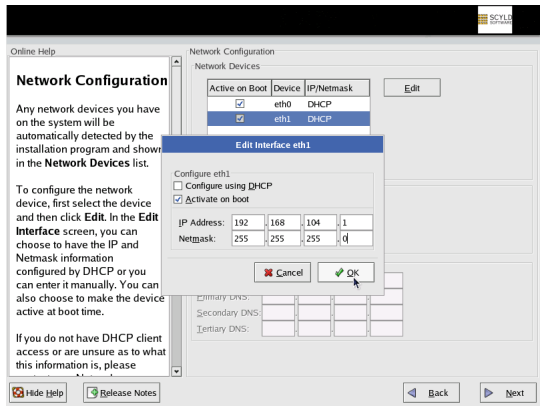

Figure 2-2. Manually set IP Address

For eth1 (or interface connected to the internal private cluster network), you must not choose a dynamic IP address, and we recommend choosing a non-reroutable address (such as 192.168.x.x or 10.x.x.x). Once you have specified the *IP Address*, you must also set your *Netmask* based on the address. Click **OK** to return to the screen described in Figure 2-1.

Configure the network settings for all of the devices listed. Click **Next** to continue.

5. Proceed through the dialog boxes to configure a firewall, additional language support, and your timezone.
6. You must supply a root password. Refer to Chapter 3, Graphical Installation. An alphanumeric password of at least 8 characters, with special characters is recommended.
7. Review the packages to be installed.
8. At this point, the system is ready to install the software you configured in earlier steps. The installer prompts for additional disks as necessary.
9. After the software is installed to the hard drive, you are prompted to verify the video hardware determined when the installer probed the system. If the default selections are incorrect, see Chapter 3, Graphical Installation. The system prompts you to reboot when necessary.
10. On the subsequent boot, a *Welcome* screen appears with the Scyld Software logo. The next screen enables you to set up your Scyld Beowulf cluster by choosing which ethernet interface to use, choosing the number of nodes in your cluster, and establishing the initial IP address for the cluster. Choose an IP numbering system that will encompass your entire cluster, with room for expansion. For example, you might use the range 192.168.104.10 through 192.168.104.50 for a 30-node cluster, with room to add 10 more nodes later.

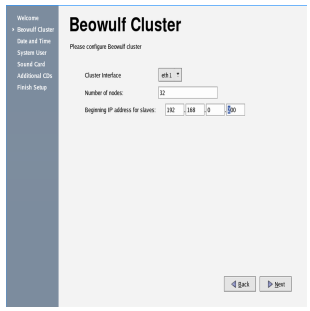

Figure 2-3. Manually set IP Address

11. Continue through the installation screens to set the date and time, set up a system user, and verify the sound card, if one is installed. Scyld Beowulf is now installed and set up on your system. Next, boot and configure the compute nodes to get your cluster up and running.

    Note: if prompted to restart Beowulf services, click *Yes*.

Boot and Configure the Compute Nodes

1. If you are not logged in as root already, log into the head node using the root username and password set up earlier. Start the cluster configuration tool, **beosetup**, by clicking on the Beosetup "Building Blocks" icon at the bottom of the screen (hover the cursor over the tray icon that looks like triangle of yellow blocks, then click). Note, if **beosetup** fails to start, check the syslog for possible errors. You can manually start it by typing **/usr/bin/beosetup** in a terminal window.
2. If your compute nodes can't PXE boot, or if for some reason you don't want them to use PXE, you may create compute node disks. *You only need to perform this step if you want to boot compute nodes from boot media.* You may create either boot floppy disks, or boot CD-ROM disks. You need a CD-RW drive installed on the head node to create boot CDs, and CD-ROM drives on each compute node in order to use them.
    a. Click **Node Floppy** or **Node CD** in the *beosetup* window.
    b. Click **OK** in the *Create BeoBoot Floppy* or **Create BeoBoot CD Image** window to write the node boot image. This image consists of a basic boot image which first boots the node and then downloads the full compute node boot image.
    c. Click **Close** to close the window.
    d. Write the boot image to blank media, and then place the boot media in each compute node. Repeat for each compute node.
3. Boot the compute nodes by powering them on in the order you want them to be numbered, typically one by one from the top of a rack down (or from the bottom up). You can reorder nodes later if necessary (see the Administrator's Guide).

As compute nodes join the cluster, they are listed in `beosetup` by Ethernet Station (MAC) Addresses and given node numbers in the order they initially contact the head node. After installation is complete, this order may be manually changed, but powering up nodes in the desired order is easier and is recommended.

4. The nodes transition through the boot phases and after a few seconds be shown in the **up** state in `beosetup`. The cluster is now fully operational with diskless compute nodes.

Status of the compute nodes is listed in the `beosetup` window. All compute nodes should show status of *`up`* when ready for use. Note: an *`error`* state may be encountered due to lack of a partition table.

Chapter 3. Graphical Installation on the Head Node

This chapter guides you through the graphical installation of the Scyld Beowulf software. This software installation is intended for the first computer ("node") of the Scyld Beowulf cluster, which functions as the "head node" in the cluster, controlling and monitoring other nodes as well as distributing jobs. It is assumed that you are familiar with the concepts outlined in the previous chapters and that you have correctly assembled the hardware for your Scyld Beowulf cluster. If this is not the case, please refer to the previous chapters to acquaint yourself with the Scyld Beowulf software and then verify that your hardware configuration is set-up properly.

Each screen displayed within the graphical installation sequence has an online help panel on the left and an installation dialog panel on the right. You can scroll through the online help before answering the questions.

Starting the Graphical Installation

This section helps you get your Scyld Beowulf graphical installation running. If your machine does not boot from the CD-ROM, you must change the settings in the machine's BIOS. Refer to your computer's reference manual if you need instruction regarding how to change settings in the BIOS.

Booting the Head Node

> **Caution**
>
> Installing Scyld Beowulf over another Linux or Scyld Beowulf installation does not preserve any existing system files or data. You must exit the Scyld Beowulf installer, and backup any important data which you wish to save by a suitable means. Or, if you are experienced, when asked in the the Section called *Disk Partitioning*, select "Manually partition" as your disk partitioning strategy.

Running the graphical installation on the head node is just one of the uses for the Scyld Beowulf CD-ROM installation set. It may also be used to perform a text-mode installation. When you boot the head node with the Scyld Beowulf CD-ROM (Disc 1), you have 20 seconds to select the preferred installation mode, graphical or text. The CD-ROM defaults to installing in graphical mode. The selection screen is shown in Figure 3-1.

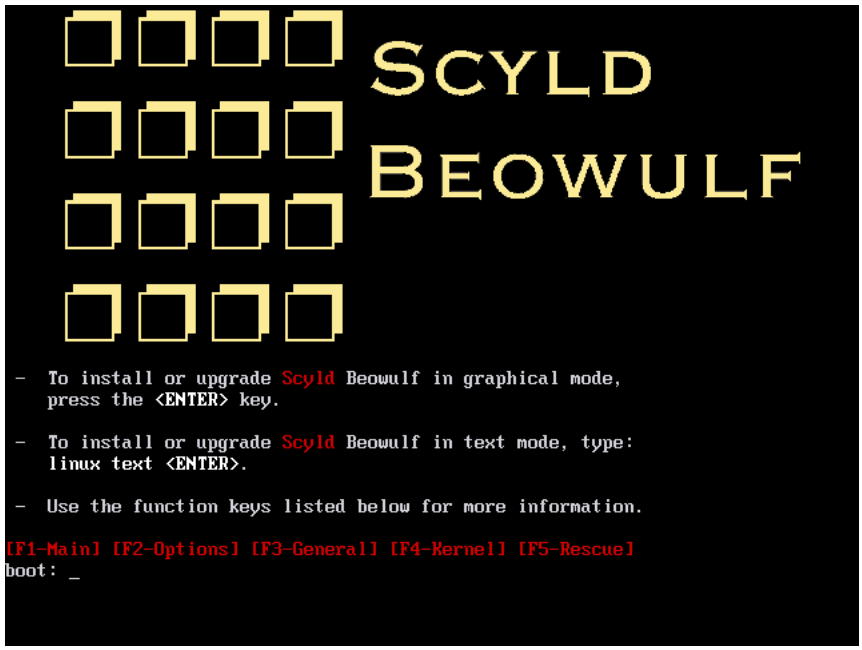

Figure 3-1. Scyld Beowulf CD-ROM Selection Screen

Selecting the Graphical Installation

1. Insert the Scyld Beowulf CD-ROM into the CD-ROM drive on the head node. Restart the system by powering it on or resetting it.
2. When the installation selection screen appears (see Figure 3-1) and prompts you with `boot:`, type **Enter** to start the graphical installation. Linux boots from the CD-ROM. The installer may prompt you to test the installation media, or to skip the test. You can make either choice; if you choose Ok, the test takes a few minutes to complete.

   The Anaconda-based installer runs. This may take a few minutes. Text messages scroll on the screen, then the video system is probed, which may cause the screen to flash.

   Finally, the *Welcome* screen appears.

There are several options that may be selected from this screen other than the graphical install documented in this chapter. *It is normally not required that these options be used. They are provided for experienced Linux administrators, or for use at the direction of your support representative.*

- *text mode*: to select text mode install, type **linux text**.
- *F1–Main*: returns you to the installation selection screen.
- *F2–Options*: describes some Installer Boot Options. Some of the options available are:
  - To disable hardware probing, type **linux noprobe** at the `boot:` prompt. This is useful if installation fails because the installer can’t probe the hardware correctly.
  - To test your installation media, type **linux mediacheck** at the `boot:` prompt. This is useful if installation fails because the installer can’t read the installation CD.

  - To enter rescue mode, type **linux rescue** at the `boot:` prompt. This is more fully documented under the *F5–Rescue* item below.
  - If you have a driver disk, type **linux dd** at the `boot:` prompt. This is useful if you must provide a driver that is not included on the standard Scyld installation media, for example for a hard disk or RAID controller on which you want to install the operating system.
  - To prompt for the install method being used, type **linux askmethod** at the `boot:` prompt.
  - To use an installer update disk, type **linux updates** at the `boot:` prompt.
  - To force the use of the lowest graphical screen resolution, type **linux lowres** at the `boot:` prompt. This is useful if your monitor, video cable, or a monitor connection involving a KVM can't handle a higher screen resolution even though the video card and monitor report they can handle a higher resolution.

- *F3–General* provides some additional advice for working around installation failures.
- *F4–Kernel* provides some help with parameters that may be passed to the kernel as it boots. The options described under *F2* above are specific examples of the general method described here. Take care using this option unless you are an experienced Linux administrator.
- *F5–Rescue* invokes the Scyld Beowulf rescue mode. Rescue mode includes some utilities that are useful if the head node does not boot after installation, such as an editor you can use in examining and editing configuration files, and tools to work with hard drives. To invoke this option, type **linux rescue** at the `boot:` prompt.

Welcome to Scyld Beowulf

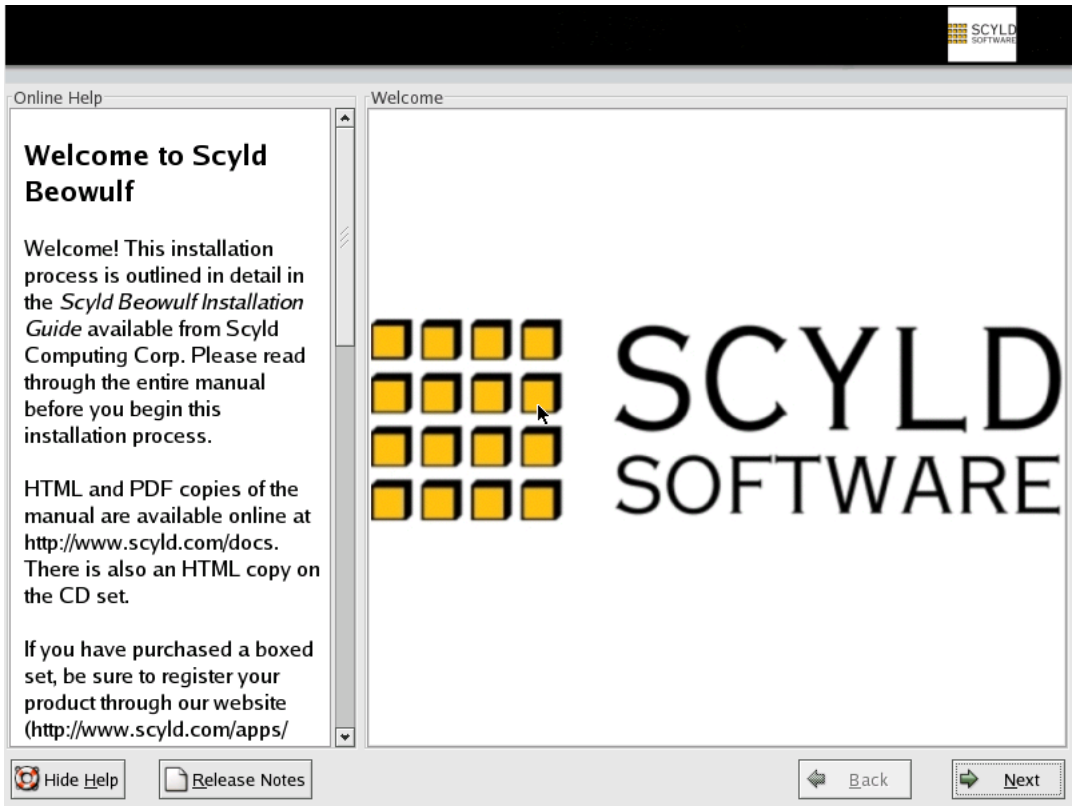

Figure 3-2. Welcome to Scyld Beowulf

The Scyld Beowulf cluster software provides its functionality through extensions to the Linux kernel and additional system libraries. The Scyld Beowulf advancements are seamlessly incorporated so that there is no loss of previous Linux capabilities. However, a full installation of the Scyld Beowulf overwrites and replace any existing Linux installation, meaning that any previous settings are lost.

The *Welcome* screen is shown in Figure 3-2. You do not need to supply any information on this screen.

This installation program is based on the "Anaconda" installer, which is used by many Linux distributions. The Scyld Beowulf head node interface is designed to appear as a standard Linux installation, thus most of the questions are the same as a workstation Linux installation. Most cluster-specific questions are at the end of this process, but there are a few that must be handled during the regular installation process, networking in particular. All Scyld Beowulf specific items are fully detailed in this installation guide.

You can look at the release notes by clicking on the **Release Notes** button on the left panel of the *Welcome* screen. You may also browse the documentation directly from the last of the installation CDs on any Linux or Windows workstation or PC. This Autorun CD launches a browser from which the entire documentation set is available.

The left-hand side of each screen presented contains **Online Help** for that specific screen. It is recommended that you read the information presented, especially if you are unsure of your selections. If you do not wish to see the *Online Help*, you may click the **Hide Help** button, which is located under the *Online Help* frame. To bring the *Online Help* back, click the **Show Help** button.

Click on **Next** when you are ready to proceed to the next installation screen.

Language Selection

Figure 3-3. Choosing Mouse Type

Choose the language for this installation. The default is English. Click **Next** when you are ready to proceed to the next installation screen.

Keyboard Configuration

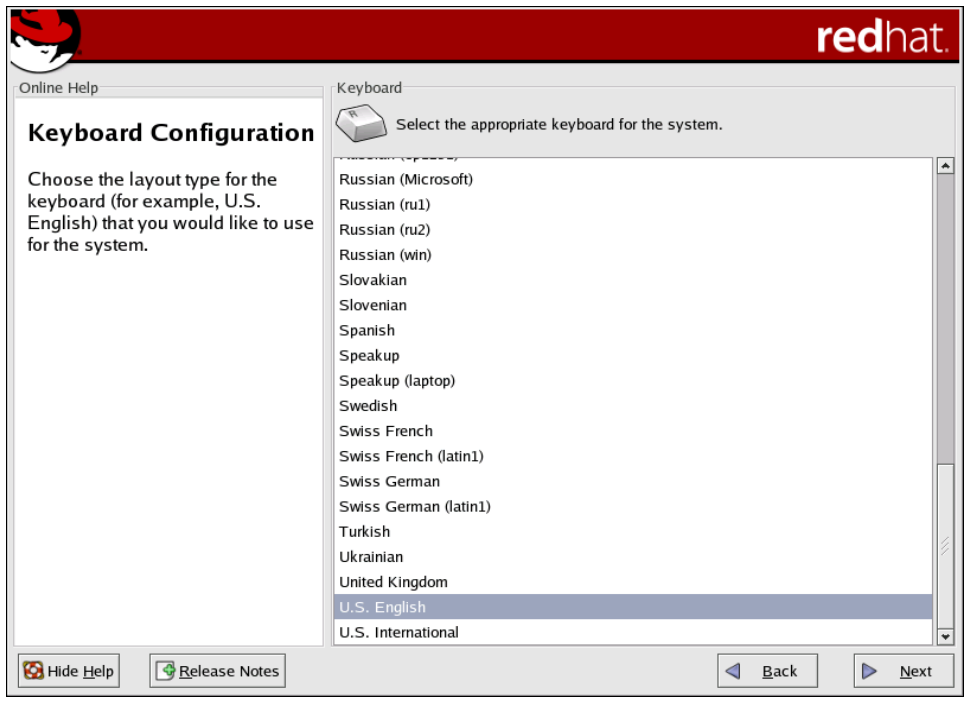

Figure 3-4. Choosing Mouse Type

The installer probes your hardware and selects the keyboard language configuration best suited to what it finds. You can change its cohice by scrolling through the choices in the windows and clicking a different language. Generally, the keyboard configuration selected by the installer is the correct one.

Click **Next** when you are ready to proceed to the next installation screen.

Choosing Mouse Type

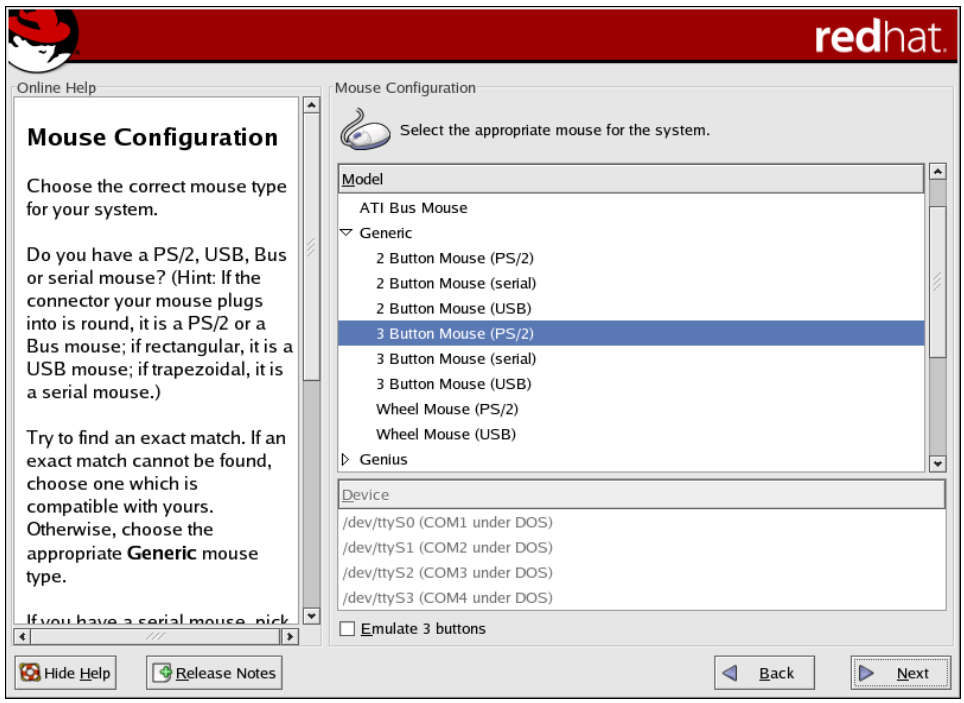

Figure 3-5. Choosing Mouse Type

The installer probes your hardware and selects the mouse type best suited to what it finds. You can change its choice by scrolling through the choices in the windows and clicking a different manufacturer and/or mouse type. Note also that the connection method (PS/2, serial, or USB) is also among the choices. Generally, the mouse selected by the installer is the correct one.

Click **Next** when you are ready to proceed to the next installation screen.

Disk Partitioning

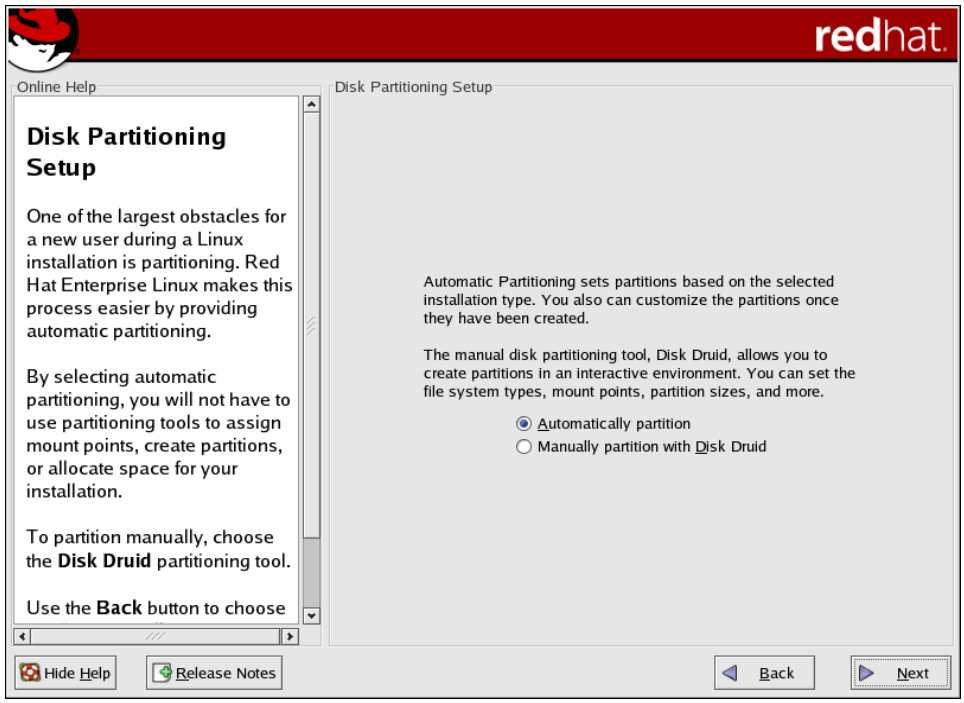

Figure 3-6. Partitioning Strategy

The purpose of this step is to establish where the information you selected in the previous step is installed on your hard drive. This involves partitioning the data into several sections, and formatting your drive accordingly using a disk parition program called Disk Druid. You have the following options:

Automatic Partition

With this option, the installer uses a pre-configured format to determine how best to utilize the machine's drive capacity, and presents the result with Disk Druid. Scyld recommends allowing the installer to automatically partition your hard drive.

Note that if you wish to preserve existing data, this option is usually unacceptable, and you should choose a manual installation instead. If you choose this method, proceed with the Section called *Automatic Partitioning*.

Manual with Disk Druid

Disk Druid provides a graphical interface with which you can choose your own partition scheme. This is recommended for advanced users only. If you choose this method, proceed with the Section called *Manual Partitioning with Disk Druid*.

Automatic Partitioning

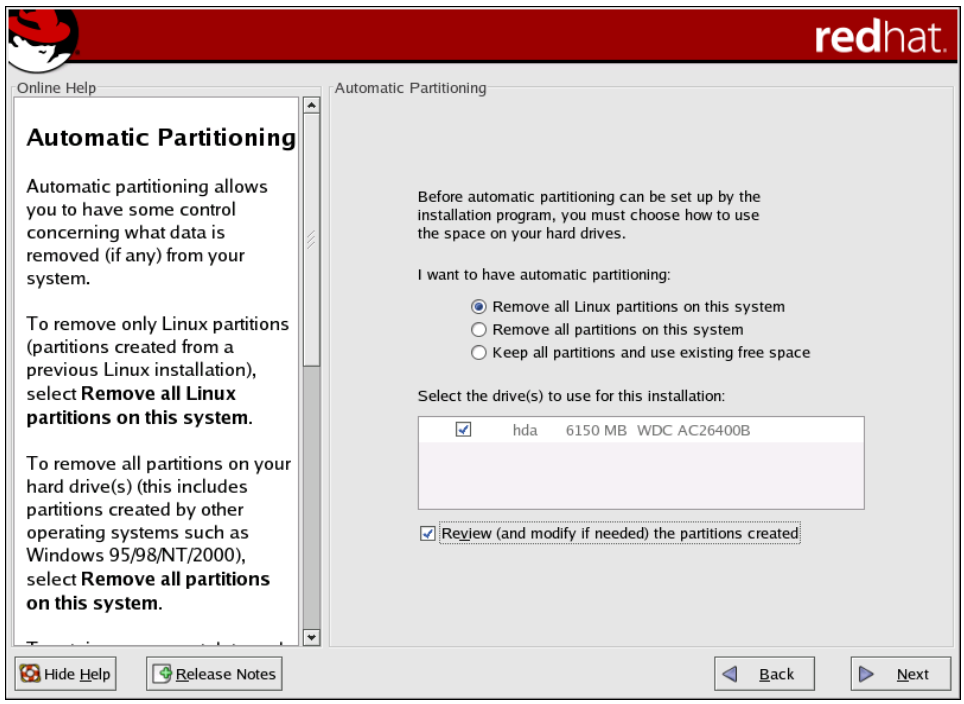

Figure 3-7. Automatic Partitioning

The automatic partition option lets you decide what device(s) are used for the installation, and what part of each device is be used. The default option is to erase any existing partition on the selected device(s) and perform an installation using the entire drive. Alternatively, you may choose to perform an installation on the currently existing available space, or on the space that would be made available by removing existing Linux partitions (but leaving non-Linux partitions, such as Windows).

The "Review" option is also selected by default, so that you may accept or refuse the configuration made by the installer prior to committing the changes to the device. If you select this option, the next screen is that given by Disk Druid, Figure 3-9, showing the suggested partitioning scheme. You can change the automatic partitioning results if desired.

You are prompted to select the hard drive to use for the installation, and to direct Disk Druid's behavior with respect to existing partitions. Check the Review box to review all partitions before they are created. Click **Next** when you are ready to proceed to the next installation screen.

Clicking **Next** may display a Warning dialog box asking you to confirm that you want to remove disk partitions. If you are sure you can safely delete the disk partitions, click **Yes** to proceed. Otherwise click **No** to return to Figure 3-7. You then see a Disk Druid screen showing the default partitioning.

If your system contains a new disk that has no partition table, the installer displays a warning like Figure 3-8.

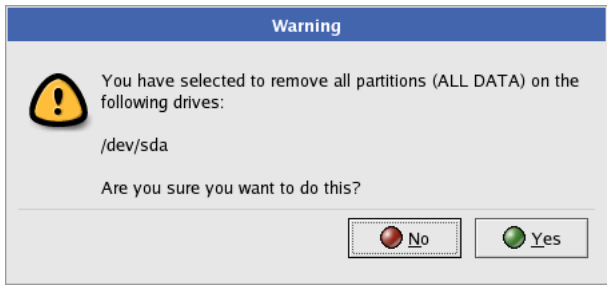

Figure 3-8. Warning-Partition Table Unreadable

If you have any doubts about whether the contents of the disk can be destroyed, click **No** to return to the screen in Figure 3-6. Otherwise, click **Yes** to erase any old data, and create a new partition table.

Manual Partitioning with Disk Druid

Manual partitioning gives you the greatest flexibility, but requires some knowledge on the part of the user. Partitioning the device amounts to creating various partitions, keeping in mind two issues:

Partition type and name

: You must define some core partitions with specific names and type, such as `/boot`, `/home`, `/` (root partition) and a swap partition.

Partition size

: Your partitions must respect some minimal size constraints so that they can hold the data, and have room for potential growth.

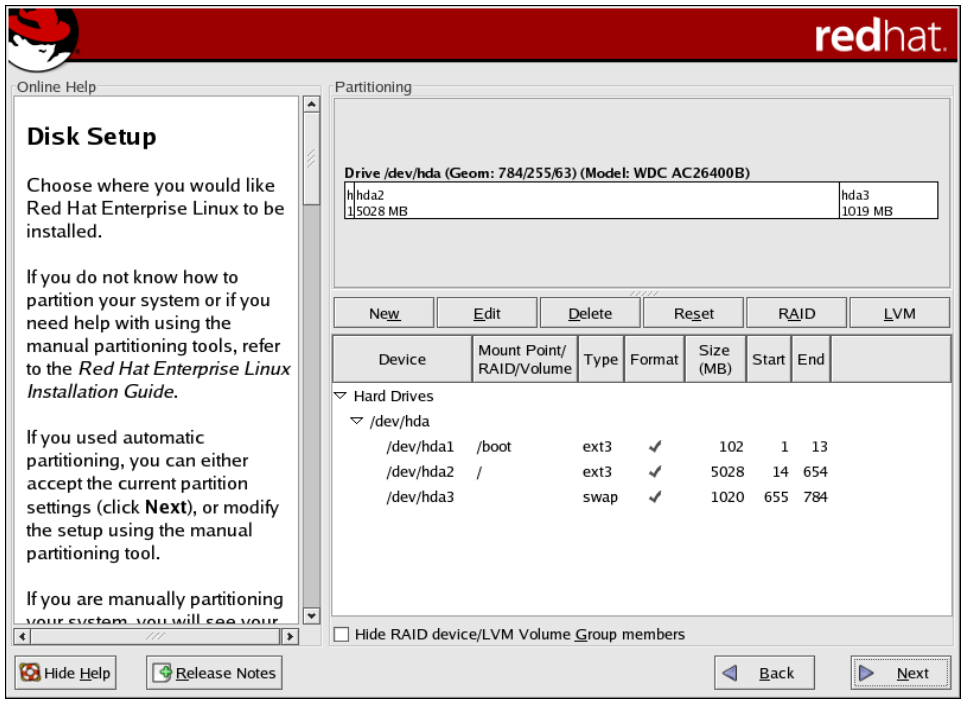

Figure 3-9. Partitioning with Disk Druid

Disk Druid's Action Buttons

Displayed between the "Partitions" section and the "Drive Summaries" are the **Disk Druid** action buttons. They are used to add and delete partitions, change partition attributes, accept changes and exit the program. Each button is listed below with its functionality.

New

Used to request a new partition. When selected, a dialog box appears requesting mount point and size for the partition. See Figure 3-10.

Edit

Used to modify attributes of the partition currently highlighted in the "Partitions" section. When selected, a dialog box appears with changeable fields. Which fields are changeable depends on whether the partition information has already been written to disk.

Delete

Used to delete the partition currently highlighted in the "Current Disk Partitions" section. You are prompted to confirm your intention to delete the partition.

Reset

Used to restore the partition table settings to its original state. Any changes that you made are lost.

RAID

Used to provide redundancy to any or all disk partitions. *This should only be used if you have experience using RAID.*

LVM

Used to configure the logical volume manager (LVM). *This should only be used if you have experience with LVM setup.*

Minimal Partitioning

Disk Druid displays the hard disks found on the system, along with the existing partitions on those disks. Select the first hard disk, usually `/dev/hda` or `/dev/sda`, and click **New**. A screen appears that is similar to Figure 3-10. Type `/boot` in the **Mount Point** box. Provide a *Size* of *at least 100 MB*. Check the box *Force to be a primary partition*. Its a good idea to also check *Check for bad blocks*. Click **OK** to create the partition.

Repeat this process to create a swap partition (select **`swap`** as the *File System Type* and choose a size *at least two times your computer's RAM*. Do *not* check *Force to be a primary partition*. Click **OK** to create the **swap** partition.

Repeat the above for Mount Points `/` and `/home`, choosing ext3 or another file system other than swap. You may also create mount points for `/var` and `/usr`, or let Linux create these within the `/`(root) partition. Generally, you need at least 4GB of space in `/`, of which at least 3GB is needed for `/usr`

To configure hardware RAID, refer to your RAID vendor's documentation. To configure software RAID or LVM, refer to Administrator's Guide or a good Linux reference. The rest of the space may be allocated to `/home`.

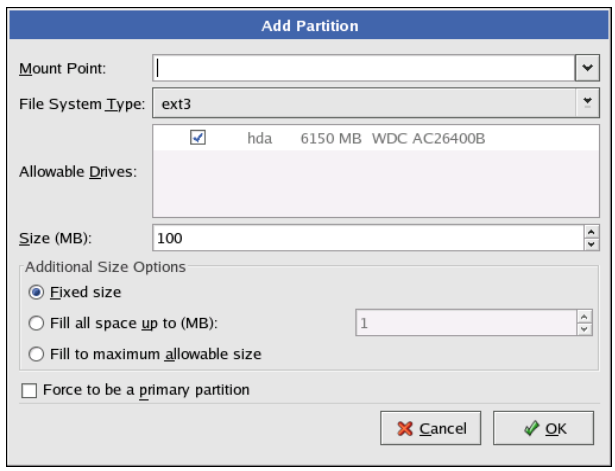

Figure 3-10. Add Partition in Disk Druid

Partitioning Problems

If **Disk Druid** can not allocate a partition, a dialog box appears which lists the unallocated partition and the reason for the failure. Lack of disk space is the most common reason. To remedy the situation, you have a few choices. You may move the partition to another drive that has sufficient space available, resize the partition (by deleting it and then re-adding a smaller one), or just delete the partition entirely. To modify the partition, double-click on it or use the **Edit** button.

Bootloader Configuration

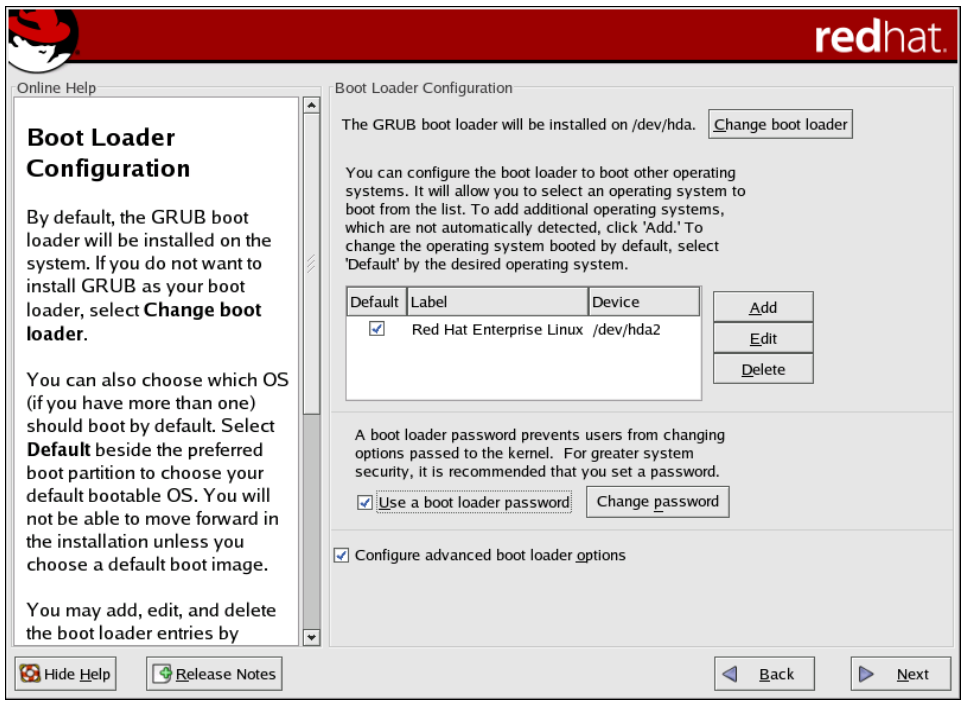

Figure 3-11. Bootloader Configuration

This section assists you in configuring the method used to boot your system. The *bootloader* is a piece of software that starts the Linux kernel or other operating systems.

The first decision is whether or not to install a boot loader. One may already exist on your harddrive, or you may want to use another device (such as a floppy) to boot the Scyld Beowulf OS. In these cases you do not want to replace the existing bootloader.

By default, *GRUB* is chosen as the boot loader (see Figure 3-11). *LILO* is the legacy bootloader, while *GRUB* is a newer bootloader which may be easier for new users. This may be changed by clicking the **Change boot loader** button (see Figure 3-12). You may select *LILO* or *GRUB*, or elect to not install a boot loader. (Most users should install a bootloader.)

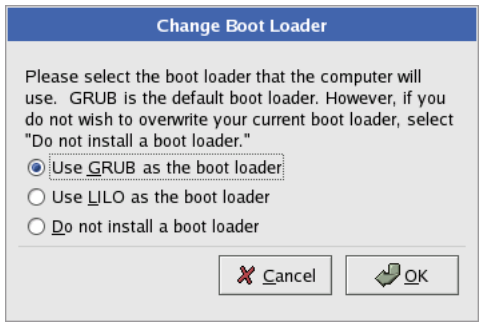

Figure 3-12. Change Boot Loader

Note that, if you select *Do not install a boot loader* and *OK* you see a confirmation dialog box (Figure 3-13) reminding you that you need to use removeable boot media (floppy or CD) to boot your master node if you proceed. You may cancel this dialog box and return to Figure 3-12.

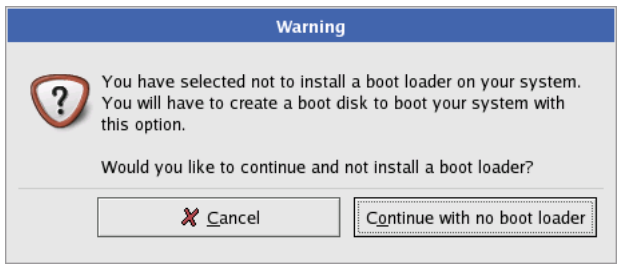

Figure 3-13. No boot loader warning

Requiring a password for the bootloader provides a higher level of system security. To set a password, select the **Use a boot loader password** checkbox. Enter a password and confirm it (see Figure 3-14).

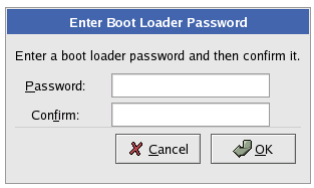

Figure 3-14. Bootloader Password

You can gain more control over the boot process by checking the *Configure advanced boot loader options* box before clicking **Next** (see Figure 3-15).

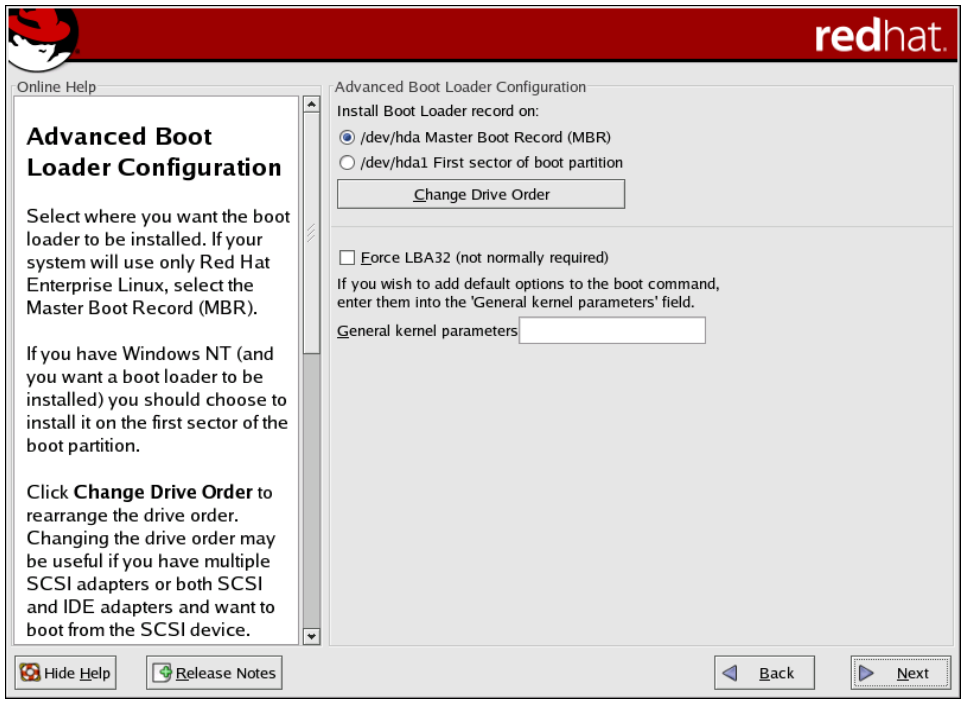

Figure 3-15. Advanced Boot Loader Configuration

The bootloader can be installed either in the master boot record (MBR) of the device you are booting from, or in the first sector of the boot partition. For help on deciding which is right for your system, refer to the online help during installation. Generally, installing on the MBR is recommended.

The *Force LBA32* option should only be used if you experience problems with large drives (see the online help).

This screen also provides a place to enter kernel parameters, which the boot loader passes to the kernel upon boot. These parameters depend on the specific kernel you are booting, and should be changed only by experienced users.

Click **Next** when you are ready to proceed to the next installation screen.

Network Configuration

A typical Scyld Beowulf cluster has one interface dedicated to the private cluster network, and one to the external network. The following setup is assuming that eth0 is the interface connected to the external network and eth1 is the interface connected to the private network. You can configure your network settings for each network device on your system.

Tip: To proceed with configuring the network, you must know which interface is connected to the public network and which is connected to the private network.

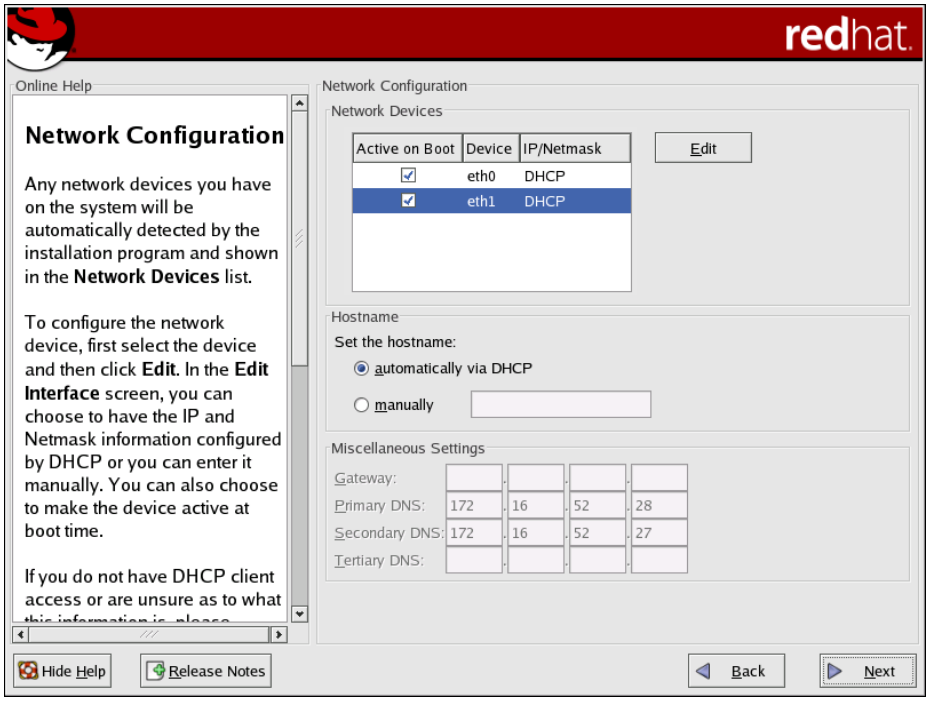

Figure 3-16. Network Interface configuration

For eth0 (or the interface connected to the public network), *DHCP* is selected by default. If your external network is set up to use static IP addresses, select this interface and click **Edit**—your network administrator should provide you with the IP address and netmask. Set the *IP Address* and *Netmask*, then click **OK** (see Figure 3-17). If you set a static IP address for the public interface, you also must click *manually* for *Set the hostname* and provide a hostname, gateway, and primary DNS IP addresses.

> **Caution**
>
> For eth1 (or the device connected to the private cluster network), you must configure the network interface manually by clicking the *Manually* radio button and assigning a static IP address and netmask to the private network interface.

For eth1, check the *Activate on Boot* box to make the specific network device initialized at boot-time.

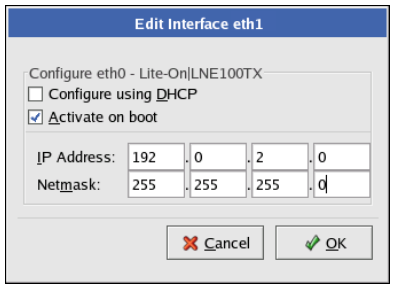

Figure 3-17. Manually set IP Address

For eth1 (or interface connected to the internal private cluster network), you must un-check *configure using DHCP* and manually set up a static IP address. We recommend choosing a non-reroutable address (such as 192.168.x.x or 10.x.x.x). Once you specify the *IP Address*, you must also set your *Netmask* based on the address. Click **OK** to return to the screen in Figure 3-16.

Configure the network settings for all of the devices listed. Click **Next** when you are ready to proceed to the next installation screen.

Network Security Configuration

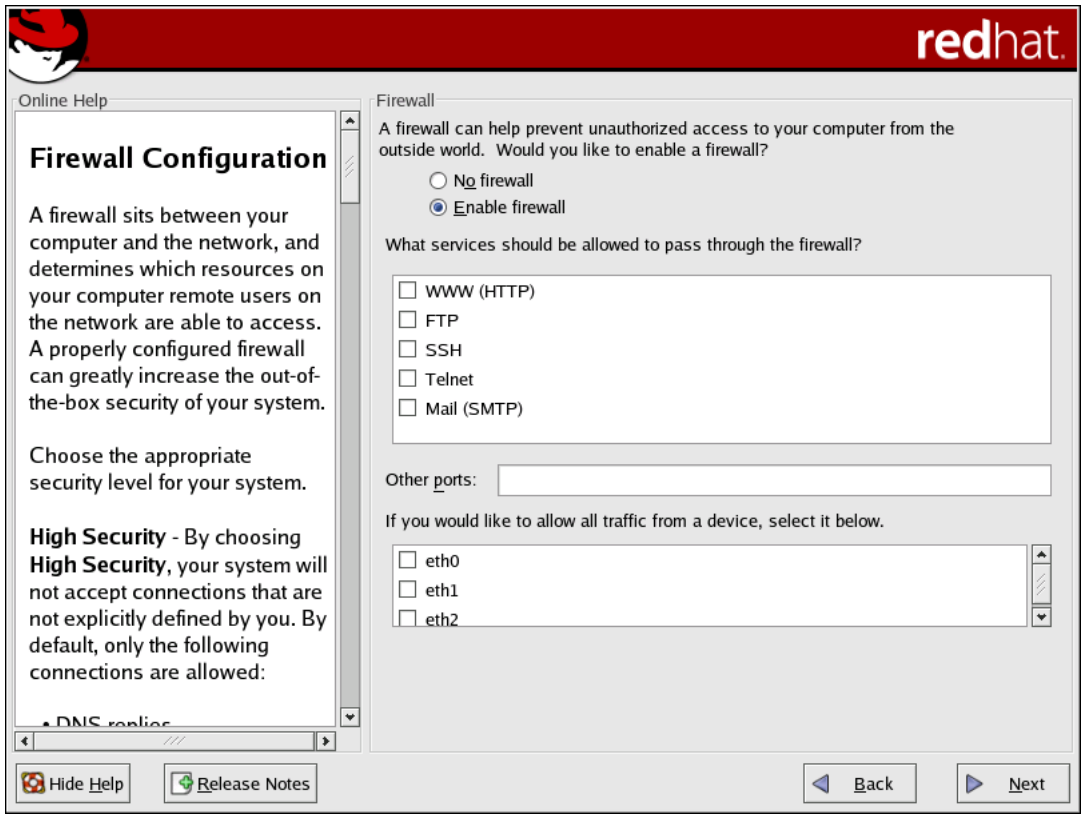

Figure 3-18. Firewall configuration

This step of the installation process allows you to customize several aspects of the firewall that protects your cluster from possible network security violations. Please note that the security features provided with this system don't guarantee a completely secure system.

The first item to consider is the *Security Level*. You may choose:

High

All incoming requests are blocked, isolating the cluster from the rest of the network.

Medium

Blocks any incoming requests from ports below 1023 as well as the NFS server port and ports used for remote X clients.

No Firewall

All connections are allowed. This option is not recommended unless you plan to configure your firewall after the installation.

The rest of the configuration is only available if you click the *Customize* radio button. Be aware that, from this screen, you can not specify different rules for different interfaces, other than trusted or not trusted. All untrusted interfaces allow the same incoming traffic.

You can select which network devices are trusted. For these devices, all traffic is accepted without regard to the content. It is not recommended to use a trusted device for an interface with a public network. ‘

Tip: You must select your private network interface as a *Trusted Device* in order for the cluster to operate.

Caution

The *private* interface used for your cluster, usually eth1, must be set up as a *Trusted Device* in order for the cluster software to work properly. It must be checked in the list titled *If you would like to allow all traffic from a device*.

Selecting a *public* network interface, usually eth0, as a *Trusted Device* may compromise the security of your cluster. In addition to the security considerations, selecting to allow DHCP on any interface other than the private interface can lead to conflicts with DHCP servers setup for those other networks.

Select services for which you want to allow possible connections.

Select additional ports. Empty default is fine for Scyld Beowulf to run properly. However, if you are planing to run services such as SSH or FTP between the public network and the master node, these services must be explicitly allowed.

Click **Next** to continue.

Additional Language Support

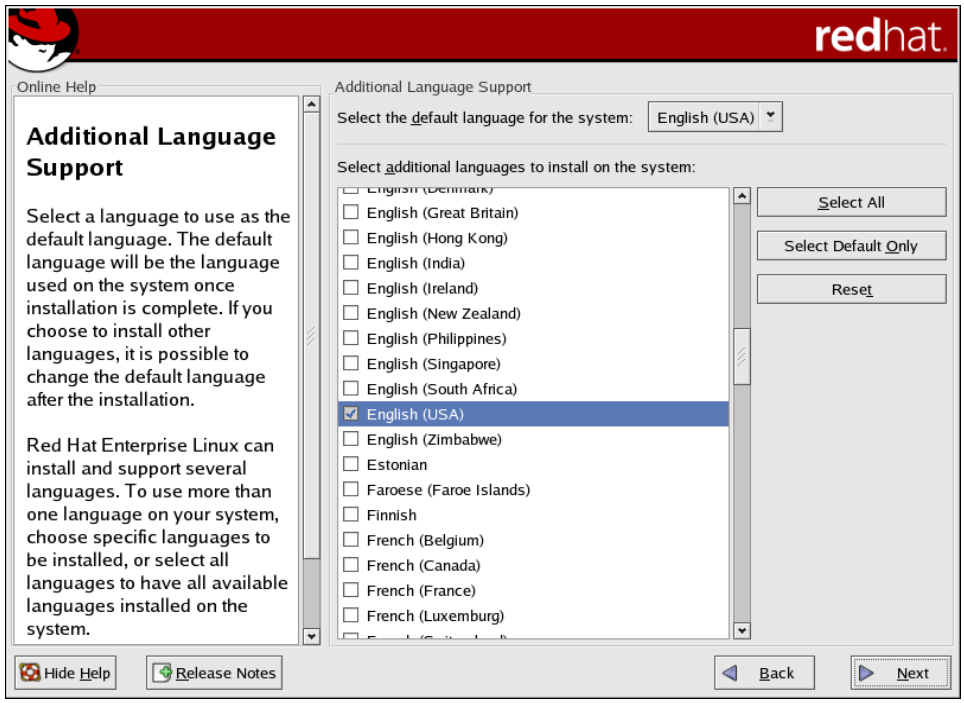

Figure 3-19. Language configuration

Some applications can display messages in a variety of languages. In this section, you need to select the default language used by your system, and the additional languages it can support. Language support affects not only the content, but also the format of messages, including date, monetary values, etc.

Click **Next** to continue.

Setting the Time Zone

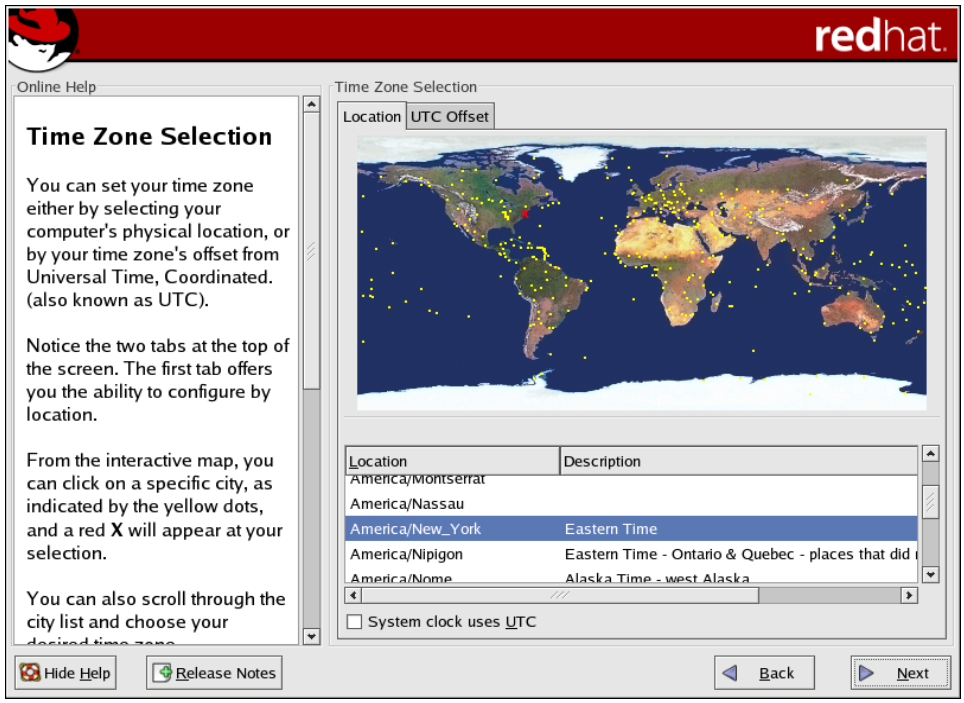

Figure 3-20. Time Zone configuration

Select the appropriate time zone for the location of your cluster. You have the option of setting this according to location or your time zone's offset from Universal Coordinated Time (UTC).

First, click the *Location* or the *UTC* tab.

- For the *Location* tab, select a city on the map or in the text listing below the map. You can change the map that appears by changing the geographical area listed in the **View** menu.
- For the *UTC* option, select the appropriate offset from those listed.

In either case, highlight the box labeled *System clock uses UTC* if this is true of your system clock. The time zone selection you make here should match your system hardware clock. Click **Next** to continue.

root Password Selection

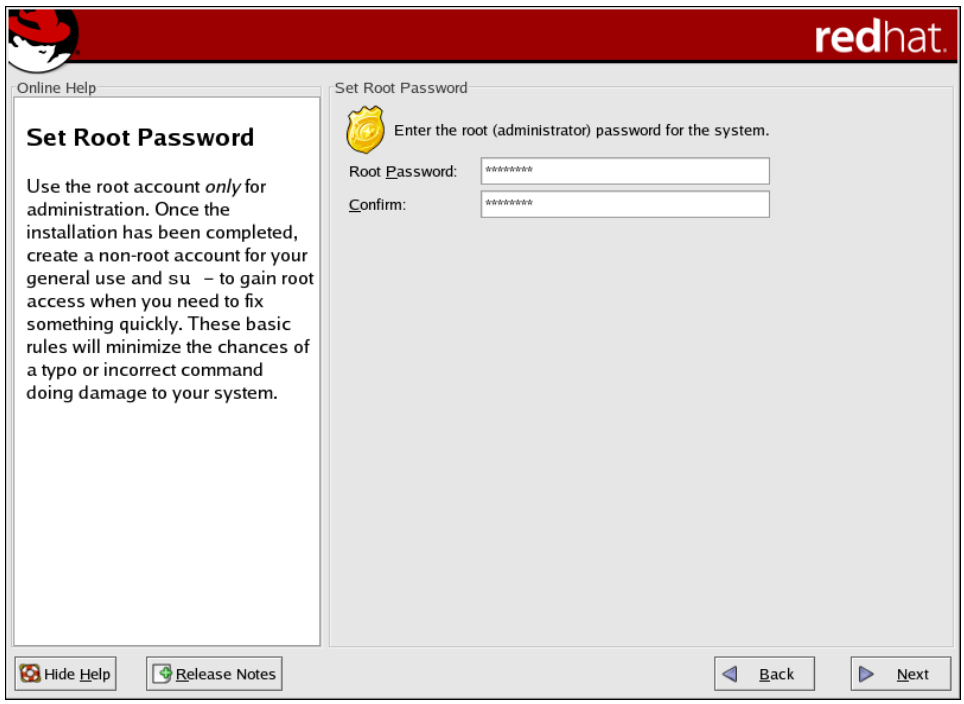

Figure 3-21. Setting the root password

You must choose a password for root the user. An alphanumeric password of at least 8 characters, with special characters is recommended.

Click **Next** to continue.

Selecting Package Groups

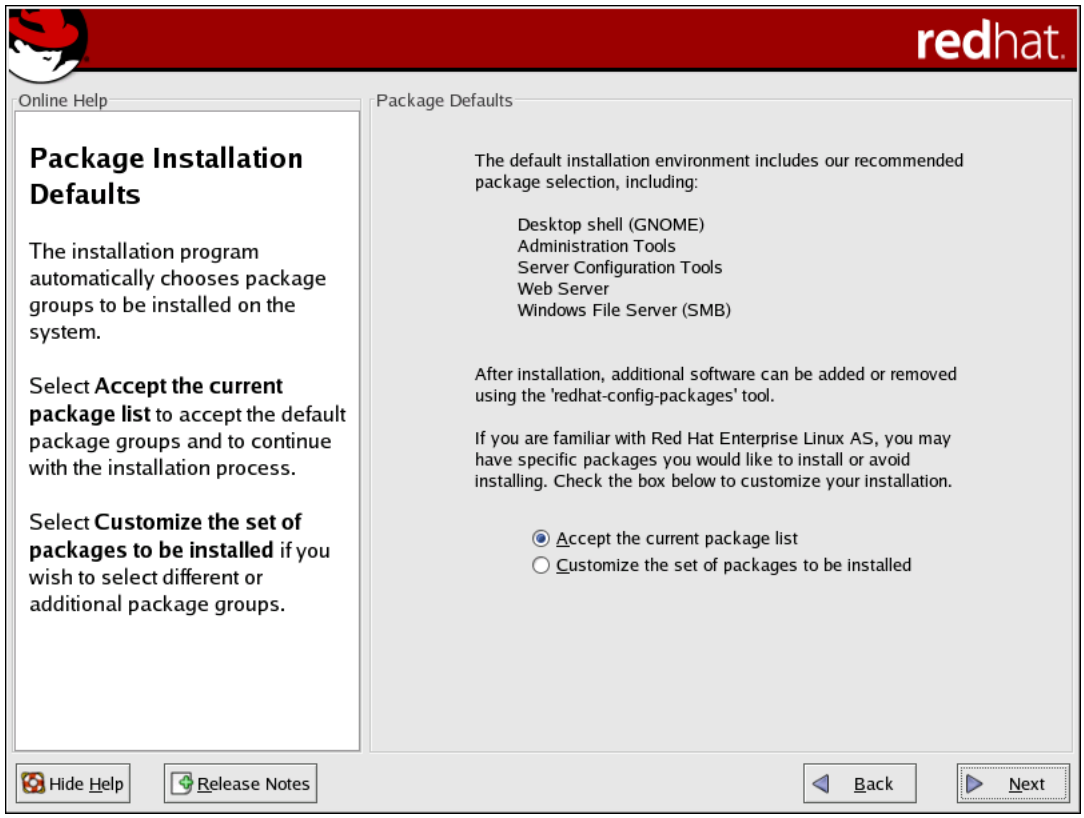

Figure 3-22. Selecting Package Groups

This section enables you to select the particular software packages that you wish to install. The default package selections should be good for most users. To examine or change package details, check *Customize the set of packages to be installed* before clicking *Next*, otherwise leave the default *Accept the current package list*.

> **Tip:** Ensure that X Software Development and GNOME Software Development are checked. These packages are required in order to run the cluster management tools.

If you chose to customize (or examine) packages, the next screen shows the details of all packages to be installed (see Figure 3-24).

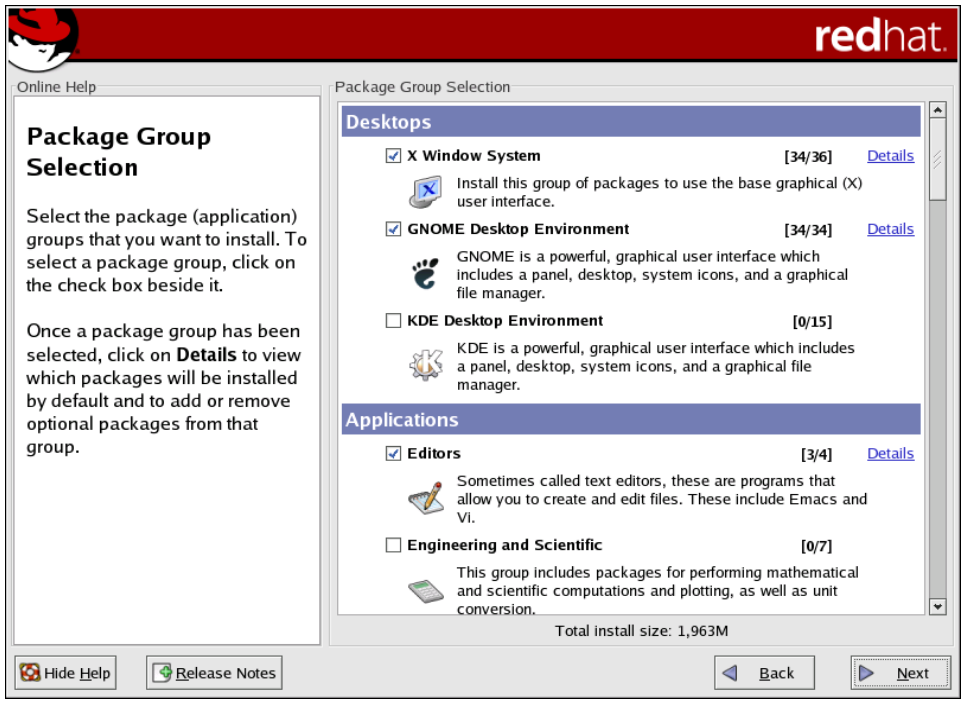

Figure 3-23. Package Details

If you want to know more about a particular package, select it and click on the *Details* link (see Figure 3-24).

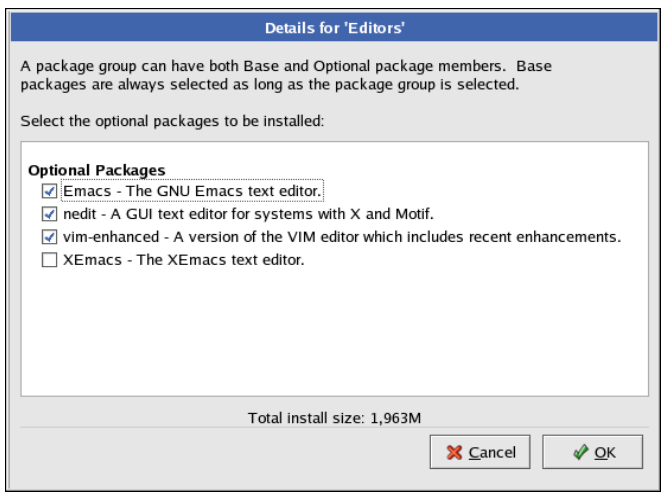

Figure 3-24. Package Details

Click **Next** to continue.

Unresolved Dependencies

You may encounter unresolved dependencies when installing the packages for the Scyld Beowulf. That is, some software packages depend upon others for the system to function properly. If any required packages are missing, you have the opportunity to rectify the situation. Simply select the *Install packages to satisfy dependencies* button in the ensuing dialog box.

About to Install

All of the information required for installation of Scyld Beowulf has been collected. To start the process of formatting your disks and installing the Scyld Beowulf software, click on the **Next** button.

Now sit back and relax while the installer does the work. This may be a lengthy process depending on your computer and the number of packages you chose to install. A progress indicator is displayed so that you may monitor the time remaining. The installation may stop at some point requiring another disk to proceed. When this happens, insert the next disc into the CD-ROM drive, and click **OK**.

Graphical Interface (X) Configuration

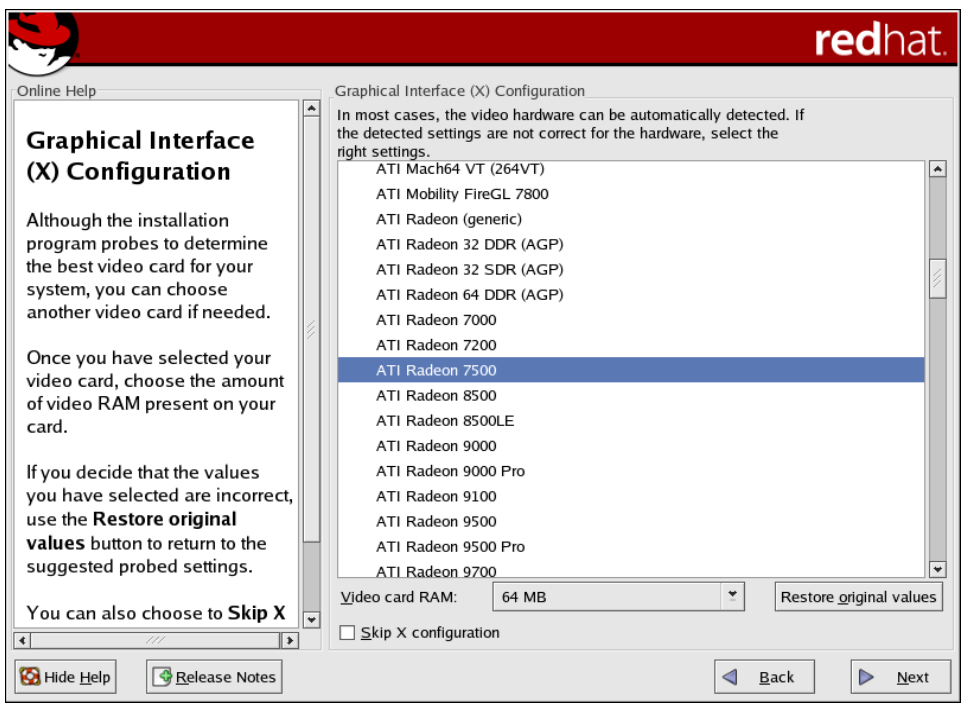

Figure 3-25. Video Configuration

The installer automatically probes your system to find the best match for your video card and memory. If it fails to detect them automatically, you must choose from the list of video cards.

If you do not see your card listed, it may be because XFree86 does not support it. However, if you are technically knowledgeable, you may choose *Unlisted Card* and attempt to configure it by matching the card's video chipset with one of the available X servers. You are also prompted for the amount of video memory installed on your video card—consult the video card documentation and enter the accurate amount of memory for XFree86 to work properly. If your video card has a video clock chip, choose *No Clockchip Setting* to let XFree86 automatically detect the proper clockchip, which works in most cases.

You can also choose to skip this step by checking the box labelled *Skip X configuration*. Click **Next** to proceed.

Monitor Configuration

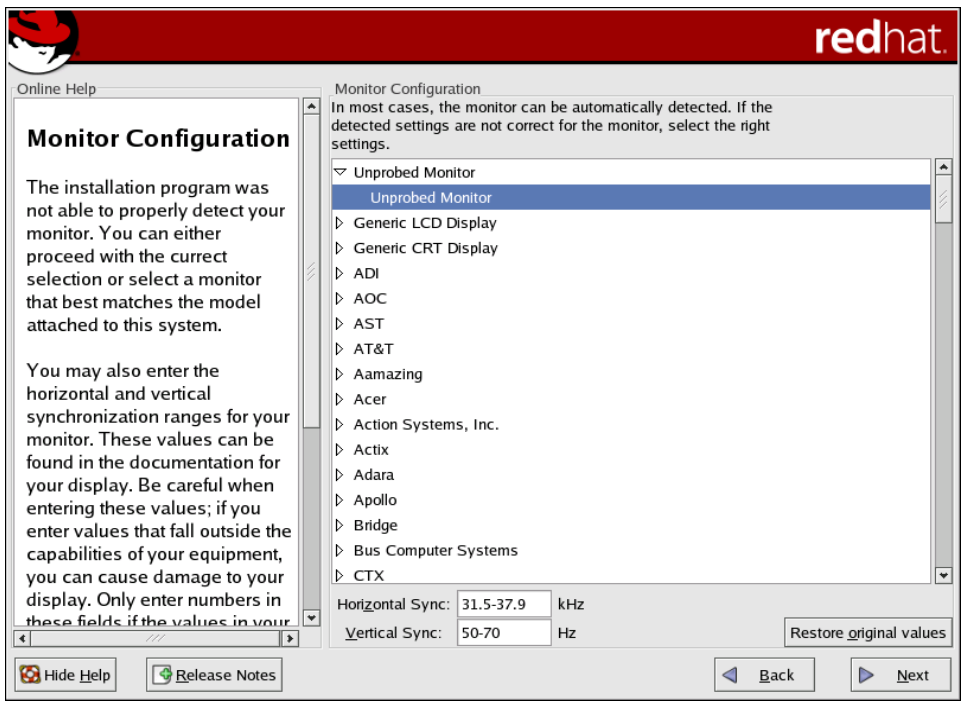

Figure 3-26. Monitor Configuration

The installer probes your monitor and normally identifies it correctly. If probing fails to correctly identify your monitor, or if you wish to change the settings, select the appropriate monitor type. Do not select a monitor that is merely "similar" to yours, unless you are positive that the selected monitor does not exceed the capabilities of your monitor. With older monitors there is a possibility of physical damage if you choose a more capable monitor type.

Clicking the **Restore original values** button reverts to the values discovered by probing the monitor. Click **Next** to proceed.

Customize Graphics Configuration

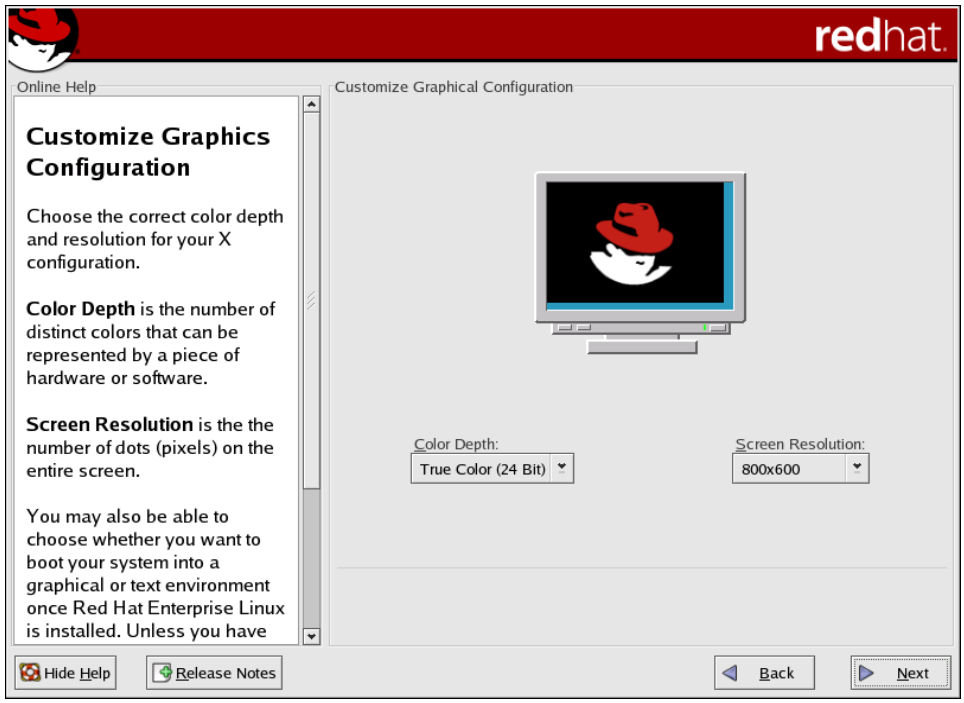

Figure 3-27. X Configuration

On this screen, select what color depth and resolution you would like for running X-Windows. Reasonable defaults are chosen by the installer based on your video card and monitor. from this screen you may change the Color Depth and Screen Resolution. Click **Next** to continue.

Reboot the System

Congratulations, you have successfully installed Scyld Beowulf on the head node of your Scyld Beowulf cluster. Remove the Scyld Beowulf CD-ROM from your drive and click **Next** to reboot your machine in order to finalize the installation and set up the cluster software.

Welcome

After the system reboots from its own hard disk, you are presented with a *Welcome* screen. The following steps enable you to finalize your system configuration and to install Scyld Beowulf software.

Beowulf Cluster

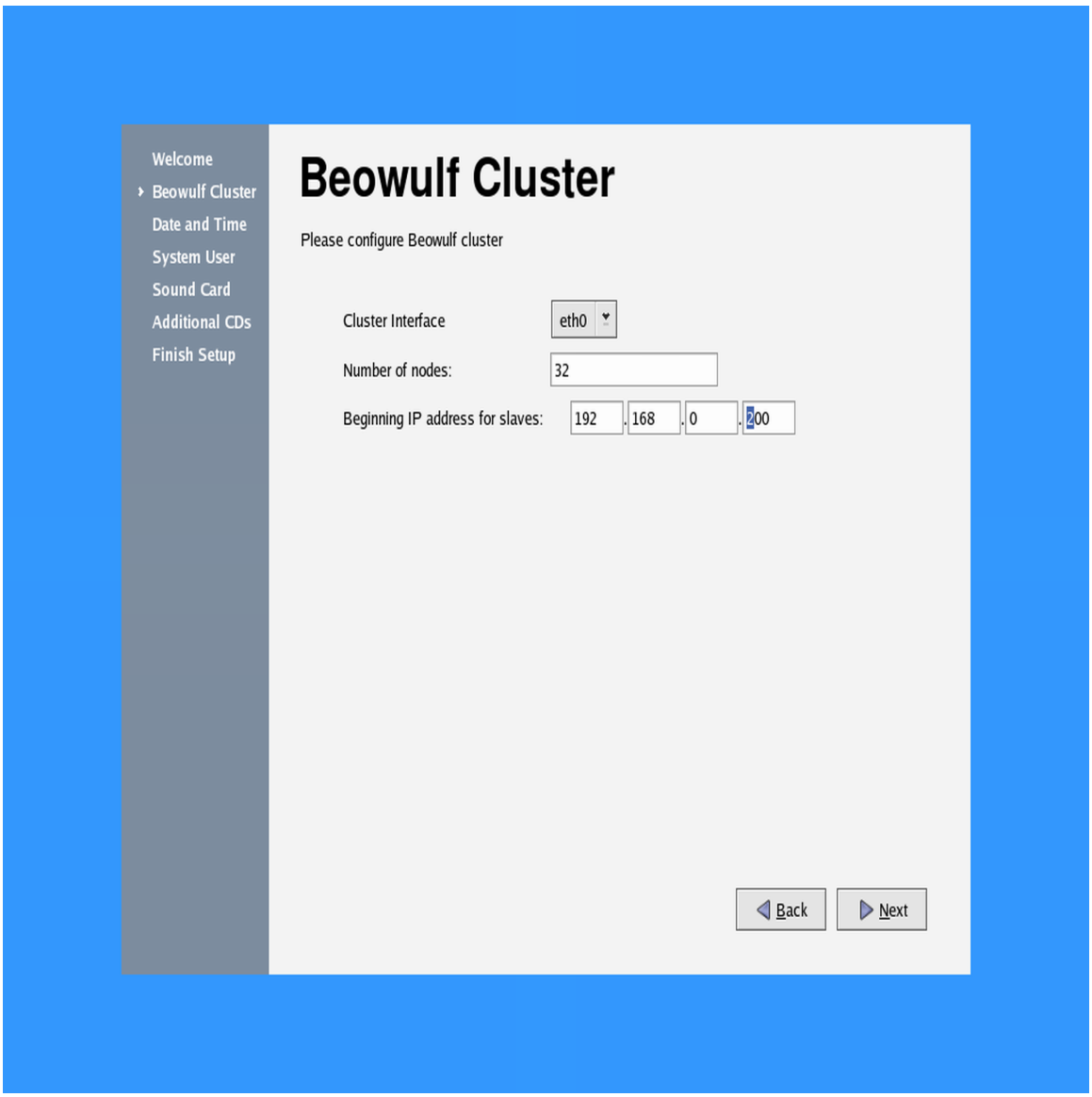

Figure 3-28. Beowulf Cluster

This page is very important. Under *Cluster Interface* choose the private ethernet interface which you set up in the Section called *Network Configuration*. Under *Number of nodes*, use the number of compute nodes you have in your cluster as a minimum---choosing a higher number here gives you room to easily expand your cluster in the future. For the beginning IP address, choose an address that corresponds to the manual IP configuration you chose for the private network (see Figure 3-17). For example, if you chose 192.168.104.1 as the address for eth1, you could use 192.168.104.10 as the beginning IP address for your cluster.

License Agreement

Verify the standard license agreement by checking *Yes, I agree to the License Agreement*. Click **Next** to continue.

Date and Time

Set the date and time for your system. If you are connected to the Internet and want your computer to synchronize its clock with a remote time server using the Network Time Protocol (NTP), check the box labelled *Enable Network Time Protocol* and choose or provide an NTP server. Click **Next** to continue.

System User

Although most cluster administration activities require root access, you may wish to set up at least one non-administrative system user. On this page you can provide a username, full name, and password. To enable network login facilities such as Kerberos or NIS, click the **Use Network Login** button and configure remote authentication. Click **Next** to continue.

Sound Card

If the system detects a sound card, you are given an opportunity to play a test sound to ensure that the card is configured correctly. Click **Play test sound** to play a test sound through all available channels. Click **Next** to continue.

Additional CDs

Nothing needs to be done for this page. Click **Next** to continue.

Finish Setup

Congratulations! The head node installation is complete, and the system has been configured. When you click **Next**, the Welcome series ends and you are presented with a standard Linux login screen.

Now that you have a functioning head node, it is time to boot and configure the compute nodes. Please go on to the next chapter.

Chapter 4. Installation of the Compute Nodes

In Scyld Beowulf clusters, no explicit installation is required on the compute nodes. The head node controls booting, provisioning, and operation of the compute nodes using the configuration solely from the head node.

Compute Node Boot Media

One of the innovations of Scyld Beowulf is the ability to boot compute nodes using a variety of boot mechanisms, yet always use a consistent run-time environment for applications, provisioned dynamically from the head node. This is accomplished without changing the administrative procedures or end-user interface. A second innovation is an architecture that provisions machines as operational compute nodes in as little as one second, even when they have not been previously configured.

PXE Network Boot

The easiest and recommended boot mechanism is PXE, the *Preboot eXecution Environment*. PXE is a network boot protocol that is nearly ubiquitous on current machines. Older machines may be inexpensively retrofitted by replacing the NIC or adding a boot ROM.

Using direct PXE boot has several advantages over using other boot media. The most significant is that the driver needed to support the specific NIC is included with the hardware. While this driver is not suitable for its run-time use, it eliminates the need to install and update network drivers in two places. A second advantage is speed: PXE boot is faster than using spinning disks, especially floppy disks. For these reasons we recommend using PXE boot whenever it is available.

BeoBoot Stage One

For older machines or network types that do not support PXE, Scyld developed the *Scyld BeoBoot* system. BeoBoot is a system to create boot media, such as a floppy boot disk or a bootable CD-ROM, that directs the machine to network boot using a Scyld-developed network boot protocol.

FIXME - running beoboot by hand is deprecated, use beosetup

In either case, compute nodes download their run-time kernel and operating system from the head node. Compute nodes are incorporated into the cluster using the Scyld cluster configuration tool, Beosetup, on the head node.

Beosetup

Scyld Beowulf includes the **BeoSetup** cluster configuration tool for simplifying the installation of the compute nodes. To begin configuring your compute nodes, you must log in as `root` user using the password you set in the Section called *root Password Selection* in Chapter 3. If you chose GNOME for your desktop (the default), you will see a graphical desktop

similar to Figure 4-1.

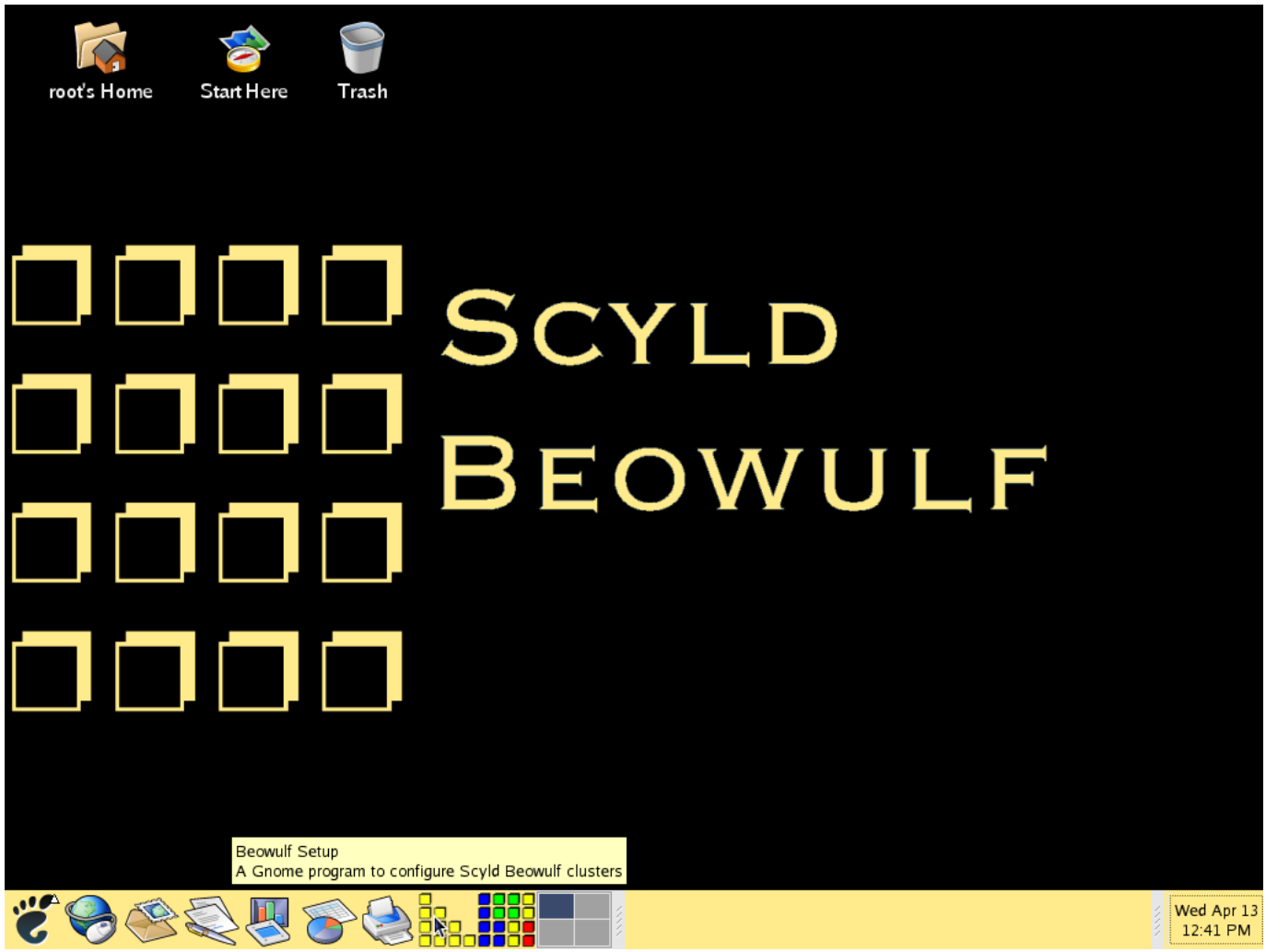

Figure 4-1. Gnome Desktop

Starting the BeoSetup Tool

BeoSetup is used to execute the installation of the compute nodes in the cluster. These sections provide descriptions of some of the basic functionality. For a detailed description of `BeoSetup`, see the Administrator's Guide.

To start `BeoSetup`, click the link to this tool on the tray in the GNOME desktop (see Figure 4-1). If it is not, you may start the `BeoSetup` GUI from a terminal window with the command,

```
bash$beosetup
```

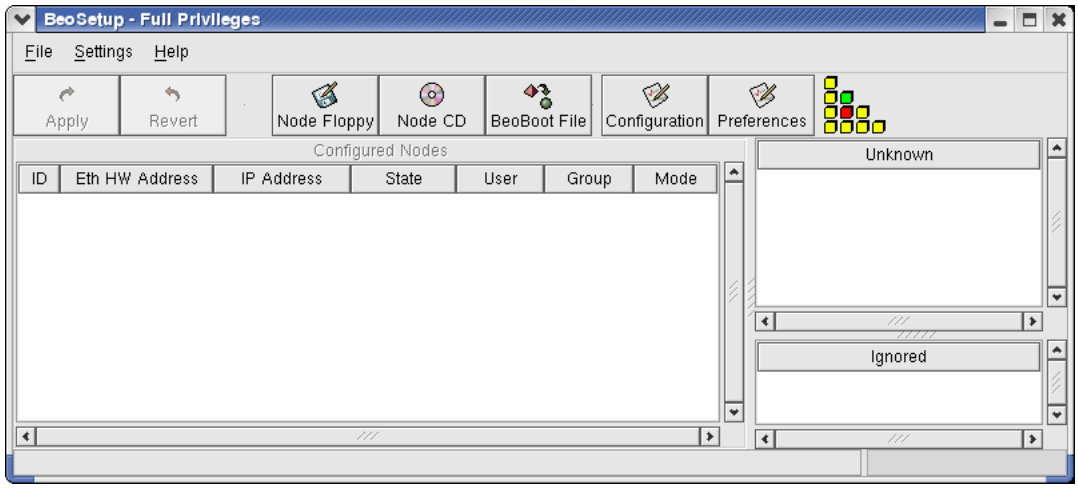

Figure 4-2. The BeoSetup Tool

The BeoSetup program (**BeoSetup**) is a graphical front-end for configuring and controlling a Scyld Beowulf cluster. It may be run by any user to monitor cluser node state, run commands, and read node logs, but the full functionality is only available to the 'root' user. When you start this tool as a user other than root, you are asked for the root password. If you don't supply it, functionality is limited. For this reason, Scyld recommends running Beosetup as root.

The BeoSetup program is a thin layer over the underlying cluster functionality, not the cluster interface itself. Every operation that it performs and every status that it reports is available from the command line, using scripts and with a library interface. Most of the configuration settings are written to the configuration file `/etc/beowulf/config`. Many of the actions, such as generating a boot floppy, report the command and options used to accomplish the task.

The first time you run `BeoSetup`, you see a dialog box asking if you want to *Auto Activate* nodes as they appear to the head node. Normally you answer **Yes**, and not have to take any action to add nodes other than powering them on. Answering **No** requires you to manually activate nodes as described below.

The Main Window

The main window contains three panes with Ethernet hardware (MAC) addresses, uniquely identifying machines. The *Configured Nodes* pane contains machines assigned a node number, along with the relevant state. The other two panes contain a list of MAC addresses. The *Ignored* pane lists machines that should never be added to this cluster, even though they have requested an IP address from the DHCP service, or a PXE image. The *Unknown* pane lists machines that have requested an IP address or PXE image, but have not yet been assigned to either of the other two lists.

Addresses may be moved between lists by dragging an address with the left (first) mouse button or by right (third button) clicking on the address with the mouse and choosing the appropriate pop-up menu item. Configured nodes may only be moved if they are in the Down state.

Note that, if you elect to automatically add new nodes to the cluster, or manually configure nodes to be added as described below, nodes do not appear in the *Unknown* or *Ignored* lists unless the maximum number of nodes is already connected to the master.

Apply and Revert buttons

After moving addresses between lists, the **Apply** button must be clicked for changes to take effect. Clicking on the **Apply** button saves the changes to the configuration file and signals the Beowulf daemons to re-read the configuration file.

Revert re-reads the existing Beowulf configuration file. This has the effect of undoing any undesired changes that have not yet been applied or synchronizing **beosetup** with any changes that have been made to the configuration file by an external editor.

Short Cuts

Next to the **Apply** and **Revert** buttons are short-cut buttons for generating node boot images, **Node Floppy** and **Node CD**, generating a new **BeoBoot File**, and changing **Configuration** settings or **Preferences**. These items are also accessible through the **File** Menu and **Settings** Menus.

Pop-up Menus

Each list item has a pop-up menu associated with it that may be accessed by right-clicking while pointing the cursor to the list item. Only those functions in the pop-up menu which may be applied to the highlighted line are clearly visible. Some operations are invalid at certain times, and are "grayed out" (not selectable).

Node Floppy button

If you plan to boot the compute nodes from floppy disk, you may use `BeoSetup` to create node floppy disks (recommended 1 disk per node) using the following procedure:

1. Click on the **Node Floppy** button in the main window.
2. Insert a floppy disk into the floppy drive of the head node.
3. Type in any *Kernel boot flags* required for your nodes. Normally, you need not change these.
4. Click OK to write the boot image to the floppy disk (/dev/fd0).

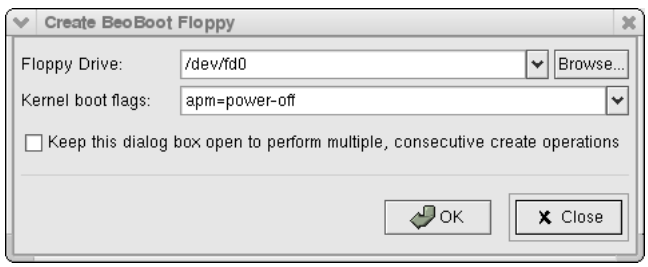

Figure 4-3. Creating BeoBoot Disks

Caution

Boot Floppy Diskettes are only usable for 32-bit operating systems. They can not be used to boot a 64-bit operating system.

Node CD button

If you plan to boot the compute nodes from CD-ROM, you may use `BeoSetup` to create node CDs (recommended 1 disc per node) using the following procedure:

1. Click on the **Node CD** button in the main window.
2. Insert an appropriate blank CD-R or CD-RW into the CD-RW drive of the head node.
3. Type in any *Kernel boot flags* required for your nodes. Normally, you need not change these.
4. Click OK to write the boot image to the disc (/dev/cdrom).

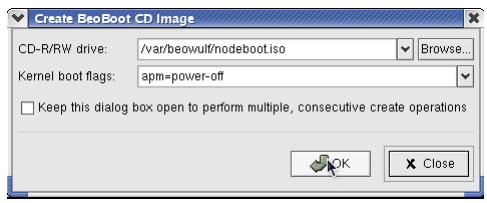

Figure 4-4. Creating BeoBoot CDs

Booting the Compute Nodes

Boot the compute nodes by powering them on, using the method selected at the beginning of this section (PXE or boot media). As the compute nodes boot, they are listed in BeoSetup by Ethernet Station (MAC) Addresses in the order they connect to the cluster. You may change this order, but it is easiest to power them up in order. The nodes appear in the *Configured Nodes* pane if you answered **Yes** to the *Auto Activate* dialog box (in), or in the *Unknown* pane Figure 4-5if you answered **No**.

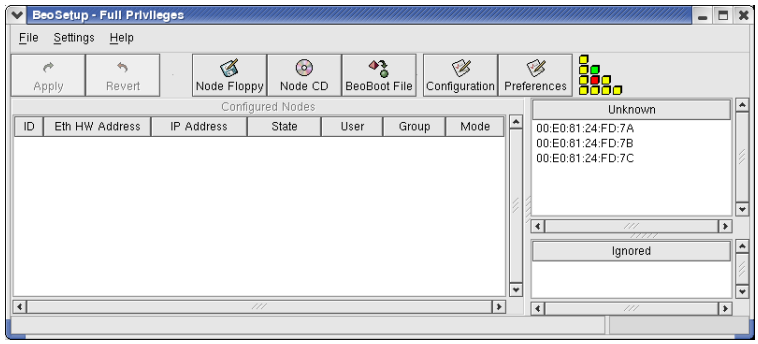

Figure 4-5. Listing of Available Compute Nodes in BeoSetup

Incorporating the Compute Nodes

Drag compute node MAC Addresses to the *Configured Nodes* pane; click **Apply**. This assigns the nodes to the cluster, using numbers `0` through `N-1`, where `N` is the maximum number of compute nodes configured.

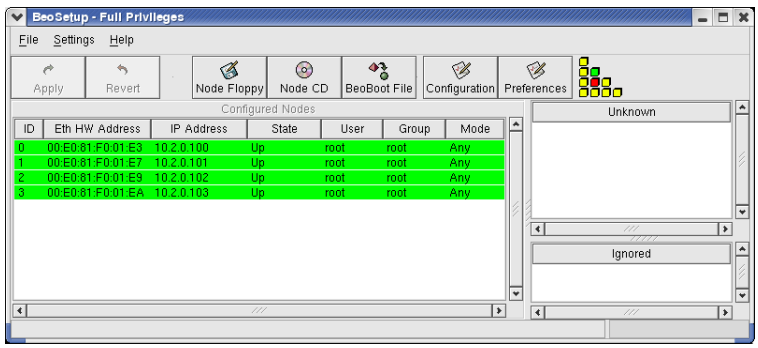

Figure 4-6. BeoSetup Display of Compute Nodes Assigned to Cluster

Optional Compute Node Disk Partitioning

If your compute nodes are diskless, you may skip this section.

Compute node hard disks may be remotely partitioned from the master machine. If the compute node hard disks have not been previously partitioned, you may use **beofdisk** to generate default partition tables for the compute node hard disks. For more details and options regarding the following steps, or to create custom partitioning, see Appendix A.

1. On the master machine, in the directory, `/usr/lib/beoboot/bin` , capture partition tables for the nodes.

   `bash$` **`beofdisk -q`**
   With the *`--q`* parameter, **beofdisk** queries all compute nodes. For the first drive found with a specific geometry (cylinders, heads, sectors), it reads the partition table and records it in a file (see Appendix A for more details). If the compute node disk has no partition table, a default table is created. Note: if the partition table on the disk is empty or invalid, a default partition table is not created. The command indicates whether it is creating a default partition table.

   If the compute node hard disk is unpartitioned, it is listed in the **BeoSetup** *Configured Nodes* pane in the *error* state. If the partition table is invalid or empty, you must create a default partition using the *`--d`* parameter:

   `bash$`**`beofdisk -d`**

2. For each drive of a specific geometry found on any compute node, write the appropriate partition table.

   `bash$` **`beofdisk -w`**
   This technique is useful, for example, when you boot a single compute node with a local hard disk that is already partitioned, and you want the same partitioning applied to all compute nodes. You would boot the prototypical compute node, capture its partition table, boot the remaining compute nodes and write that prototypical partition table to all nodes.

3. Optionally, write the floppy image to existing BeoBoot partitions on all nodes with disk named *`device`*, (where *`device`* is `/dev/hda` or `/dev/sda`).

   `bash$` **`beoboot-install -a`** ***`device`***
   This enables the compute nodes to be booted from their BeoBoot partitions, thus eliminating the need for a floppy disk to boot the nodes. Note that all compute nodes must have a boot partition. If compute nodes have different hard disk configurations, for example some have ATA disks (*`device`* = `/dev/hda`) and some have SCSI disks (*`device`* = `/dev/sda`), you must execute this command once for each device.

4. If needed, update the file `/etc/beowulf/fstab` on the head node to record the mapping of the partitions on the compute node disks to the filesystems.

Reboot the Compute Nodes

As the root user, reboot all of the compute nodes using these steps:

1. Select the node in the *Configured Nodes* pane of **BeoSetup**.
2. Right-click the mouse.
3. Select **Change Node State**, and select **Reboot**.

If you use **bpctl** to reboot the compute nodes, mounted partitions on the compute nodes may not be dismounted properly.

BeoBoot Floppy or CD-ROM

Compute nodes that do not implement PXE require a `BeoBoot` initial image to boot and operate as a member of the cluster. This BeoBoot image may be created using the `BeoSetup` tool. You may copy this image onto floppy disk(s), one for each compute node. For subsequent boots, you may choose to store this image in a "BeoBoot partition" on the compute node hard disk(s) if they exist.

- For the *floppy or CD boot*, see the Section called *Node Floppy button* or the Section called *Node CD button*.

- For the *hard disk boot*, the BeoBoot image must reside in the BeoBoot partition on the hard drive of each compute node that uses the hard drive boot. After the initial installation, each compute node may use whichever boot method is appropriate. Nodes need not use the same boot method.

Congratulations!

This completes the installation of the compute nodes and your entire Scyld Beowulf cluster.

Chapter 5. Cluster Verification Procedure

After you've finished configuring the master and compute nodes of your Scyld Beowulf cluster, the next step is to verify that the cluster is working properly. The following verification procedure is meant to identify common software and hardware configuration problems by running basic administrative and operational commands. When contacting your reseller for support with a new problem, typically the first question asked is if this verification procedure has been run, and what the results were.

bpstat

Entering the command **bpstat** at a shell prompt on the master node displays a table of status information for each node in your cluster. You do not need to be a privileged user to use this command. An example of using this command is shown below.

```
[root@cluster root]# bpstat
Node(s)     Status       Mode       User       Group
5-9         down         ---------- root       root
4           up           ---x--x--x any        any
0-3         up           ---x--x--x root       root
```

From the above table generated by **bpstat**, verify that you see `up` listed in the `Status` column for each node you've configured and have powered up. Status will be shown for each possible node in the cluster. The possible node count is based on the number of nodes specified by the `iprange` (see the Preference Settings in **beosetup**). Nodes that have not yet been configured are marked as `down`. If any node in the table contains `boot` in the `Status` column, this state is temporary while the node is booting. Wait 10-15 seconds and try again. If any node in the table contains `error` in the `Status` column, that node is operating but has experienced an initialization problem. As a first step, right click on the node entry in the **BeoSetup** display and select *View log* to check for error messages. Typical problems are failing network connections, unpartitioned hard disks or unavailable network file systems.

beostatus

Clicking on the **Beostatus** icon on the desktop system tray, or entering the command **beostatus** at a terminal windows on the master node displays a graphical user interface (GUI) program. You do not need to be a privileged user to use this command. The **beostatus** window is shown in Figure 5-2.

Figure 5-1. Beostatus Icon

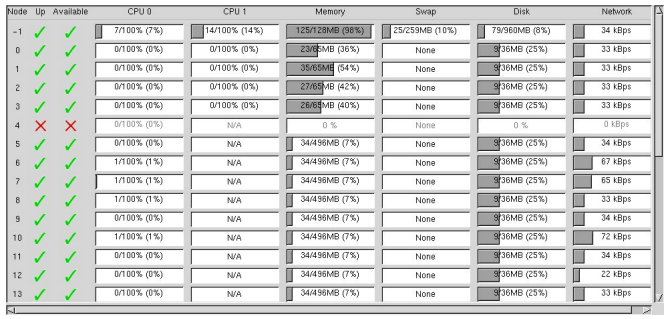

Figure 5-2. BeoStatus

The default mode of the Beostatus GUI is known as the "Classic" display. This mode displays specific state and resource usage information on a per-node table format.

Each row in the **beostatus** window corresponds to a different node in the cluster. The following list details the columns in the **beostatus** window:

Node

This is the node's assigned number in the cluster. Compute nodes are numbered starting with zero. Node -1, if shown, is the master node. The total number of node entries shown is set by the `iprange` or `nodes` keywords in the file `/etc/beowulf/config`, not the number of detected nodes. Inactive node entries display the last reported data in a faded or "grayed" row.

Up

This column gives a graphical representation of the node's status. A green checkmark is shown if the node is up and available. Otherwise, a red 'X' is shown.

State

This column prints the last known state of the node. The information in this column should agree with that reported by both **bpstat** and `BeoSetup`.

CPU 'x'

The next set of columns show the CPU loads for the node. At a minimum, there will be one column displaying the CPU load for the first processor in each node. Since it is possible to mix uni-processor machines with multi-processor machines in a Scyld Beowulf, the number of CPU load columns is equal to the maximum number of processors for a given node in your cluster. For those nodes that contain less than the maximum number of processors, their columns display N/A.

Memory

This column displays the current memory usage of the node.

Swap

This column displays the current swap space (virtual memory) usage of the node.

Disk

This column displays the current hard disk usage of the node. If the nodes are using a RAMdisk, they will show a maximum of 36MB.

Network

This column displays the current network bandwidth usage of the node. The total amount of bandwidth available is the sum of all network interfaces for the individual node.

Verify that the information shown in the **beostatus** window is correct. The configured nodes that are powered up (those with a green checkmark in the `Up` column) should show expected values in the subsequent usage columns. Assuming there are no active jobs on your cluster, the CPU and Network usage columns should be fairly close to zero. The memory usage columns (Memory, Swap and Disk) should be showing reasonable values.

bpsh

The **bpsh** command is the Beowulf shell command. It is analogous in functionality to both the **rsh** and **ssh** commands. It is used to execute commands on the nodes in your cluster from the master. For example, this command will execute on node number 3:

```
[root@cluster root]$ bpsh 3 ls -al /tmp
```

linpack

HPL is a portable version of the High Performance Linpack benchmark. Run it with all available nodes using the following shell script (wait up to a minute to see its complete output).

```
[root@cluster root]$ linpack
```

Caution

The linpack script runs a non-optimized version of the HPL benchmark, and is intended for verification purposes only. Do not use the results for performance characterization.

mpi-mandel

The **mpi-mandel** program is a visualizer for the Mandelbrot set. The following command is an example of how to run this program using 4 processors:

```
[root@cluster root]$ NP=4 mpi-mandel  --demo \
                     /usr/share/doc/mpi-mandel-1.0.20a/mandel.fav
```

Chapter 6. Troubleshooting a Scyld Beowulf Installation

Failing PXE Network Boot

If upon initially powering on the compute nodes (or subsequently), the nodes fail to join the cluster (or fail to appear in either the main or *Unknown* panels of `BeoSetup`, the problem may be that they are not finding the head node's DHCP server, or another DHCP server is answering and supplying an IP address.

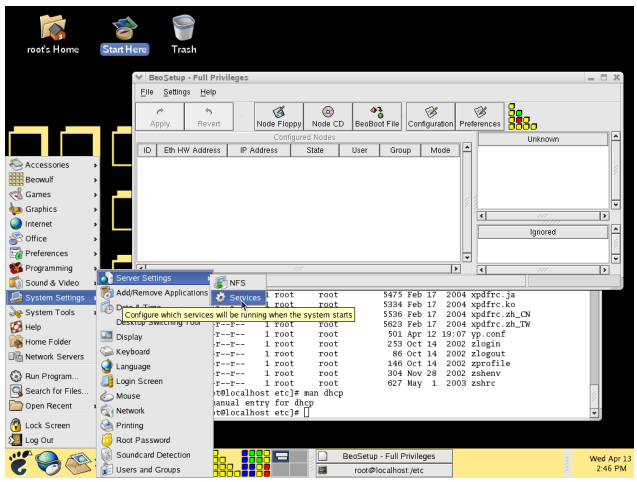

Figure 6-1. Starting Services Applet

Verify that the Beowulf services are started on the head node by opening the `Server Settings` applet (see Figure 6-1), and observing whether the beowulf service is started. If not, click **StartCluster** item under the **File** menu (see Figure 6-2). Then power-cycle a compute node to see if it now joins the cluster.

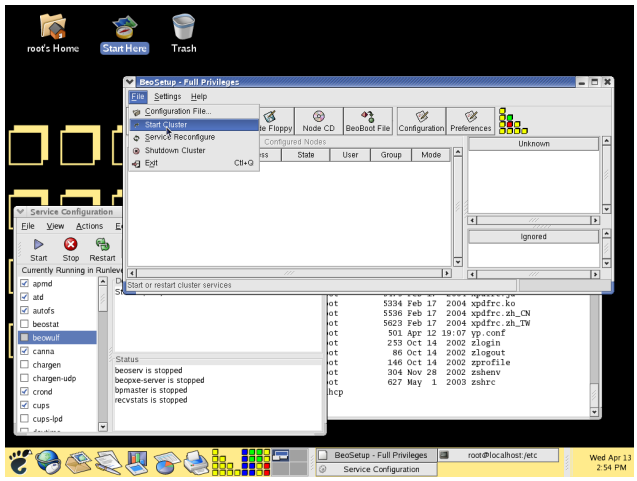

Figure 6-2. Starting Beowulf Services

If you are unable to start the cluster services (or Service Reconfigure), verify that the Master network interface is properly set using the **Configuration** button, Network Properties tab (see Figure 6-3), then start or reconfigure cluster services again. Verify that the Beowulf services have started (see Figure 6-4). Checking the boxes next to the beostat and beowulf services (in the `Service Configuration` applet) will insure these services start at boot time. Be sure to click **Save** before exiting the applet.

Try booting your compute nodes again.

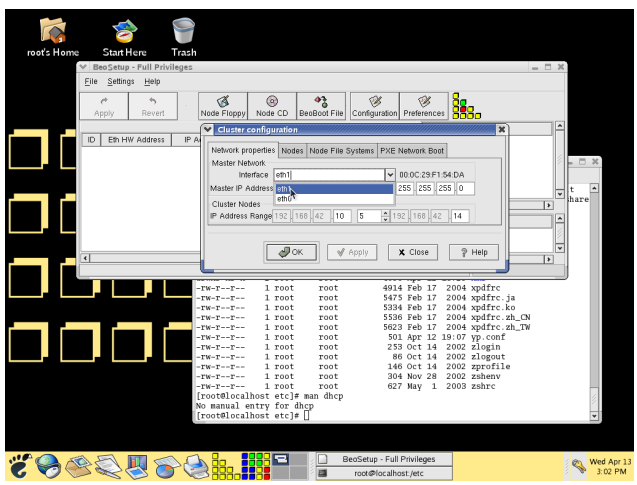

Figure 6-3. Checking Master network

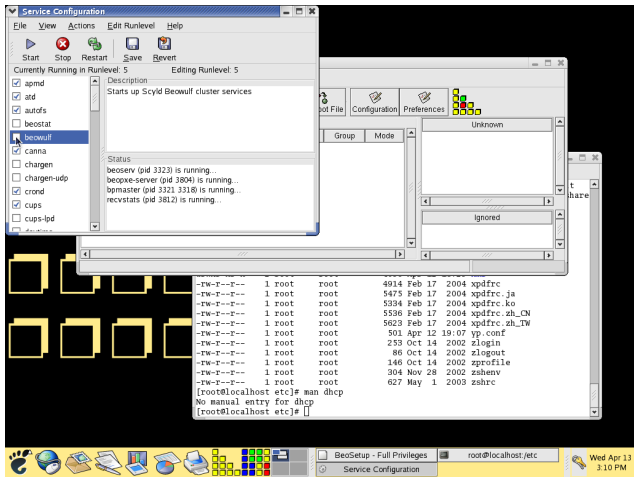

Figure 6-4. Beowulf Services Running

If the compute nodes fail to boot immediately after power-up, but successfully boot later, a common problem is the configuration of a managed switch.

Some Ethernet switches delay forwarding packets approximately one minute after link is established, attempting to verify that no network loop has been created ("spanning tree"). This delay is longer than the PXE boot timeout on some servers.

The solution is to disable the spanning tree check on the switch. The parameter is typically named "fast link enable". Note that BeoBoot Stage One was designed to attempt network boot for longer than the spanning tree timeout, thus the observed symptom is a delay booting the computing nodes rather than a failure.

Mixed Uniprocessor and SMP Cluster Nodes

One of the benefits of the Scyld Beowulf system architecture is that it eliminates the problem of unintentionally running different versions of a program over the cluster. The system eliminates version skew among compute nodes.

One requirement is that cluster nodes must run the same kernel version, typically with the same features and optimization. Uniprocessor machines can run the SMP kernel and SMP machines can run the uniprocessor kernel (although it will use only one processor). The best choice for a mixed cluster is to run the SMP kernel.

The kernel selection is handled at master installation time, based on the type of hardware detected and the response to questions. An SMP kernel is installed if the master is detected as an SMP, or if the *SMP* CPU selection box is checked during installation. There is a similar decision made based on the processor generation, for instance a kernel compiled to use Intel® Pentium® IV features will not work on a Transmeta(tm) Crusoe(tm) processor. If you installed a specialized kernel on a master that now needs to support slave nodes with a different set of features (e.g. uniprocessor master with SMP slave nodes), you must execute the following steps:

1. Mount the Scyld Beowulf CD-ROM on the head node.
2. Change to the directory `mount-point/Scyld/RPMS`, where mount-point is typically /mnt/cdrom.
3. Copy the kernel rpm, `kernel-smp-kernel-version` to the head node.
4. Install this kernel on the head node, by executing:

```
bash$ rpm -i kernel-smp-version
```

5. Reboot the head node and select the name of the SMP kernel from the boot loader prompt or GUI.
6. Make a new Phase 2 image by executing:

```
bash$ /usr/bin/beoboot -2 -n
```

Note that rebooting the head node automatically reboots the compute nodes, causing them to automatically use the updated kernel.

Mixed 32- and 64-bit cluster nodes

Mixing 32- and 64-bit nodes is not possible. The head node is migrating processes to the compute nodes. All nodes in the cluster must have the same CPU architecture. If you want to mix Opteron nodes and IA32 (Pentium or Xeon), you must boot the Opteron in 32-bit mode.

Device Driver Updates

Scyld Beowulf releases are tested on many different machine configurations, but it is not possible to provide device drivers for hardware that was unknown at the time of release.

Most unsupported hardware, or device-specific problem are resolved by updating to a newer device driver, but some devices may not yet be supported under Linux. Check with your hardware vendor.

The Scyld Beowulf architecture makes most driver updates simple. Drivers are installed and updated on the head node exactly as with a single machine installation. The new drivers are immediately available to compute nodes, although already-loaded drivers are not replaced.

There are two irregular device driver types that require special actions: disk drivers on the head node, and network drivers on the compute nodes. In both cases the drivers must be available to load additional drivers and programs, and are thus packaged in initial ramdisk images.

Device Driver Notes

Scyld Beowulf uses XFree86 version 4.3.0-1_Scyld for video card support. Any driver compatible with XFree86 will work with the system; check xfree86.org[1] for driver updates and video related trouble shooting information.

The LM sensor subsystem is an optional package that allows monitoring temperature, fan speed and other physical parameters. Before configuring this driver package, check that your chipset is supported. Installing on unsupported chipsets has been known to hang machines during the boot phase. If compute nodes hang during boot, the last line in the node boot log, `/var/log/beowulf/node.x` usually indicates the problem.

Finding Further Information

If you encounter a problem installing your Scyld Beowulf cluster, and you find this guide cannot help you, check the following sources for pertinent information:

- See *Installation Guide, Graphical Install of Front-End Node* on the head node or on the Scyld disc in the installation kit for detailed installation instructions.

- The *Administrator's Guide* is available on the head node or the Scyld disc in the installation kit for a description of more advanced administration and setup options.
- The *Reference Guide* on the head node or on the Scyld disc in the installation kit for a complete technical reference to the Scyld Beowulf software.
- Run the `BeoSetup` application for access to detailed error info regarding the status of booting the compute nodes.

Please visit the Scyld MasterLink™ website at http://www.scyld.com/support.html for the most up to date product documentation and other helpful information about your Scyld Beowulf software.

Notes

1. http://www.xfree86.org
2. http://www.scyld.com/support.html

Appendix A. Compute Node Disk Partitioning

Architectural Overview

The Scyld Beowulf system uses a "diskless administration" model for compute nodes. This means that the compute nodes boot and operate without the need for mounting any file system, either on a local disk or a network file system. By using this approach, the cluster system does not depend on the storage details or potential misconfiguration of the compute nodes, instead putting all configuration information and initialization control on the master.

This does not mean that the cluster cannot or does not use local disk storage or network file systems. Instead it allows the storage to be tailored to the needs of the application rather than the underlying cluster system.

The first operational issue after installing a cluster is initializing and using compute node storage. While the concept and process is similar to configuring the master machine, the "diskless administration" model makes it much easier to change the storage layout on the compute nodes.

Operational Overview

Compute node hard disks are used for three primary purposes:

Swap Space:

expanding the Virtual Memory of the local machine

Application file storage:

providing scratch space and persistent storage for application output

System caching:

increasing the size and count of executables and libraries cached by the local node

In addition, local disk may be used to hold a BeoBoot Stage One image (for use when PXE booting is not available) or a cluster file system (for use when the node acts as a file server to other nodes).

To make this possible, Scyld provides programs to create disk partitions, a system to automatically create and check file systems on those partitions, and a mechanism to mount file systems.

Partitioning Disks

Deciding on a partitioning schema for compute node disks is no easier than with the head node, but at least it may be more easily changed.

Compute node hard disks may be remotely partitioned using the **beofdisk** command. The **beofdisk** command automates the partitioning process, allowing all compute node disks with a matching hard drive geometry (cylinders, heads, sectors) to be partitioned simultaneously.

The **beofdisk** command may also be used to read an existing partition table on a compute node hard disk, as long as that disk is properly positioned in the cluster. The command captures the partition table of the first hard disk of its type and geometry (cylinder, heads, sectors) in each position on a compute node's controller (e.g., sda or hdb). The script sequentially queries the compute nodes numbered, `0` through `N-1`, where `N` is the number of nodes currently in the cluster.

The default partition table allocates three partitions: a BeoBoot partition equal to 2 MB, a swap partition equal to two times the node's physical memory, and a single root partition equal to the remainder of the disk. The partition table for each disk

geometry is stored in the directory `/etc/beowulf/fdisk` on the master machine with the filename specified in nomenclature which reflects the disk type, position and geometry (example filenames: `hda:2495:255:63`, `hdb:3322:255:63`, `sda:2495:255:63`).

While it is not possible to predict every configuration that might be desired, the typical procedure to partition node disks is as follows:

1. Capture partition tables for the nodes. Note, if the nodes' drives have no partition tables, this command creates a default partition set (and reports this activity to the console). If there is an empty partiton table, or an invalid partition table, it is captured and recorded as described, but no default partition set is created. See the Section called *Default Partitioning* to set up default partitions.

   ```
   bash$ beofdisk -q
   ```

2. Write the appropriate partition table to every drive on every node.

   ```
   bash$ beofdisk -w
   ```

3. Optionally, write the BeoBoot image to all existing BeoBoot partitions on nodes with hard disk named *`device`* (where *`device`* = `/dev/hda` or `/dev/sda`).

   ```
   bash$ beoboot-install -a device
   ```

4. Reboot all compute nodes using **BetSetup** to make the partitioning effective.

Default Partitioning

To apply the recommended default partitioning to all of your disks follow all of these steps:

1. Generate default partition maps to `/etc/beowulf/fdisk`:

   ```
   bash$ beofdisk -d
   ```

2. Write these out to the nodes:

   ```
   bash$ beofdisk -w
   ```

3. It is recommended that you let the compute nodes PXE boot, rather than writing a boot image to the local disk. If you choose to write a bootable image to the compute node disks, first install the first-stage boot kernel image:

   ```
   bash$ beoboot-install -a device
   ```
   Then enable the head node to provide a stage-2 kernel and complete booting the compute nodes:

   ```
   bash$ beoboot-install -2 -a device
   ```

You must reboot the compute nodes before the new partitions are usable. Rebooting should be done using **BeoSetup**.

Mapping Compute Node Partitions

If your compute node hard disks are already partitioned, edit the file `/etc/beowulf/fstab` on the head node to record the mapping of the partitions on your compute node disks to your filesystems. This file contains example lines (commented out) showing mapping of file systems to drives (read the comments in the `fstab` file for guidance. First *`query`* the disks on the compute nodes to determine how that are partitioned.

```
bash$ beofdisk --q
```

This creates a partition file in `/etc/beowulf/fdisk` with a name similar to `sda:512:128:32`, containing lines similar to:

```
[root@cluster root]# cat sda:512:128:32
/dev/sda1  :  start=    32,  size=  8160,   id=89,  bootable
/dev/sda2  :  start=    8192,   size=    1048576,  Id=82
/dev/sda3  :  start=    1056768,    size=    1040384,  Id=83
/dev/sda4  :  start=    0, size=  0,  Id=0
```

Read the comments in `/etc/beowulf/fstab`. Add the lines to the file to use the devices named in the `sda` file:

```
# This is the default setup from beodfisk
#/dev/hda2        swap     swap   defaults     0 0
#/dev/hda3        /        ext2   defaults     0 0
/dev/sda1         /boot    ext23   defaults     0 0
/dev/sda2         swap     swap   defaults     0 0
/dev/sda3         /        ext3   defaults     0 0
```

After saving `fstab`, you must reboot the compute nodes for the changes to take affect. You may also have to set the BIOS to boot from the proper hard disk.

Generalized, User-Specified Partitions

To create a unique partition table for each disk type/position/geometry triplet, remotely run the **fdisk** command on each compute node where the disk resides:

```
bash$  bpsh n fdiskdevice
```

where *n* is the node number or the first compute node with the drive geometry you want to partition, and `device` is the device you wish to partition (e.g., `/dev/sda`, `/dev/hdb`). One you have created the partition table and written it to the disk using **fdisk**, capture it and write it to all disks with the same geometry using **beofdisk -q**.

```
bash$ beofdisk -w
```

Reboot the compute nodes using **BeoSetup** before the partitioning is effective.

You must then map filesystems to partitions as described in the Section called *Mapping Compute Node Partitions*.

While it is recommended to PXE boot compute nodes, you can optionally write a boot image to the compute node disks after the compute nodes reboot:

```
bash$ beoboot-install -a device
```

Unique Partitions

To generate a unique partition for a particular disk, first partition your disks using one of the above scenarios. Then, from the head node, remotely run **fdisk** on the appropriate compute node to re-create a unique partition table using:

```
bash$  bpsh n fdisk device
```

where *n* is the compute node number for which you wish to create a unique partition table and `device` is the device you wish to partition (e.g., `/dev/sda`). You will then need to map file systems to partitions as described in the Section called *Mapping Compute Node Partitions*.