Fedora 11

Installation Guide

Installing Fedora 11 on x86, AMD64, and Intel® 64 architectures

fedora

Fedora Documentation Project

Fedora 11 Installation Guide
Installing Fedora 11 on x86, AMD64, and Intel® 64 architectures
Edition 1.0

Author   Fedora Documentation Project   *fedora-docs-list@redhat.com*
Copyright © 2009 Red Hat, Inc. and others

Provides documentation for the installation process.

Preface

1. Document Conventions

This manual uses several conventions to highlight certain words and phrases and draw attention to specific pieces of information.

In PDF and paper editions, this manual uses typefaces drawn from the *Liberation Fonts*[1] set. The Liberation Fonts set is also used in HTML editions if the set is installed on your system. If not, alternative but equivalent typefaces are displayed. Note: Red Hat Enterprise Linux 5 and later includes the Liberation Fonts set by default.

1.1. Typographic Conventions

Four typographic conventions are used to call attention to specific words and phrases. These conventions, and the circumstances they apply to, are as follows.

`Mono-spaced Bold`

Used to highlight system input, including shell commands, file names and paths. Also used to highlight key caps and key-combinations. For example:

> To see the contents of the file `my_next_bestselling_novel` in your current working directory, enter the `cat my_next_bestselling_novel` command at the shell prompt and press `Enter` to execute the command.

The above includes a file name, a shell command and a key cap, all presented in Mono-spaced Bold and all distinguishable thanks to context.

Key-combinations can be distinguished from key caps by the hyphen connecting each part of a key-combination. For example:

> Press `Enter` to execute the command.
>
> Press `Ctrl+Alt+F1` to switch to the first virtual terminal. Press `Ctrl+Alt+F7` to return to your X-Windows session.

The first sentence highlights the particular key cap to press. The second highlights two sets of three key caps, each set pressed simultaneously.

If source code is discussed, class names, methods, functions, variable names and returned values mentioned within a paragraph will be presented as above, in `Mono-spaced Bold`. For example:

> File-related classes include `filesystem` for file systems, `file` for files, and `dir` for directories. Each class has its own associated set of permissions.

Proportional Bold

This denotes words or phrases encountered on a system, including application names; dialogue box text; labelled buttons; check-box and radio button labels; menu titles and sub-menu titles. For example:

---

[1] https://fedorahosted.org/liberation-fonts/

Choose **System > Preferences > Mouse** from the main menu bar to launch **Mouse Preferences**. In the **Buttons** tab, click the **Left-handed mouse** check box and click **Close** to switch the primary mouse button from the left to the right (making the mouse suitable for use in the left hand).

To insert a special character into a **gedit** file, choose **Applications > Accessories > Character Map** from the main menu bar. Next, choose **Search > Find…** from the **Character Map** menu bar, type the name of the character in the **Search** field and click **Next**. The character you sought will be highlighted in the **Character Table**. Double-click this highlighted character to place it in the **Text to copy** field and then click the **Copy** button. Now switch back to your document and choose **Edit > Paste** from the **gedit** menu bar.

The above text includes application names; system-wide menu names and items; application-specific menu names; and buttons and text found within a GUI interface, all presented in Proportional Bold and all distinguishable by context.

Note the **>** shorthand used to indicate traversal through a menu and its sub-menus. This is to avoid the difficult-to-follow 'Select **Mouse** from the **Preferences** sub-menu in the **System** menu of the main menu bar' approach.

`Mono-spaced Bold Italic` or `Proportional Bold Italic`

Whether Mono-spaced Bold or Proportional Bold, the addition of Italics indicates replaceable or variable text. Italics denotes text you do not input literally or displayed text that changes depending on circumstance. For example:

To connect to a remote machine using ssh, type `ssh username@domain.name` at a shell prompt. If the remote machine is `example.com` and your username on that machine is john, type `ssh john@example.com`.

The `mount -o remount file-system` command remounts the named file system. For example, to remount the `/home` file system, the command is `mount -o remount /home`.

To see the version of a currently installed package, use the `rpm -q package` command. It will return a result as follows: `package-version-release`.

Note the words in bold italics above — username, domain.name, file-system, package, version and release. Each word is a placeholder, either for text you enter when issuing a command or for text displayed by the system.

Aside from standard usage for presenting the title of a work, italics denotes the first use of a new and important term. For example:

When the Apache HTTP Server accepts requests, it dispatches child processes or threads to handle them. This group of child processes or threads is known as a *server-pool*. Under Apache HTTP Server 2.0, the responsibility for creating and maintaining these server-pools has been abstracted to a group of modules called *Multi-Processing Modules* (*MPMs*). Unlike other modules, only one module from the MPM group can be loaded by the Apache HTTP Server.

1.2. Pull-quote Conventions

Two, commonly multi-line, data types are set off visually from the surrounding text.

Output sent to a terminal is set in `Mono-spaced Roman` and presented thus:

```
books       Desktop   documentation  drafts  mss    photos   stuff  svn
books_tests Desktop1  downloads      images  notes  scripts  svgs
```

Source-code listings are also set in `Mono-spaced Roman` but are presented and highlighted as follows:

```
package org.jboss.book.jca.ex1;

import javax.naming.InitialContext;

public class ExClient
{
   public static void main(String args[])
       throws Exception
   {
      InitialContext iniCtx = new InitialContext();
      Object         ref    = iniCtx.lookup("EchoBean");
      EchoHome       home   = (EchoHome) ref;
      Echo           echo   = home.create();

      System.out.println("Created Echo");

      System.out.println("Echo.echo('Hello') = " + echo.echo("Hello"));
   }

}
```

1.3. Notes and Warnings

Finally, we use three visual styles to draw attention to information that might otherwise be overlooked.

Note

A note is a tip or shortcut or alternative approach to the task at hand. Ignoring a note should have no negative consequences, but you might miss out on a trick that makes your life easier.

Important

Important boxes detail things that are easily missed: configuration changes that only apply to the current session, or services that need restarting before an update will apply. Ignoring Important boxes won't cause data loss but may cause irritation and frustration.

> **⚠ Warning**
>
> A Warning should not be ignored. Ignoring warnings will most likely cause data loss.

2. We Need Feedback!

If you find a typographical error in this manual, or if you have thought of a way to make this manual better, we would love to hear from you! Please submit a report in Bugzilla: *http://bugzilla.redhat.com/bugzilla/* against the product **Fedora.**

When submitting a bug report, be sure to mention the manual's identifier: *Installation_Guide*

If you have a suggestion for improving the documentation, try to be as specific as possible when describing it. If you have found an error, please include the section number and some of the surrounding text so we can find it easily.

Introduction

This guide covers installation of Fedora, a Linux distribution built on free and open source software. This manual helps you install Fedora on desktops, laptops, and servers. The installation system is easy to use even if you lack previous knowledge of Linux or computer networks. If you select default options, Fedora provides a complete desktop operating system, including productivity applications, Internet utilities, and desktop tools.

This document does not detail all of the features of the installation system.

1. Background

1.1. About Fedora

To find out more about Fedora, refer to *http://fedoraproject.org/*. To read other documentation on Fedora related topics, refer to *http://docs.fedoraproject.org/*.

1.2. Getting Additional Help

For information on additional help resources for Fedora, visit *http://fedoraproject.org/wiki/Communicate*.

2. About This Document

2.1. Goals

This guide helps a reader:

1. Understand how to locate the Fedora distribution online

2. Create configuration data that allows a computer to boot Fedora

3. Understand and interact with the Fedora installation program

4. Complete basic post-installation configuration of a Fedora system

> **Other Sources of Documentation**
> This guide does not cover use of Fedora. To learn how to use an installed Fedora system, refer to *http://docs.fedoraproject.org/* for other documentation.

2.2. Audience

This guide is intended for new and intermediate Fedora users. Advanced Fedora users with questions about detailed operation of expert installation features should consult the Anaconda development mailing list at *http://www.redhat.com/archives/anaconda-devel-list/*.

Quick Start for Experts

This section offers a very brief overview of installation tasks for experienced readers who are eager to get started. Note that many explanatory notes and helpful hints appear in the following chapters of this guide. If an issue arises during the installation process, consult the appropriate chapters in the full guide for help.

> **Experts Only**
>
> This section is intended only for experts. Other readers may not be familiar with some of the terms in this section, and should move on to *Chapter 2, New Users* instead.

1.1. Overview

The installation procedure is fairly simple, and consists of only a few steps:

1. Download files to make media or another bootable configuration.

2. Prepare system for installation.

3. Boot the computer and run the installation process.

4. Reboot and perform post-installation configuration.

1.2. Download Files

Do any one of the following:

> **Verify your downloads**
>
> Downloads may fail for any number of reasons. Always verify the sha1sum of the downloaded files.

1. Download the ISO image for a Live image. Create CD media from the ISO file using your preferred application. You may also use the *livecd-tools* package to write the image to other bootable media such as a USB flash disk. To install the distribution to your hard disk, use the shortcut on the desktop after you log in.

2. Download the ISO images for the full distribution on CD or DVD. Create CD or DVD media from the ISO files using your preferred application, or put the images on a Windows FAT32 or Linux ext2/ext3 partition.

3. Download the `boot.iso` image for a minimal boot CD or USB flash drive. Write the image to the approriate physical media to create bootable media. The boot media contains no packages but must be pointed at a hard disk or online repository to complete the installation.

4. Download the `netinst.iso` image for a reduced-size boot CD. Write the image to the appropriate physical media to create bootable media.

5. Download the `vmlinuz` kernel file and the `initrd.img` ramdisk image from the distribution's `isolinux/` directory. Configure your operating system to boot the kernel and load the ramdisk

image. For further information on installation without media, refer to *Chapter 10, Installing Without Media*.

For information on setting up a network boot server from which you can install Fedora, refer to *Chapter 11, Setting Up an Installation Server*.

1.3. Prepare for Installation

Back up any user data you need to preserve.

> **Resizing Partitions**
>
> The installation program provides functions for resizing ext2, ext3, ext4, and NTFS formatted partitions. Refer to *Section 7.21, "Partitioning Your System"* for more information.

1.4. Install Fedora

Boot from the desired media, with any options appropriate for your hardware and installation mode. Refer to *Chapter 9, Boot Options* for more information about boot options. If you boot from the Live CD, select the "Install to Hard Disk" option from the desktop to run the installation program. If you boot from minimal media or a downloaded kernel, select a network or hard disk resource from which to install.

Proceed through all the steps of the installation program. The installation program does not change your system until you make a final confirmation to proceed. When installation is finished, reboot your system.

1.5. Perform Post-installation Steps

After the system reboots, it displays additional configuration options. Make appropriate changes to your system and proceed to the login prompt.

New Users

This chapter explains how to get the files you need to install and run Fedora on your computer. Concepts in this chapter may be new, especially if this is your first free and open source operating system. If you have any trouble with this chapter, find help by visiting the Fedora Forums at *http://www.fedoraforum.org/*.

> **Download Links**
>
> To follow a Web-based guide to downloading, visit *http://get.fedoraproject.org/*. For guidance on which architecture to download, refer to *Section 2.2, "Which Architecture Is My Computer?"*.

2.1. How Do I Download Installation Files?

The Fedora Project distributes Fedora in many ways, mostly free of cost and downloaded over the Internet. The most common distribution method is CD and DVD media. There are several types of CD and DVD media available, including:

- A full set of the software on DVD media

- Live images you can use to try Fedora, and then install to your system if you so choose

- Reduced-size bootable CD and USB flash disk images you can use to install over an Internet connection

- Source code on DVD media

Most users want the Fedora Live image or the full set of installable software on DVD or CDs. The reduced bootable images are suitable for use with a fast Internet connection and install Fedora on one computer. Source code discs are not used for installing Fedora, but are resources for experienced users and software developers.

> **Downloading media**
>
> Users with a broadband Internet connection can download *ISO images* of CD and DVD media or images of USB flash disks. An ISO image is a copy of an entire disc in a format suitable for writing directly to a CD or DVD. A USB flash disk image is a copy of an entire disk in a format suitable for writing directly to a USB flash disk.
>
> For more information on burning CDs and DVDs, refer to *Section 2.4, "How Do I Make Fedora Media?"*.

Fedora software is available for download at no cost in a variety of ways.

2.1.1. From a Mirror

To find the freely downloadable distributions of Fedora, look for a *mirror*. A mirror is a computer server open to the public for free downloads of software, including Fedora. Mirrors offer both free open source software and closed source software. To locate a mirror, visit *http://mirrors.fedoraproject.org/*

publiclist using a Web browser, and choose a server from the list. The web page lists mirrors by geographic location. Mirrors geographically closer to you are ideal for faster downloading speeds.

Mirrors publish Fedora software under a well-organized hierarchy of folders. For example, the Fedora 11 distribution normally appears in the directory **fedora/linux/releases/11/**. This directory contains a folder for each architecture supported by that release of Fedora. CD and DVD media files appear inside that folder, in a folder called **iso/**. For example, you can find the file for the DVD distribution of Fedora 11 for x86_64 at **fedora/linux/releases/11/Fedora/x86_64/iso/ Fedora-11-x86_64-DVD.iso**.

2.1.2. From BitTorrent

BitTorrent is a way to download information in cooperation with other computers. Each computer cooperating in the group downloads pieces of the information in a particular torrent from other peers in the group. Computers that have finished downloading all the data in a torrent remain in the swarm to *seed*, or provide data to other peers. If you download using BitTorrent, as a courtesy you should seed the torrent at least until you have uploaded the same amount of data you downloaded.

If your computer does not have software installed for BitTorrent, visit the BitTorrent home page at *http://www.bittorrent.com/download/* to download it. BitTorrent client software is available for Windows, Mac OS, Linux, and many other operating systems.

You do not need to find a special mirror for BitTorrent files. The BitTorrent protocol ensures that your computer participates in a nearby group. To download and use the Fedora BitTorrent files, visit *http:// torrent.fedoraproject.org/*.

> **Minimal Boot Images**
> Minimal boot CD and USB flash disk images are not available through BitTorrent.

2.2. Which Architecture Is My Computer?

Releases are separated by *architecture*, or type of computer processor. Use the following table to determine the architecture of your computer according to the type of processor. Consult your manufacturer's documentation for details on your processor, if necessary.

Processor manufacturer and model	Architecture type for Fedora
Intel (except Atom 230, Atom 330, Core 2 Duo, Centrino Core2 Duo, and recent vintage Xeon); AMD (except Athlon 64, Athlon x2, Sempron 64, and Opteron); VIA C3, C7; Apple MacBook Pro	i386
Intel Atom 230, Atom 330, Core 2 Duo, Centrino Core 2 Duo, and Xeon; AMD Athlon 64, Athlon x2, Sempron64, and Opteron	x86_64
Apple Macintosh G3, G4, G5, PowerBook, and other non-Intel models	ppc

Table 2.1. Processor and architecture types

> **i386 Works for Most Windows Compatible Computers**
> If you are unsure what type of processor your computer uses, choose i386.

> The exception is if your computer is a non-Intel based Apple Macintosh. Refer to *Table 2.1, "Processor and architecture types"* for more information.

Intel Atom Processor Architectures Vary

The N and Z Series Atom processors are based on the `i386` architecture. The 230 and 330 Series Atom processors are based on the `x86_64` architecture. Refer to *http://ark.intel.com/cpugroup.aspx?familyID=29035* for more details.

2.3. Which Files Do I Download?

You have several options to download Fedora. Read the options below to decide the best one for you.

Each file available for download in a Fedora distribution includes the architecture type in the file name. For example, the file for the DVD distribution of Fedora 11 for x86_64 is named **Fedora-11-x86_64-DVD.iso**. Refer to *Section 2.2, "Which Architecture Is My Computer?"* if you are unsure of your computer's architecture.

Full Distribution on DVD

If you have plenty of time, a fast Internet connection, and wish a broader choice of software on the install media, download the full DVD version. Once burned to DVD, the media is bootable and includes an installation program. The DVD version contains a mode to perform rescue operations on your Fedora system in an emergency. You can download the DVD version directly from a mirror, or via BitTorrent.

Live Image

If you want to try Fedora before you install it on your computer, download the Live image version. If your computer supports booting from CD or USB, you can boot the operating system without making any changes to your hard disk. The Live image also provides an **Install to Hard Disk** desktop shortcut. If you decide you like what you see, and want to install it, simply activate the selection to copy Fedora to your hard disk. You can download the Live image directly from a mirror, or using BitTorrent.

Minimal Boot Media

If you have a fast Internet connection but do not want to download the entire distribution, you can download a small boot image. Fedora offers images for a minimal boot environment on CD. Once you boot your system with the minimal media, you can install Fedora directly over the Internet. Although this method still involves downloading a significant amount of data over the Internet, it is almost always much less than the size of the full distribution media. Once you have finished installation, you can add or remove software to your system as desired.

Download Size

Installing the default software for Fedora over the Internet requires more time than the Live image, but less time than the entire DVD distribution. Actual results depend on the software you select and network traffic conditions.

The following table explains where to find the desired files on a mirror site. Replace *arch* with the architecture of the computer being installed.

Media type	File locations
Full distribution on DVD	`fedora/linux/releases/11/Fedora/`*`arch`*`/iso/` `Fedora-11-`*`arch`*`-DVD.iso`

Live image	`fedora/linux/releases/11/Live/`*`arch`*`/iso/`
	`Fedora-11-`*`arch`*`-Live.iso, fedora/linux/releases/11/`
	`Live/`*`arch`*`/iso/Fedora-11-KDE-`*`arch`*`-Live.iso`

Minimal CD boot media	`fedora/linux/releases/11/Fedora/`*`arch`*`/os/images/` `boot.iso`

Table 2.2. Locating files

2.4. How Do I Make Fedora Media?

A Fedora ISO file can be turned into either CD or DVD discs. You can turn Fedora Live ISO files into bootable USB media, as well as a CD or DVD.

2.4.1. Making CD or DVD Discs

To learn how to turn ISO images into CD or DVD media, refer to *http://docs.fedoraproject.org/readme-burning-isos/*.

2.4.2. Making USB Media

To make bootable USB media, use a Fedora Live image. Use either a Windows or Linux system to make the bootable USB media.

> **USB Image Writing is Nondestructive**
>
> Writing the Live image to the USB media is *nondestructive*. Any existing data on the media is not harmed.
>
> *It is always a good idea to back up important data before performing sensitive disk operations.*

To begin, make sure there is sufficient free space available on the USB media. There is no need to repartition or reformat your media. *It is always a good idea to back up important data before performing sensitive disk operations.*

2.4.2.1. USB Image Creation from Windows

1. Download a Live ISO file as explained in *Section 2.3, "Which Files Do I Download?"*.

2. Download the Windows **liveusb-creator** program at *http://fedorahosted.org/liveusb-creator*.

3. Follow the instructions given at the site and in the **liveusb-creator** program to create the bootable USB media.

2.4.2.2. USB Image Creation in Linux

USB media often comes in the form of flash devices sometimes called *pen drives*, *thumb disks*, or *keys*; or as an externally connected hard disk device. Almost all media of this type is formatted as a vfat file system. You can create bootable USB media on media formatted as ext2, ext3, or vfat.

> **Unusual USB Media**
>
> In a few cases with oddly formatted or partitioned USB media, the image writing may fail.

1. Download a Live ISO file as shown in *Section 2.3, "Which Files Do I Download?"*.

2. Install the *livecd-tools* package on your system. For Fedora systems, use the following command:

   ```
   su -c 'yum -y install livecd-tools'
   ```

3. Plug in your USB media.

4. Find the device name for your USB media. If the media has a volume name, look up the name in **/dev/disk/by-label**, or use the **findfs**:

   ```
   su -c 'findfs LABEL="MyLabel"'
   ```

   If the media does not have a volume name, or you do not know it, consult the **/var/log/messages** log for details:

   ```
   su -c 'less /var/log/messages'
   ```

5. Use the **livecd-iso-to-disk** command to write the ISO image to the media:

   ```
   su -c 'livecd-iso-to-disk the_image.iso /dev/sdX1'
   ```

   Replace *sdX1* with the device name for the partition on the USB media. Most flash drives and external hard disks use only one partition. If you have changed this behavior or have oddly partitioned media, you may need to consult other sources of help.

2.5. What If I Cannot Download Fedora?

If you do not have a fast Internet connection, or if you have a problem creating boot media, downloading may not be an option. Fedora DVD and CD distribution media is available from a number of online sources around the world at a minimal cost. Use your favorite Web search engine to locate a vendor, or refer to *http://fedoraproject.org/wiki/Distribution*.

2.6. How Do I Start the Installation Program?

To start the installation program from minimal boot media, a Live image, or the distribution DVD, follow this procedure:

1. Power off your computer system.

2. Disconnect any external FireWire or USB disks that you do not need for installation. Refer to *Section 7.18.1.3, "FireWire and USB Disks"* for more information.

3. Insert the media in your computer and turn it on.

You may need to press a specific key or combination of keys to boot from the media, or configure your system's *Basic Input/Output System*, or BIOS, to boot from the media. On most computers you must select the boot or BIOS option promptly after turning on the computer. Most Windows-compatible computer systems use a special key such as `F1`, `F2`, `F12`, or `Del` to start the BIOS configuration menu. On Apple computers, the `C` key boots the system from the DVD drive. On older Apple hardware you may need to press `Cmd +Opt+Shift+Del` to boot from DVD drive.

> ### Configuring the BIOS
> If you are not sure what capabilities your computer has, or how to configure the BIOS, consult the documentation provided by the manufacturer. Detailed information on hardware specifications and configuration is beyond the scope of this document.

Part I. Before you begin

This part of the *Fedora Installation Guide* covers decisions that you should make and resources that you should gather before installing Fedora, including:

- the decision whether to upgrade an existing installation of Fedora or install a new copy.

- hardware considerations, and hardware details that you may need during installation.

- preparing to install Fedora over a network.

- preparing driver media.

Steps to Get You Started

3.1. Upgrade or Install?

For information to help you determine whether to perform an upgrade or an installation refer to *Chapter 18, Upgrading Your Current System*.

3.2. Is Your Hardware Compatible?

Hardware compatibility is particularly important if you have an older system or a system that you built yourself. Fedora 11 should be compatible with most hardware in systems that were factory built within the last two years. However, hardware specifications change almost daily, so it is difficult to guarantee that your hardware is 100% compatible.

The most recent list of supported hardware can be found in the Release Notes for Fedora 11, available at *http://docs.fedoraproject.org/release-notes* .

3.3. Do You Have Enough Disk Space?

Nearly every modern-day operating system (OS) uses *disk partitions*, and Fedora is no exception. When you install Fedora, you may have to work with disk partitions. If you have not worked with disk partitions before (or need a quick review of the basic concepts), refer to *Appendix A, An Introduction to Disk Partitions* before proceeding.

The disk space used by Fedora must be separate from the disk space used by other OSes you may have installed on your system, such as Windows, OS/2, or even a different version of Linux. For x86, AMD64, and Intel® 64 systems, at least two partitions (`/` and `swap`) must be dedicated to Fedora.

Before you start the installation process, you must

- have enough *unpartitioned*[1] disk space for the installation of Fedora, or

- have one or more partitions that may be deleted, thereby freeing up enough disk space to install Fedora.

To gain a better sense of how much space you really need, refer to the recommended partitioning sizes discussed in *Section 7.21.4, "Recommended Partitioning Scheme"*.

If you are not sure that you meet these conditions, or if you want to know how to create free disk space for your Fedora installation, refer to *Appendix A, An Introduction to Disk Partitions*.

3.4. Can You Install Using the CD-ROM or DVD?

There are several methods that can be used to install Fedora.

Installing from a CD-ROM or DVD requires that you have a Fedora 11 CD-ROM or DVD, and you have a DVD/CD-ROM drive on a system that supports booting from it.

Your BIOS may need to be changed to allow booting from your DVD/CD-ROM drive. For more information about changing your BIOS, refer to *Section 7.3.1, "Booting the Installation Program on x86, AMD64, and Intel 64 Systems"*.

3.4.1. Alternative Boot Methods

Boot DVD/CD-ROM

> If you can boot using the DVD/CD-ROM drive, you can create your own CD-ROM to boot the installation program. This may be useful, for example, if you are performing an installation over a network or from a hard drive. Refer to *Section 3.4.2, "Making an Installation Boot CD-ROM"* for further instructions.

USB pen drive

> If you cannot boot from the DVD/CD-ROM drive, but you can boot using a USB device, such as a USB pen drive, the following alternative boot method is available:

> To boot using a USB pen drive, use the **dd** command to copy the **diskboot.img** image file from the **/images/** directory on the DVD or CD-ROM #1. For example:

```
dd if=diskboot.img of=/dev/sda
```

> Your BIOS must support booting from a USB device in order for this boot method to work.

3.4.2. Making an Installation Boot CD-ROM

isolinux To create your own CD-ROM to boot the installation program, use the following instructions:

Copy the **isolinux/** directory from the Fedora DVD or CD #1 into a temporary directory (referred to here as **<path-to-workspace>**) using the following command:

```
cp -r <path-to-cd>/isolinux/ <path-to-workspace>
```

Change directories to the **<path-to-workspace>** directory you have created:

```
cd <path-to-workspace>
```

Make sure the files you have copied have appropriate permissions:

```
chmod u+w isolinux/*
```

Finally, issue the following command to create the ISO image file:

```
mkisofs -o file.iso -b isolinux.bin -c boot.cat -no-emul-boot \
-boot-load-size 4 -boot-info-table -R -J -v -T isolinux/
```

> **Note**
>
> The above command was split into two lines for printing purposes only. When you execute this command, be sure to type it as a single command, all on the same line.

Burn the resulting ISO image (named **file.iso** and located in **<path-to-workspace>**) to a CD-ROM as you normally would.

3.5. Preparing for a Network Installation

> **Note**
>
> Make sure an installation CD (or any other type of CD) is not in your system's CD/DVD drive if you are performing a network-based installation. Having a CD in the drive may cause unexpected errors.

The Fedora installation media must be available for either a network installation (via NFS, FTP, or HTTP) or installation via local storage. Use the following steps if you are performing an NFS, FTP, or HTTP installation.

The NFS, FTP, or HTTP server to be used for installation over the network must be a separate machine which can provide the complete contents of the installation DVD-ROM or the installation CD-ROMs.

> **Note**
>
> The Fedora installation program has the ability to test the integrity of the installation media. It works with the CD / DVD, hard drive ISO, and NFS ISO installation methods. We recommend that you test all installation media before starting the installation process, and before reporting any installation-related bugs (many of the bugs reported are actually due to improperly-burned CDs). To use this test, type the following command at the `boot:` prompt:
>
> ```
> linux mediacheck
> ```

> **Note**
>
> In the following examples, the directory on the installation staging server that will contain the installation files will be specified as */location/of/disk/space*. The directory that will be made publicly available via FTP, NFS, or HTTP will be specified as */publicly/available/directory*. For example, */location/of/disk/space* may be a directory you create called **/var/isos**. */publicly/available/directory* might be **/var/www/html/f11**, for an HTTP install.

To copy the files from the installation DVD or CD-ROMs to a Linux machine which acts as an installation staging server, perform the following steps:

- Create an iso image from the installation disk(s) using the following command (for DVDs):

    dd if=/dev/*dvd* of=/*location/of/disk/space*/f11.iso

    where *dvd* refers to your DVD drive device.

For instructions on how to prepare a network installation using CD-ROMs, refer to the instructions on the **README-en** file in **disk1**.

3.5.1. Preparing for FTP and HTTP installation

For FTP and HTTP installation, the iso image or images should be mounted via loopback in the publicly available directory, in the following manner:

- For DVD:

mount -o loop *`/location/of/disk/space`***/f11.iso** *`/publicly/available/`*
*`directory`***/**

In this case *`/publicly/available/directory`* will be a directory that is shared via FTP or HTTP.

- For CDROMs:

mount -o loop *`/location/of/disk/space`***/disk*X*.iso** *`/publicly/available/`*
*`directory`***/disk*X*/**

Again, *`/publicly/available/directory`* will be a directory that is shared via FTP or HTTP. Do the above for each of the CDROM iso images, for example:

mount -o loop /var/isos/Fedora-11-i386-disk1.iso /var/www/html/f11-
install/disk1/

Example 3.1. Making a Fedora installation CD available on a HTTP server

Next make sure that the *`/publicly/available/directory`* directory is shared via FTP or HTTP, and verify client access. You can check to see whether the directory is accessible from the server itself, and then from another machine on the same subnet that you will be installing to.

3.5.2. Preparing for an NFS install

For NFS installation it is not necessary to mount the iso image. It is sufficient to make the iso image itself available via NFS. You can do this by moving the iso image or images to the NFS exported directory:

- For DVD:

mv *`/location/of/disk/space`***/f11.iso** *`/publicly/available/directory`***/**

- For CDROMs:

mv *`/location/of/disk/space`***/f11-disk*.iso** *`/publicly/available/directory`***/**

Ensure that the *`/publicly/available/directory`* directory is exported via NFS via an entry in **/etc/exports**.

To export to a specific system:

`/publicly/available/directory` **client.ip.address(ro,no_root_squash)**

To export to all systems use an entry such as:

/publicly/available/directory ***(ro,no_root_squash)**

Start the NFS daemon (on a Fedora system, use **/sbin/service nfs start**). If NFS is already running, reload the configuration file (on a Fedora system use **/sbin/service nfs reload**).

3.6. Preparing for a Hard Drive Installation

> **Note**
>
> Hard drive installations only work from ext2, ext3, or FAT file systems. If you have a file system other than those listed here, such as reiserfs, you will not be able to perform a hard drive installation.

Hard drive installations require the use of the ISO (or DVD/CD-ROM) images. An ISO image is a file containing an exact copy of a DVD/CD-ROM image. After placing the required ISO images (the binary Fedora DVD/CD-ROMs) in a directory, choose to install from the hard drive. You can then point the installation program at that directory to perform the installation.

To prepare your system for a hard drive installation, you must set the system up in one of the following ways:

• Using a set of CD-ROMs, or a DVD — Create ISO image files from each installation CD-ROM, or from the DVD. For each CD-ROM (once for the DVD), execute the following command on a Linux system:

```
dd if=/dev/cdrom of=/tmp/file-name.iso
```

• Using ISO images — transfer these images to the system to be installed.

Verifying that ISO images are intact before you attempt an installation, helps to avoid problems. To verify the ISO images are intact prior to performing an installation, use an **md5sum** program (many **md5sum** programs are available for various operating systems). An **md5sum** program should be available on the same Linux machine as the ISO images.

> **Note**
>
> The Fedora installation program has the ability to test the integrity of the installation media. It works with the CD / DVD, hard drive ISO, and NFS ISO installation methods. We recommend that you test all installation media before starting the installation process, and before reporting any installation-related bugs (many of the bugs reported are actually due to improperly-burned CDs). To use this test, type the following command at the boot: prompt:
>
> ```
> linux mediacheck
> ```

Additionally, if a file called **updates.img** exists in the location from which you install, it is used for updates to **anaconda**, the installation program. Refer to the file **install-methods.txt** in the

`anaconda` RPM package for detailed information on the various ways to install Fedora, as well as how to apply the installation program updates.

System Specifications List

The installation program automatically detects and installs your computer's hardware. Although you should make sure that your hardware meets the minimum requirements to install Fedora (refer to *Section 3.2, "Is Your Hardware Compatible?"*) you do not usually need to supply the installation program with any specific details about your system.

However, when performing certain types of installation, some specific details might be useful or even essential.

- If you plan to use a customized partition layout, record:

  - The model numbers, sizes, types, and interfaces of the hard drives attached to the system. For example, Seagate ST3320613AS 320 GB on SATA0, Western Digital WD7500AAKS 750 GB on SATA1. This will allow you to identify specific hard drives during the partitioning process.

- If you are installing Fedora as an additional operating system on an existing system, record:

  - The mount points of the existing partitions on the system. For example, `/boot` on `sda1`, `/` on `sda2`, and `/home` on `sdb1`. This will allow you to identify specific partitions during the partitioning process.

- If you plan to install from an image on a local hard drive:

  - The hard drive and directory that holds the image – see *Table 7.2, "Location of ISO images for different partition types"* for examples.

- If you plan to install from a network location, or install on an iSCSI target:

  - The make and model numbers of the network adapters on your system. For example, Netgear GA311. This will allow you to identify adapters when manually configuring the network.

  - IP, DHCP, and BOOTP addresses

  - Netmask

  - Gateway IP address

  - One or more name server IP addresses (DNS)

  If any of these networking requirements or terms are unfamiliar to you, contact your network administrator for assistance.

- If you plan to install from a network location:

  - The location of the image on an FTP server, HTTP (web) server, or NFS server – see *Section 7.9, "Installing via FTP or HTTP"* and *Section 7.8, "Installing via NFS"* for examples.

- If you plan to install on an iSCSI target:

  - The location of the iSCSI target. Depending on your network, you might also need a CHAP username and password, and perhaps a reverse CHAP username and password – see *Section 7.19, " Advanced Storage Options "*.

- If your computer is part of a domain:

- You should verify that the domain name will be supplied by the DHCP server. If not, you will need to input the domain name manually during installation.

Driver Media for Intel and AMD Systems

5.1. Why Do I Need Driver Media?

While the Fedora installation program is loading, a screen may appear asking you for driver media. The driver media screen is most often seen in the following scenarios:

- If you need to perform an installation from a block device

- If you need to perform an installation from a PCMCIA device

- If you run the installation program by entering `linux dd` at the installation boot prompt

- If you run the installation program on a computer which does not have any PCI devices

5.2. So What Is Driver Media Anyway?

Driver media can add support for hardware that may or may not be supported by the installation program. Driver media could include a driver diskette or image produced by Red Hat, it could be a diskette or CD-ROM you make yourself from driver images found on the Internet, or it could be a diskette or CD-ROM that a hardware vendor includes with a piece of hardware.

Driver media is used if you need access to a particular device to install Fedora. Drivers can be used for network (NFS) installations, installations using a PCMCIA or block device, non-standard or very new optical drives, SCSI adapters, NICs, and other uncommon devices.

> **Note**
>
> If an unsupported device is not needed to install Fedora on your system, continue with the installation and add support for the new piece of hardware once the installation is complete.

5.3. How Do I Obtain Driver Media?

Driver images may be available from a hardware or software vendor's website. If you suspect that your system may require one of these drivers, you should create a driver diskette or CD-ROM before beginning your Fedora installation.

> **Note**
>
> It is also possible to use a driver image via a network file. Instead of using the `linux dd` boot command, use the `linux dd=url` command, where `url` is replaced by an HTTP, FTP, or NFS address of the driver image to be used.

5.3.1. Creating a Driver Diskette from an Image File

To create a driver diskette from a driver diskette image using Linux:

1. Insert a blank, formatted diskette into the first diskette drive.

2. From the same directory containing the driver diskette image, such as *drvnet.img*, type **dd if=*drvnet.img* of=/dev/fd0** as root.

> **Note**
>
> The installation program supports using an external flash drive as a way to add driver images during the installation process. The best way to do this is to mount the flash drive and copy the desired **driverdisk.img** onto the flash drive. For example:
>
> ```
> dd if=driverdisk.img of=/dev/sda
> ```
>
> You are then prompted during the installation to select the partition and specify the file to be used.

5.4. Using a Driver Image During Installation

If you need to use a driver image, such as during a PCMCIA device or NFS installation, the installation program prompts you to insert the driver (as a diskette, CD-ROM, or file name) when it is needed.

For example, to specifically load a driver diskette that you have created, begin the installation process by booting from the Fedora DVD (or using boot media you have created). For x86-based systems, at the `boot:` prompt, enter **linux dd** if using an x86 or x86-64 system. Refer to *Section 7.3.1, "Booting the Installation Program on x86, AMD64, and Intel 64 Systems"* for details on booting the installation program.

The installation program asks you to insert the driver diskette. Once the driver diskette is read by the installation program, it can apply those drivers to hardware discovered on your system later in the installation process.

Part II. The installation process

This part of the *Fedora Installation Guide* details the installation process itself, from various methods of booting the installer up to the point where the computer must restart to finalize the installation. This part of the manual also includes a chapter on troubleshooting problems with the installation process.

Beginning the Installation

> **Aborting the Installation**
>
> To abort the installation, either press `Ctrl +Alt+Del` or power off your computer with the power switch. You may abort the installation process without consequence at any time prior to selecting **Write changes to disk** on the **Write partitioning to disk** screen. Fedora makes no permanent changes to your computer until that point. Please be aware that stopping the installation after partitioning has begun can leave your computer unusable.

6.1. The Boot Menu

The boot media displays a graphical boot menu with several options. If no key is hit within 60 seconds, the default boot option runs. To choose the default, either wait for the timer to run out or hit **Enter** on the keyboard. To select a different option than the default, use the arrow keys on your keyboard, and hit **Enter** when the correct option is highlighted. If you want to customize the boot options for a particular option, hit the **Tab** key.

> **Using Boot Options**
>
> For a listing and explanation of common boot options, refer to *Chapter 9, Boot Options*.

When using Fedora Live media, press any key during the initial boot countdown to bring up the **Boot Options** menu. The boot options include:

Boot

   This option is the default. If you select this option, only the kernel and startup programs load into memory. This option takes less time to load. As you use programs, they are loaded from the disc, which takes more time. This mode can be used on machines with less total memory.

Verify and Boot

   This option lets you verify the disc before you run the Live CD environment. Refer to *Section 6.3, "Verifying Media"* for more information on the verification process.

Memory Test

   This option runs an exhaustive test on the memory on your system. For more information, refer to *Section 9.6.1, "Loading the Memory (RAM) Testing Mode"*.

Boot from local drive

   This option boots the system from the first installed disk. If you booted this disc accidentally, use this option to boot from the hard disk immediately without starting the installer.

If you boot the DVD, rescue CD, or minimal boot media, the boot menu options include:

Install or upgrade an existing system

   This option is the default. Choose this option to install Fedora onto your computer system using the graphical installation program.

Install system with basic video driver

This option allows you to install Fedora in graphical mode even if the installation program is unable to load the correct driver for your video card. If your screen appears distorted or goes blank when using the the **Install or upgrade an existing system** option, restart your computer and try this option instead.

Rescue installed system

Choose this option to repair a problem with your installed Fedora system that prevents you from booting normally. Although Fedora is an exceptionally stable computing platform, it is still possible for occasional problems to occur that prevent booting. The rescue environment contains utility programs that allow you fix a wide variety of these problems.

Boot from local drive

(as for Live CD)

Memory Test

(as for Live CD)

6.2. Installing from a Different Source

All boot media except the distribution DVD present a menu that allows you to choose the installation source, such as the network or a hard disk. If you are booting the distribution DVD and do not want to install from the DVD, hit **Tab** at the boot menu. Add a space and the option `linux askmethod` to the end of the line that appears below the menu.

You can install Fedora from the ISO images stored on hard disk, or from a network using NFS, FTP, or HTTP methods. Experienced users frequently use one of these methods because it is often faster to read data from a hard disk or network server than from a CD or DVD.

The following table summarizes the different boot methods and recommended installation methods to use with each:

Boot method	Installation method
DVD	DVD, network, or hard disk
Minimal boot CD or USB, rescue CD	Network or hard disk
Live CD or USB	**Install to Hard Disk** application

Table 6.1. Boot methods and installation methods

Section 7.4, "Selecting an Installation Method" contains detailed information about installing from alternate locations.

6.3. Verifying Media

The distribution DVD media and the Live CD media offer an option to verify the integrity of the media. Recording errors sometimes occur while producing CD or DVD media in home computer equipment. An error in the data for package chosen in the installation program can cause the installation to abort. To minimize the chances of data errors affecting the installation, verify the media before installing.

6.3.1. Verifying the Live CD

If you boot from the Live CD, choose **Verify and Boot** from the boot menu. The verification process runs automatically during the boot process, and if it succeeds, the Live CD continues loading. If the verification fails, create a new Live CD using the ISO image you downloaded earlier.

6.3.2. Verifying the DVD

If you boot from the Fedora distribution DVD, the option to verify the media appears after you choose to install Fedora. If the verification succeeds, the installation process proceeds normally. If the process fails, create a new DVD using the ISO image you downloaded earlier.

6.4. Booting from the Network using PXE

To boot with
PXE, you need a properly configured server, and a network interface in your computer that supports PXE. For information on how to configure a PXE server, refer to *Chapter 11, Setting Up an Installation Server*.

Configure the computer to boot from the network interface. This option is in the BIOS, and may be labeled `Network Boot` or `Boot Services`. Once you properly configure PXE booting, the computer can boot the Fedora installation system without any other media.

To boot a computer from a PXE server:

1. Ensure that the network cable is attached. The link indicator light on the network socket should be lit, even if the computer is not switched on.

2. Switch on the computer.

3. A menu screen appears. Press the number key that corresponds to the desired option.

> ### PXE Troubleshooting
>
> If your PC does not boot from the netboot server, ensure that the BIOS is configured to boot first from the correct network interface. Some BIOS systems specify the network interface as a possible boot device, but do not support the PXE standard. Refer to your hardware documentation for more information.

> ### Multiple NICs and PXE installation
>
> Some servers with multiple network interfaces may not assign eth0 to the first network interface as BIOS knows it, which can cause the installer to try using a different network interface than was used by PXE. To change this behavior, use the following in **pxelinux.cfg/*** config files:
>
> ```
> IPAPPEND 2
>
> APPEND ksdevice=bootif
> ```
>
> The configuration options above causes the installer to use the same network interface as BIOS and PXE use. You can also use the following option:

```
ksdevice=link
```

This option causes the installer to use the first network device it finds that is linked to a network switch.

6.5. Graphical and Text Interfaces

Fedora 11 supports graphical and text-based installations. However, the installer image must either fit in RAM or appear on local storage, such as the installation DVD or Live Media. Therefore, only systems with more than 192MB of RAM or that boot from the installation DVD or Live Media can use the graphical installer. Systems with 192MB RAM or less automatically scale back to using the text-based installer. If you prefer to use the text-based installer, type `linux text` at the `boot:` prompt.

If one of the following situations occurs, the installation program uses a text mode:

- The installation system fails to identify the display hardware on your computer

- Your computer has less than 192 MB of RAM

- You choose the text mode installation from the boot menu

The text screens provide most of the same functions as the standard screens, although disk partitioning is simplified, and bootloader configuration and package selection are handled automatically in text mode. If you choose to install Fedora in text mode, you can still configure your system to use a graphical interface after installation.

Graphical Interface Usage

Installing in text mode does **not** prevent you from using a graphical interface on your system once it is installed. If you have trouble configuring your system for graphical interface use, consult other sources for troubleshooting help as shown in *Section 1.2, "Getting Additional Help"*.

Installing on Intel and AMD Systems

This chapter explains how to perform a Fedora installation from the DVD/CD-ROM, using the graphical, mouse-based installation program. The following topics are discussed:

- Becoming familiar with the installation program's user interface

- Starting the installation program

- Selecting an installation method

- Configuration steps during the installation (language, keyboard, mouse, partitioning, etc.)

- Finishing the installation

7.1. The Graphical Installation Program User Interface

If you have used a *graphical user interface (GUI)* before, you are already familiar with this process; use your mouse to navigate the screens, click buttons, or enter text fields.

You can also navigate through the installation using the keyboard. The **Tab** key allows you to move around the screen, the Up and Down arrow keys to scroll through lists, **+** and **-** keys expand and collapse lists, while **Space** and **Enter** selects or removes from selection a highlighted item. You can also use the **Alt+X** key command combination as a way of clicking on buttons or making other screen selections, where **X** is replaced with any underlined letter appearing within that screen.

> **Note**
>
> If you are using an x86, AMD64, or Intel® 64 system, and you do not wish to use the GUI installation program, the text mode installation program is also available. To start the text mode installation program, press the **Esc** key while the Fedora boot menu is displayed, then use the following command at the `boot:` prompt:
>
> ```
> linux text
> ```
>
> Refer to *Section 6.1, "The Boot Menu"* for a description of the Fedora boot menu and to *Section 7.2, "The Text Mode Installation Program User Interface"* for a brief overview of text mode installation instructions.
>
> It is highly recommended that installs be performed using the GUI installation program. The GUI installation program offers the full functionality of the Fedora installation program, including LVM configuration which is not available during a text mode installation.
>
> Users who must use the text mode installation program can follow the GUI installation instructions and obtain all needed information.

7.1.1. A Note about Virtual Consoles

The Fedora installation program offers more than the dialog boxes of the installation process. Several kinds of diagnostic messages are available to you, as well as a way to enter commands from a shell prompt. The installation program displays these messages on five *virtual consoles*, among which you can switch using a single keystroke combination.

A virtual console is a shell prompt in a non-graphical environment, accessed from the physical machine, not remotely. Multiple virtual consoles can be accessed simultaneously.

These virtual consoles can be helpful if you encounter a problem while installing Fedora. Messages displayed on the installation or system consoles can help pinpoint a problem. Refer to *Table 7.1, "Console, Keystrokes, and Contents"* for a listing of the virtual consoles, keystrokes used to switch to them, and their contents.

Generally, there is no reason to leave the default console (virtual console #6) for graphical installations unless you are attempting to diagnose installation problems.

console	keystrokes	contents
1	`ctrl+alt+f1`	installation dialog
2	`ctrl+alt+f2`	shell prompt
3	`ctrl+alt+f3`	install log (messages from installation program)
4	`ctrl+alt+f4`	system-related messages
5	`ctrl+alt+f5`	other messages
6	`ctrl+alt+f6`	x graphical display

Table 7.1. Console, Keystrokes, and Contents

7.2. The Text Mode Installation Program User Interface

The Fedora text mode installation program uses a screen-based interface that includes most of the on-screen *widgets* commonly found on graphical user interfaces. *Figure 7.1, "Installation Program Widgets as seen in Boot Loader Configuration"*, and *Figure 7.2, "Installation Program Widgets as seen in Disk Druid"*, illustrate the screens that appear during the installation process.

The cursor is used to select (and interact with) a particular widget. As the cursor is moved from widget to widget, it may cause the widget to change color, or the cursor itself may only appear positioned in or next to the widget.

> **Note**
>
> While text mode installations are not explicitly documented, those using the text mode installation program can easily follow the GUI installation instructions. One thing to note is that manipulation of LVM (Logical Volume Management) disk volumes is only possible in graphical mode. In text mode it is only possible to view and accept the default LVM setup.

> **Note**
>
> Not every language supported in graphical installation mode is also supported in text mode. Specifically, languages written with a character set other than the Latin or Cyrillic alphabets are not available in text mode. If you choose a language written with a character set that is not supported in text mode, the installation program will present you with the English versions of the screens.
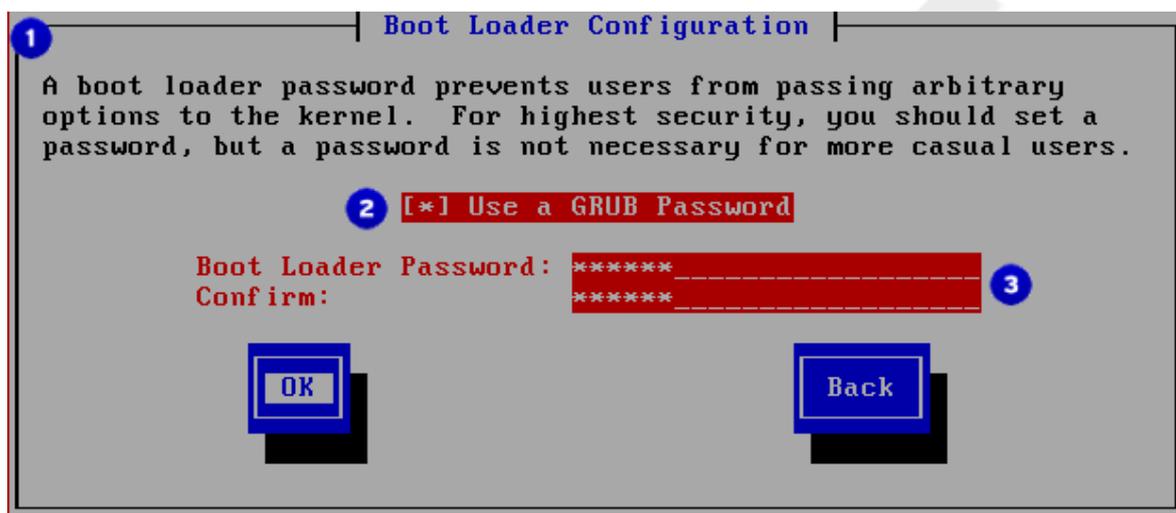
Figure 7.1. Installation Program Widgets as seen in **Boot Loader Configuration**

Legend

1. Window — Windows (usually referred to as *dialogs* in this manual) appear on your screen throughout the installation process. At times, one window may overlay another; in these cases, you can only interact with the window on top. When you are finished in that window, it disappears, allowing you to continue working in the window underneath.

2. Checkbox — Checkboxes allow you to select or deselect a feature. The box displays either an asterisk (selected) or a space (unselected). When the cursor is within a checkbox, press **Space** to select or deselect a feature.

3. Text Input — Text input lines are regions where you can enter information required by the installation program. When the cursor rests on a text input line, you may enter and/or edit information on that line.
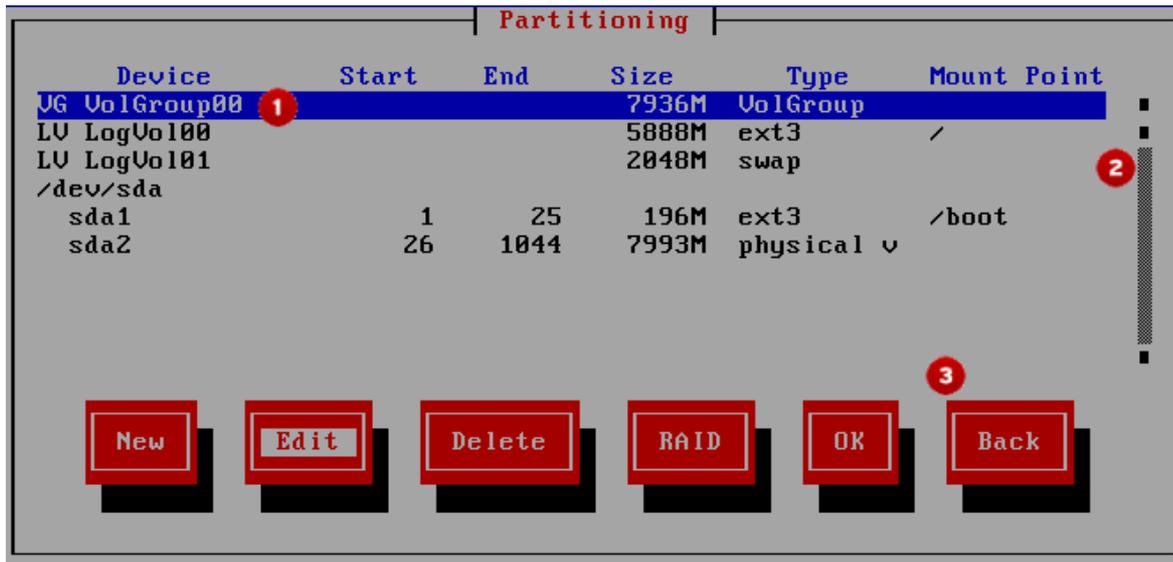
Figure 7.2. Installation Program Widgets as seen in **Disk Druid**

Legend

1. Text Widget — Text widgets are regions of the screen for the display of text. At times, text widgets may also contain other widgets, such as checkboxes. If a text widget contains more information than can be displayed in the space reserved for it, a scroll bar appears; if you position the cursor within the text widget, you can then use the `Up` and `Down` arrow keys to scroll through all the information available. Your current position is shown on the scroll bar by a **#** character, which moves up and down the scroll bar as you scroll.

2. Scroll Bar — Scroll bars appear on the side or bottom of a window to control which part of a list or document is currently in the window's frame. The scroll bar makes it easy to move to any part of a file.

3. Button Widget — Button widgets are the primary method of interacting with the installation program. You progress through the windows of the installation program by navigating these buttons, using the `Tab` and `Enter` keys. Buttons can be selected when they are highlighted.

7.2.1. Using the Keyboard to Navigate

Navigation through the installation dialogs is performed through a simple set of keystrokes. To move the cursor, use the `Left`, `Right`, `Up`, and `Down` arrow keys. Use `Tab`, and `Shift-Tab` to cycle forward or backward through each widget on the screen. Along the bottom, most screens display a summary of available cursor positioning keys.

To "press" a button, position the cursor over the button (using `Tab`, for example) and press `Space` or `Enter`. To select an item from a list of items, move the cursor to the item you wish to select and press `Enter`. To select an item with a checkbox, move the cursor to the checkbox and press `Space` to select an item. To deselect, press `Space` a second time.

Pressing `F12` accepts the current values and proceeds to the next dialog; it is equivalent to pressing the **OK** button.

> ⚠ **Warning**
>
> Unless a dialog box is waiting for your input, do not press any keys during the installation process (doing so may result in unpredictable behavior).

7.3. Starting the Installation Program

To start, first make sure that you have all necessary resources for the installation. If you have already read through *Chapter 3, Steps to Get You Started*, and followed the instructions, you should be ready to start the installation process. When you have verified that you are ready to begin, boot the installation program using the Fedora DVD or CD-ROM #1 or any boot media that you have created.

> 📝 **Note**
>
> Occasionally, some hardware components require a *driver diskette* during the installation. A driver diskette adds support for hardware that is not otherwise supported by the installation program. Refer to *Chapter 5, Driver Media for Intel and AMD Systems* for more information.

7.3.1. Booting the Installation Program on x86, AMD64, and Intel® 64 Systems

You can boot the installation program using any one of the following media (depending upon what your system can support):

- *Fedora DVD/CD-ROM* — Your machine supports a bootable DVD/CD-ROM drive and you have the Fedora CD-ROM set or DVD.

- *Boot CD-ROM* — Your machine supports a bootable CD-ROM drive and you want to perform network or hard drive installation.

- *USB pen drive* — Your machine supports booting from a USB device.

- *PXE boot via network* — Your machine supports booting from the network. This is an advanced installation path. Refer to *Chapter 11, Setting Up an Installation Server* for additional information on this method.

To create a boot CD-ROM or to prepare your USB pen drive for installation, refer to *Section 3.4.2, "Making an Installation Boot CD-ROM"*.

Insert the boot media and reboot the system. Your BIOS settings may need to be changed to allow you to boot from the CD-ROM or USB device.

> 📝 **Note**
>
> To change your BIOS settings on an x86, AMD64, or Intel® 64 system, watch the instructions provided on your display when your computer first boots. A line of text appears, telling you which key to press to enter the BIOS settings.
>
> Once you have entered your BIOS setup program, find the section where you can alter your boot sequence. The default is often C, A or A, C (depending on whether you boot

> from your hard drive [C] or a diskette drive [A]). Change this sequence so that the CD-ROM is first in your boot order and that C or A (whichever is your typical boot default) is second. This instructs the computer to first look at the CD-ROM drive for bootable media; if it does not find bootable media on the CD-ROM drive, it then checks your hard drive or diskette drive.
>
> Save your changes before exiting the BIOS. For more information, refer to the documentation that came with your system.

After a short delay, a screen containing the `boot:` prompt should appear. The screen contains information on a variety of boot options. Each boot option also has one or more help screens associated with it. To access a help screen, press the appropriate function key as listed in the line at the bottom of the screen.

As you boot the installation program, be aware of two issues:

- Once the `boot:` prompt appears, the installation program automatically begins if you take no action within the first minute. To disable this feature, press one of the help screen function keys.

- If you press a help screen function key, there is a slight delay while the help screen is read from the boot media.

Normally, you only need to press **Enter** to boot. Be sure to watch the boot messages to review if the Linux kernel detects your hardware. If your hardware is properly detected, continue to the next section. If it does not properly detect your hardware, you may need to restart the installation and use one of the boot options provided in *Chapter 9, Boot Options*.

7.3.2. Additional Boot Options

While it is easiest to boot using a CD-ROM or DVD and perform a graphical installation, sometimes there are installation scenarios where booting in a different manner may be needed. This section discusses additional boot options available for Fedora.

To pass options to the boot loader on an x86, AMD64, or Intel® 64 system, use the instructions as provided in the boot loader option samples below.

> **Note**
>
> Refer to *Chapter 9, Boot Options* for additional boot options not covered in this section.

- To perform a text mode installation, at the installation boot prompt, type:

```
linux text
```

- ISO images have an md5sum embedded in them. To test the checksum integrity of an ISO image, at the installation boot prompt, type:

```
linux mediacheck
```

The installation program prompts you to insert a CD or select an ISO image to test, and select **OK** to perform the checksum operation. This checksum operation can be performed on any Fedora CD and does not have to be performed in a specific order (for example, CD #1 does not have to be the first CD you verify). It is strongly recommended to perform this operation on any Fedora CD that was created from downloaded ISO images. This command works with the CD, DVD, hard drive ISO, and NFS ISO installation methods.

- Also in the `images/` directory is the `boot.iso` file. This file is an ISO image than can be used to boot the installation program. To use the `boot.iso`, your computer must be able to boot from its CD-ROM drive, and its BIOS settings must be configured to do so. You must then burn the `boot.iso` file onto a recordable/rewriteable CD-ROM.

- If you need to perform the installation in *serial mode*, type the following command:

```
linux console=<device>
```

For text mode installations, use:

```
linux text console=<device>
```

In the above command, *<device>* should be the device you are using (such as ttyS0 or ttyS1). For example, `linux text console=ttyS0`.

Text mode installations using a serial terminal work best when the terminal supports UTF-8. Under UNIX and Linux, Kermit supports UTF-8. For Windows, Kermit '95 works well. Non-UTF-8 capable terminals works as long as only English is used during the installation process. An enhanced serial display can be used by passing the `utf8` command as a boot-time option to the installation program. For example:

```
linux console=ttyS0 utf8
```

7.3.2.1. Kernel Options

Options can also be passed to the kernel. For example, to apply updates for the anaconda installation program from a floppy disk enter:

```
linux updates
```

For text mode installations, use:

```
linux text updates
```

This command will prompt you to insert a floppy diskette containing updates for anaconda. It is not needed if you are performing a network installation and have already placed the updates image contents in **rhupdates/** on the server.

After entering any options, press **Enter** to boot using those options.

If you need to specify boot options to identify your hardware, please write them down. The boot options are needed during the boot loader configuration portion of the installation (refer to *Section 7.22, "x86, AMD64, and Intel 64 Boot Loader Configuration"* for more information).

For more information on kernel options refer to *Chapter 9, Boot Options*.

7.4. Selecting an Installation Method

What type of installation method do you wish to use? The following installation methods are available:

DVD/CD-ROM

> If you have a DVD/CD-ROM drive and the Fedora CD-ROMs or DVD you can use this method. Refer to *Section 7.5, "Installing from DVD/CD-ROM"*, for DVD/CD-ROM installation instructions.

Hard Drive

> If you have copied the Fedora ISO images to a local hard drive, you can use this method. You need a boot CD-ROM (use the **linux askmethod** boot option). Refer to *Section 7.6, "Installing from a Hard Drive"*, for hard drive installation instructions.

NFS

> If you are installing from an NFS server using ISO images or a mirror image of Fedora, you can use this method. You need a boot CD-ROM (use the **linux askmethod** boot option). Refer to *Section 7.8, "Installing via NFS"* for network installation instructions. Note that NFS installations may also be performed in GUI mode.

URL

> If you are installing directly from an HTTP (Web) server or FTP server, use this method. You need a boot CD-ROM (use the **linux askmethod** boot option). Refer to *Section 7.9, "Installing via FTP or HTTP"*, for FTP and HTTP installation instructions.

If you booted the distribution DVD and did not use the alternate installation source option askmethod, the next stage loads automatically from the DVD. Proceed to *Section 7.10, "Welcome to Fedora"*.

> ### CD/DVD Activity
>
> If you boot any Fedora installation media, the installation program loads its next stage from that disc. This happens regardless of which installation method you choose, unless you eject the disc before you proceed. The installation program still downloads *package data* from the source you choose.

7.5. Installing from DVD/CD-ROM

To install Fedora from a DVD/CD-ROM, place the DVD or CD #1 in your DVD/CD-ROM drive and boot your system from the DVD/CD-ROM. Even if you booted from alternative media, you can still install Fedora from CD or DVD media.

The installation program then probes your system and attempts to identify your CD-ROM drive. It starts by looking for an IDE (also known as an ATAPI) CD-ROM drive.

> **Note**
>
> To abort the installation process at this time, reboot your machine and then eject the boot media. You can safely cancel the installation at any point before the **About to Install** screen. Refer to *Section 7.24, "Preparing to Install"* for more information.

If your CD-ROM drive is not detected, and it is a SCSI CD-ROM, the installation program prompts you to choose a SCSI driver. Choose the driver that most closely resembles your adapter. You may specify options for the driver if necessary; however, most drivers detect your SCSI adapter automatically.

If the DVD/CD-ROM drive is found and the driver loaded, the installer will present you with the option to perform a media check on the DVD/CD-ROM. This will take some time, and you may opt to skip over this step. However, if you later encounter problems with the installer, you should reboot and perform the media check before calling for support. From the media check dialog, continue to the next stage of the installation process (refer to *Section 7.10, "Welcome to Fedora"*).

7.5.1. What If the IDE CD-ROM Was Not Found?

If you have an IDE (ATAPI) DVD/CD-ROM but the installation program fails to find it and asks you what type of DVD/CD-ROM drive you have, try the following boot command. Restart the installation, and at the `boot:` prompt enter **linux hd*X*=cdrom**. Replace *X* with one of the following letters, depending on the interface the unit is connected to, and whether it is configured as master or slave (also known as primary and secondary):

- **a** — first IDE controller, master

- **b** — first IDE controller, slave

- **c** — second IDE controller, master

- **d** — second IDE controller, slave

If you have a third and/or fourth controller, continue assigning letters in alphabetical order, going from controller to controller, and master to slave.

7.6. Installing from a Hard Drive

The **Select Partition** screen applies only if you are installing from a disk partition (that is, if you selected **Hard Drive** in the **Installation Method** dialog). This dialog allows you to name the disk partition and directory from which you are installing Fedora.

The ISO files must be located on a hard drive that is either internal to the computer, or attached to the machine by USB. In addition the **install.img** file from within the ISO files must be copied to a directory named **images** Use this option to install Fedora on computers that are without a network connection or CD/DVD drives.

To extract the **install.img** from the ISO follow the procedure:

```
mount -t iso9660 /path/to/Fedora11.iso /mnt/point -o loop,ro
cp -pr /mnt/point/images /path/images/
umount /mnt/point
```

The partition on the hard drive holding the ISO files must be formatted with the ext2, ext3, or vfat file system. In Fedora, vfat includes a range of file systems such as FAT16 and FAT32 found on most removable media. External hard drives usually contain vfat (FAT32) file systems. Some Microsoft Windows systems also use vfat file systems on internal hard disk partitions.

Before you begin installation from a hard drive, check the partition type to ensure that Fedora can read it. To check a partition's file system under Windows, use the **Disk Management** tool. To check a partition's file system under Linux, use the **fdisk** utility.

> ## Cannot Install from LVM Partitions
> You cannot use ISO files on partitions controlled by LVM (Logical Volume Management).

Figure 7.3. Selecting Partition Dialog for Hard Drive Installation

Select the partition containing the ISO files from the list of available partitions. Internal IDE, SATA, SCSI, and USB drive device names begin with **/dev/sd**. Each individual drive has its own letter, for example **/dev/sda**. Each partition on a drive is numbered, for example **/dev/sda1**.

Also specify the **Directory holding images**. Enter the full directory path from the drive that contains the ISO image files. The following table shows some examples of how to enter this information:

Partition type	Volume	Original path to files	Directory to use
VFAT, NTFS	D:\	D:\Downloads\F11	/Downloads/F11
ext2, ext3	/home	/home/user1/F11	/user1/F11

Table 7.2. Location of ISO images for different partition types

If the ISO images are in the root (top-level) directory of a partition, enter a `/`. If the ISO images are located in a subdirectory of a mounted partition, enter the name of the directory holding the ISO images within that partition. For example, if the partition on which the ISO images is normally mounted as `/home/`, and the images are in `/home/new/`, you would enter `/new/`.

> ### ⭐ Use a leading slash
> An entry without a leading slash may cause the installation to fail.

Select **OK** to continue. Proceed with *Section 7.10, "Welcome to Fedora"*.

7.7. Performing a Network Installation

The installation program is network-aware and can use network settings for a number of functions. For instance, you can install Fedora from a network server using FTP, HTTP, or NFS protocols. You can also instruct the installation program to consult additional software repositories later in the process.

If you are performing a network installation, the **Configure TCP/IP** dialog appears. This dialog asks for your IP and other network addresses. You can choose to configure the IP address and Netmask of the device via DHCP or manually.

By default, the installation program uses DHCP to automatically provide network settings. If you use a cable or DSL modem, router, firewall, or other network hardware to communicate with the Internet, DHCP is a suitable option. If your network has no DHCP server, clear the check box labeled **Use dynamic IP configuration (DHCP)**.

Enter the IP address you are using during installation and press **Enter**.

The installation program supports only the IPv4 protocol. Refer also to *Section 7.15, "Network Configuration"* for more information on configuring your network.
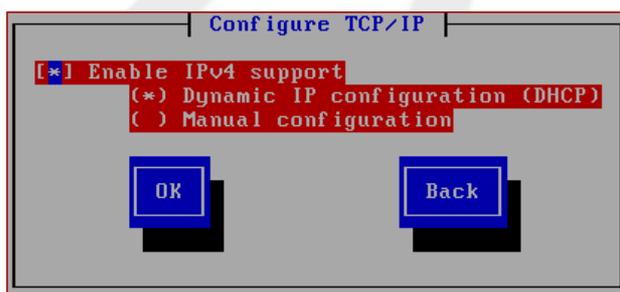
Figure 7.4. TCP/IP Configuration

These settings apply only during the installation process. The installation program allows you to configure the final network configuration later.

You can install from a Web, FTP, or NFS server either on your local network or, if you are connected, on the Internet. You may install Fedora from your own private mirror, or use one of the public mirrors maintained by members of the community. To ensure that the connection is as fast and reliable as possible, use a server that is close to your own geographical location.

The Fedora Project maintains a list of Web and FTP public mirrors, sorted by region, at *http://fedoraproject.org/wiki/Mirrors*. To determine the complete directory path for the installation files, add

/11/Fedora/*architecture*/os/ to the path shown on the web page. A correct mirror location for an `i386` system resembles the URL **http://mirror.example.com/pub/fedora/linux/ releases/11/Fedora/i386/os**.

- If you are installing via NFS, proceed to *Section 7.8, "Installing via NFS"*.

- If you are installing via Web or FTP, proceed to *Section 7.9, "Installing via FTP or HTTP"*.

7.8. Installing via NFS

The NFS dialog applies only if you are installing from an NFS server (if you selected **NFS Image** in the **Installation Method** dialog).

Enter the domain name or IP address of your NFS server. For example, if you are installing from a host named **eastcoast** in the domain **example.com**, enter **eastcoast.example.com** in the **NFS Server** field.

Next, enter the name of the exported directory. If you followed the setup described in *Section 3.5, "Preparing for a Network Installation"*, you would enter the directory **/export/directory/**.

If the NFS server is exporting a mirror of the Fedora installation tree, enter the directory which contains the root of the installation tree. You will enter an Installation Key later on in the process which will determine which subdirectories are used to install from. If everything was specified properly, a message appears indicating that the installation program for Fedora is running.
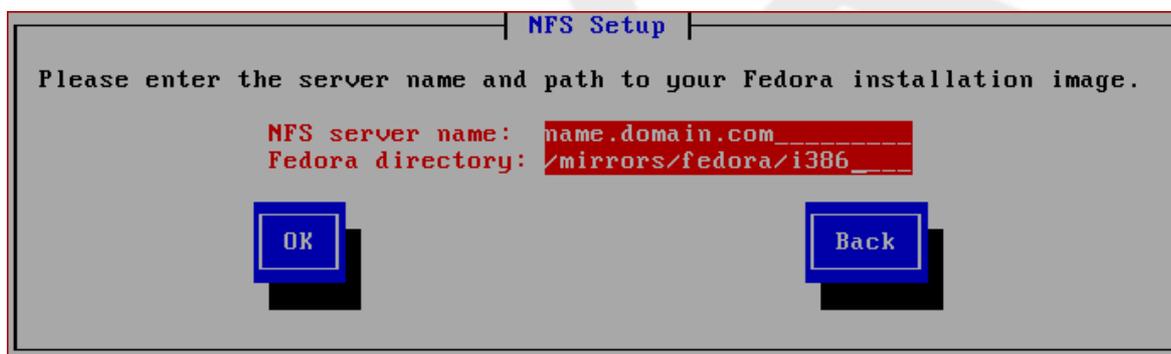
Figure 7.5. NFS Setup Dialog

If the NFS server is exporting the ISO images of the Fedora CD-ROMs, enter the directory which contains the ISO images.

Next, the **Welcome** dialog appears.

7.9. Installing via FTP or HTTP

The URL dialog applies only if you are installing from a FTP or HTTP server (if you selected **URL** in the **Installation Method** dialog). This dialog prompts you for information about the FTP or HTTP server from which you are installing Fedora.

Enter the name or IP address of the FTP or HTTP site you are installing from, and the name of the directory containing your architecture. For example, if the FTP or HTTP site contains the directory **/mirrors/Fedora/*arch*/**, enter **/mirrors/Fedora/*arch*/** (where *arch* is replaced with the

architecture type of your system, such as i386). If everything was specified properly, a message box appears indicating that files are being retrieved from the server.
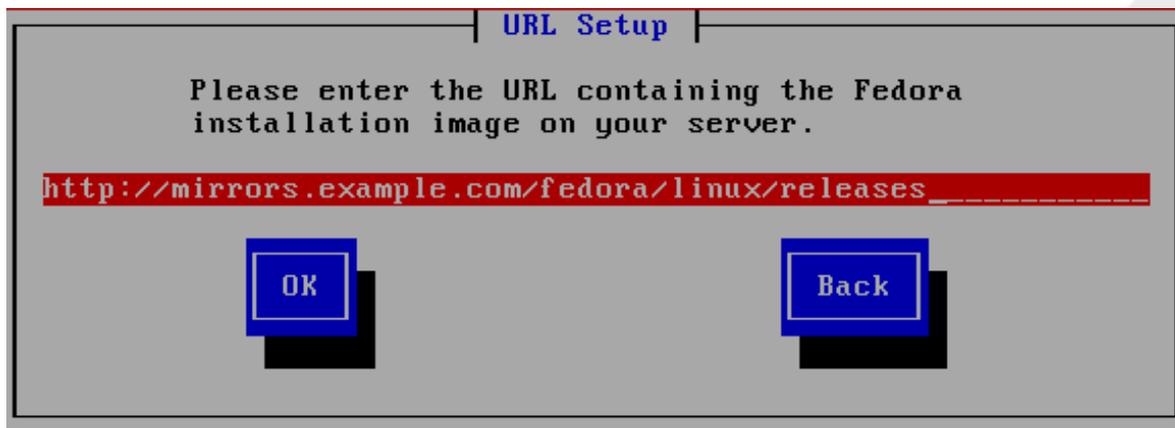
Figure 7.6. URL Setup Dialog

Next, the **Welcome** dialog appears.

> **Note**
>
> You can save disk space by using the ISO images you have already copied to the server. To accomplish this, install Fedora using ISO images without copying them into a single tree by loopback mounting them. For each ISO image:
>
> **mkdir disc*X***
>
> **mount -o loop Fedora11-discX.iso disc*X***
>
> Replace **x** with the corresponding disc number.

7.10. Welcome to Fedora

The **Welcome** screen does not prompt you for any input.
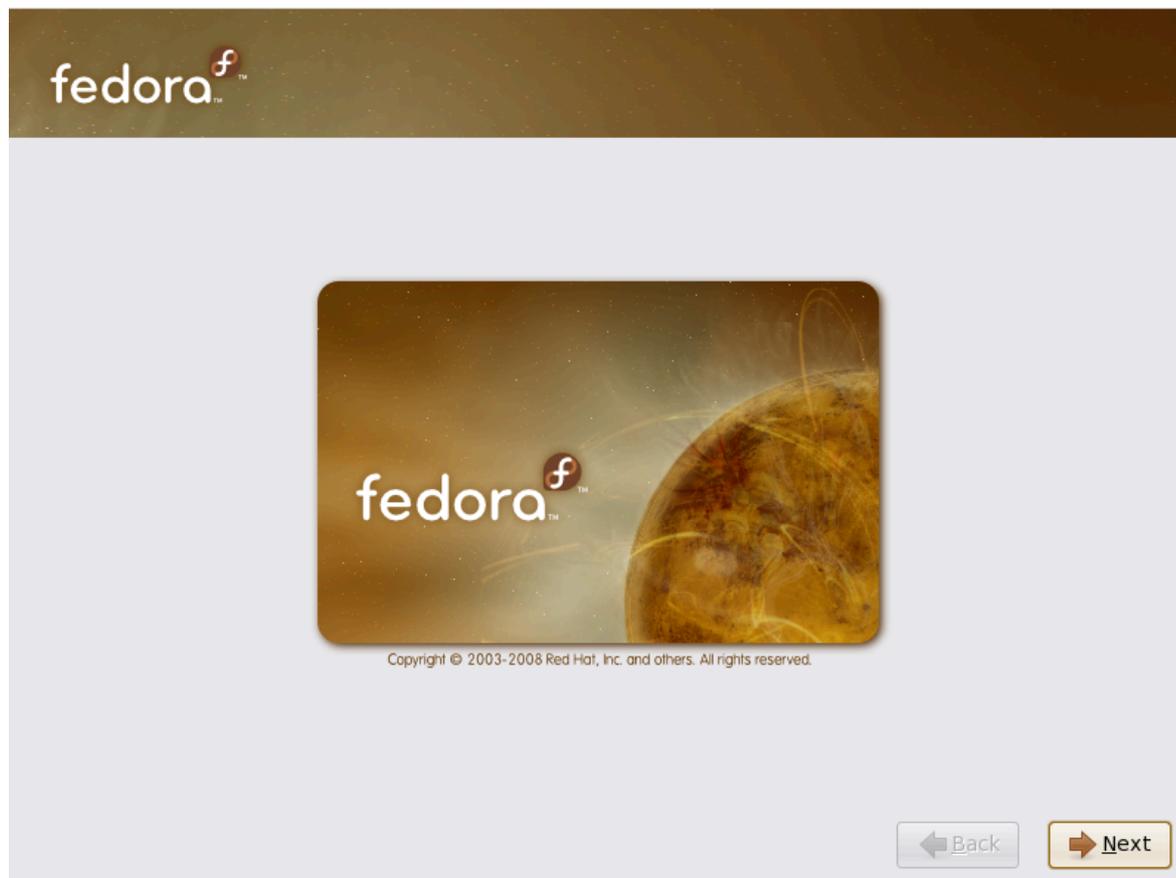
Click on the **Next** button to continue.

7.11. Language Selection

Using your mouse, select a language to use for the installation (refer to *Figure 7.7, "Language Selection"*).

The language you select here will become the default language for the operating system once it is installed. Selecting the appropriate language also helps target your time zone configuration later in the installation. The installation program tries to define the appropriate time zone based on what you specify on this screen.

To add support for additional languages, customize the installation at the package selection stage. For more information, refer to *Section 7.23.2.2, "Additional Language Support"*.
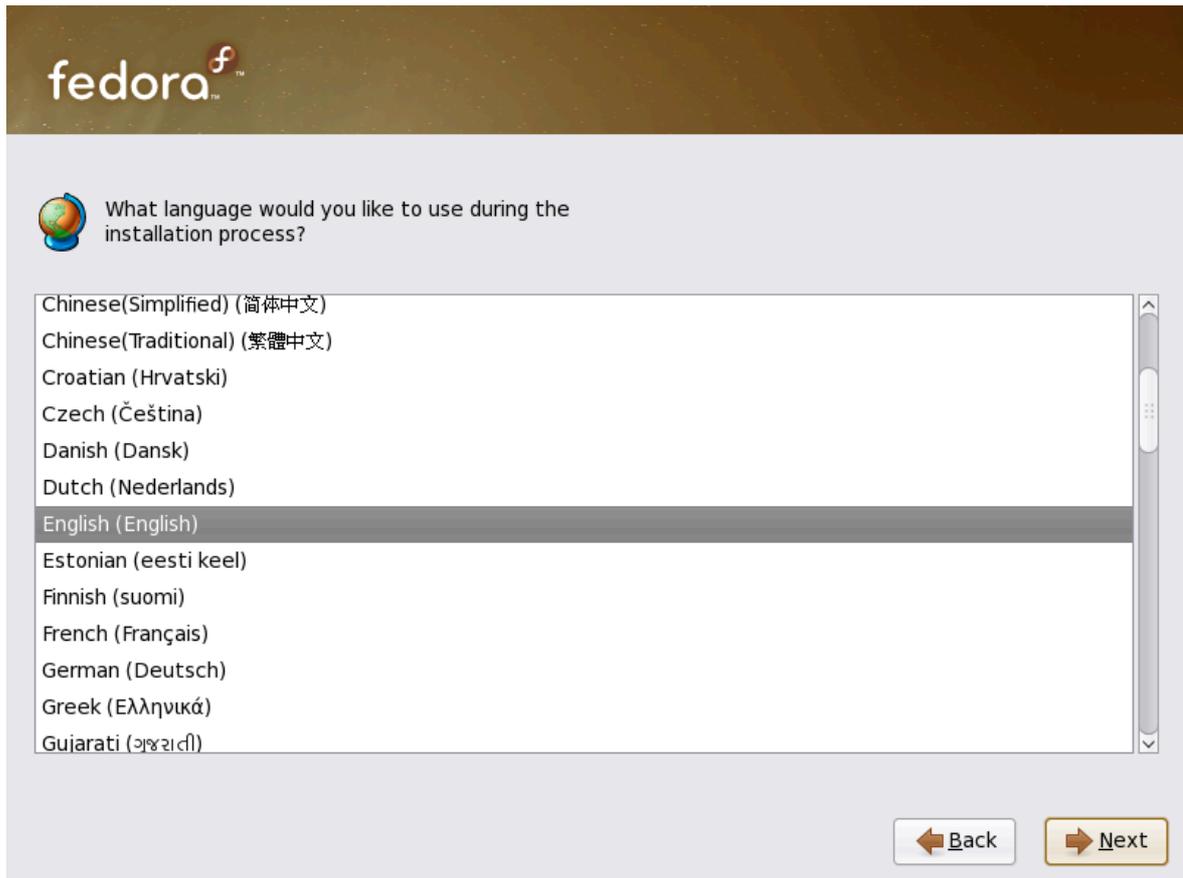
Figure 7.7. Language Selection

Once you select the appropriate language, click **Next** to continue.

7.12. Keyboard Configuration

Using your mouse, select the correct layout type (for example, U.S. English) for the keyboard you would prefer to use for the installation and as the system default (refer to the figure below).

Once you have made your selection, click **Next** to continue.
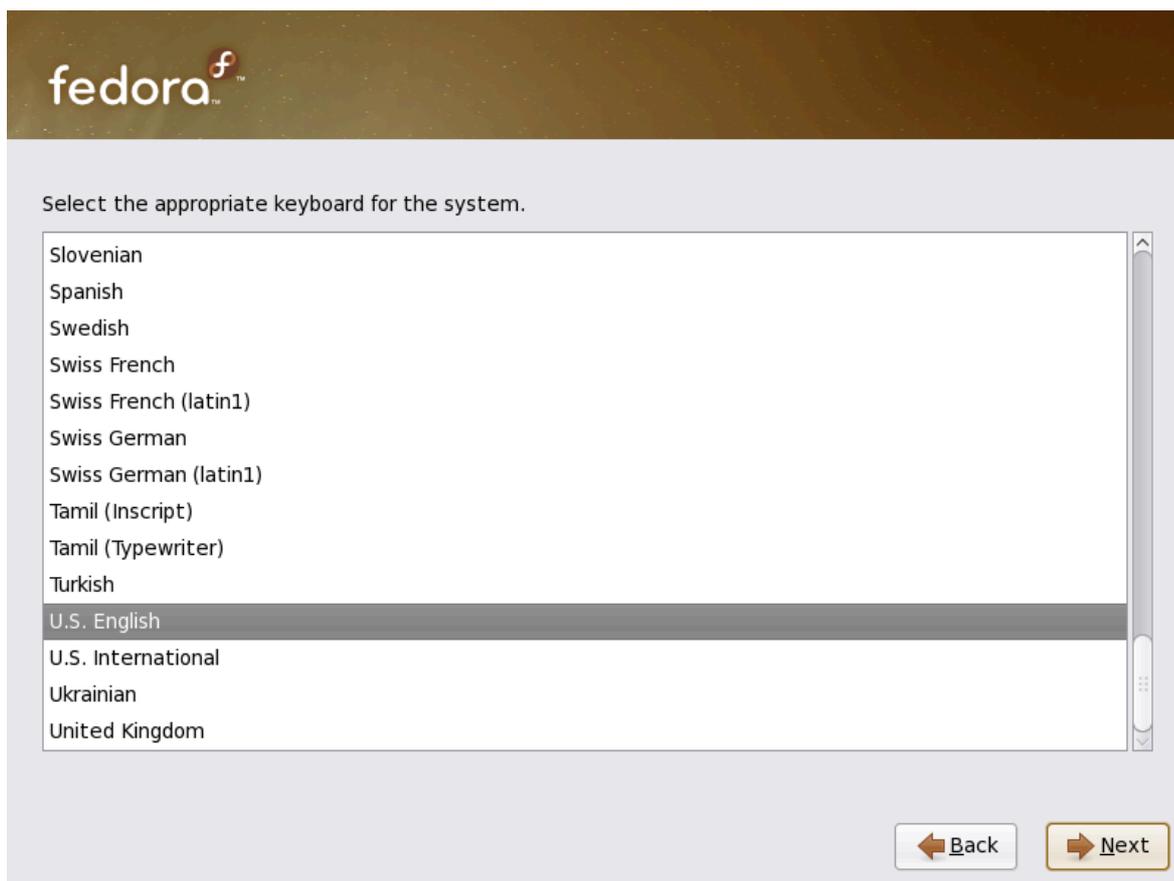
Figure 7.8. Keyboard Configuration

> **Note**
>
> To change your keyboard layout type after you have completed the installation, use the **Keyboard Configuration Tool**.
>
> Type the `system-config-keyboard` command in a shell prompt to launch the **Keyboard Configuration Tool**. If you are not root, it prompts you for the root password to continue.

7.13. Initializing the Hard Disk

If no readable partition tables are found on existing hard disks, the installation program asks to initialize the hard disk. This operation makes any existing data on the hard disk unreadable. If your system has a brand new hard disk with no operating system installed, or you have removed all partitions on the hard disk, answer **Yes**.
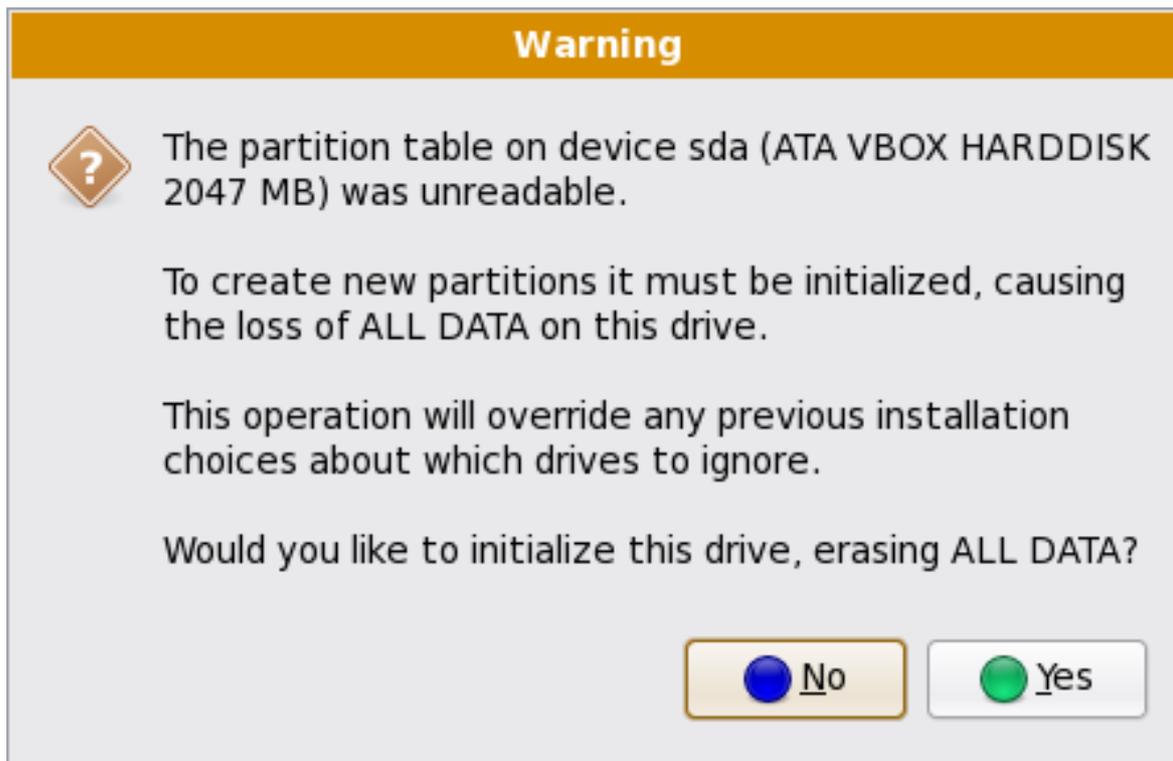
Figure 7.9. Warning screen – initializing hard drive

Certain RAID systems or other nonstandard configurations may be unreadable to the installation program and the prompt to initialize the hard disk may appear. The installation program responds to the physical disk structures it is able to detect.

> ## Detach Unneeded Disks
> If you have a nonstandard disk configuration that can be detached during installation and detected and configured afterward, power off the system, detach it, and restart the installation.

7.14. Upgrading an Existing System

The installation system automatically detects any existing installation of Fedora. The upgrade process updates the existing system software with new versions, but does not remove any data from users' home directories. The existing partition structure on your hard drives does not change. Your system configuration changes only if a package upgrade demands it. Most package upgrades do not change system configuration, but rather install an additional configuration file for you to examine later.

7.14.1. Upgrade Examine

If your system contains a Fedora or Red Hat Linux installation, a dialog appears asking whether you want to upgrade that installation. To perform an upgrade of an existing system, choose the appropriate installation from the drop-down list and select **Next**.
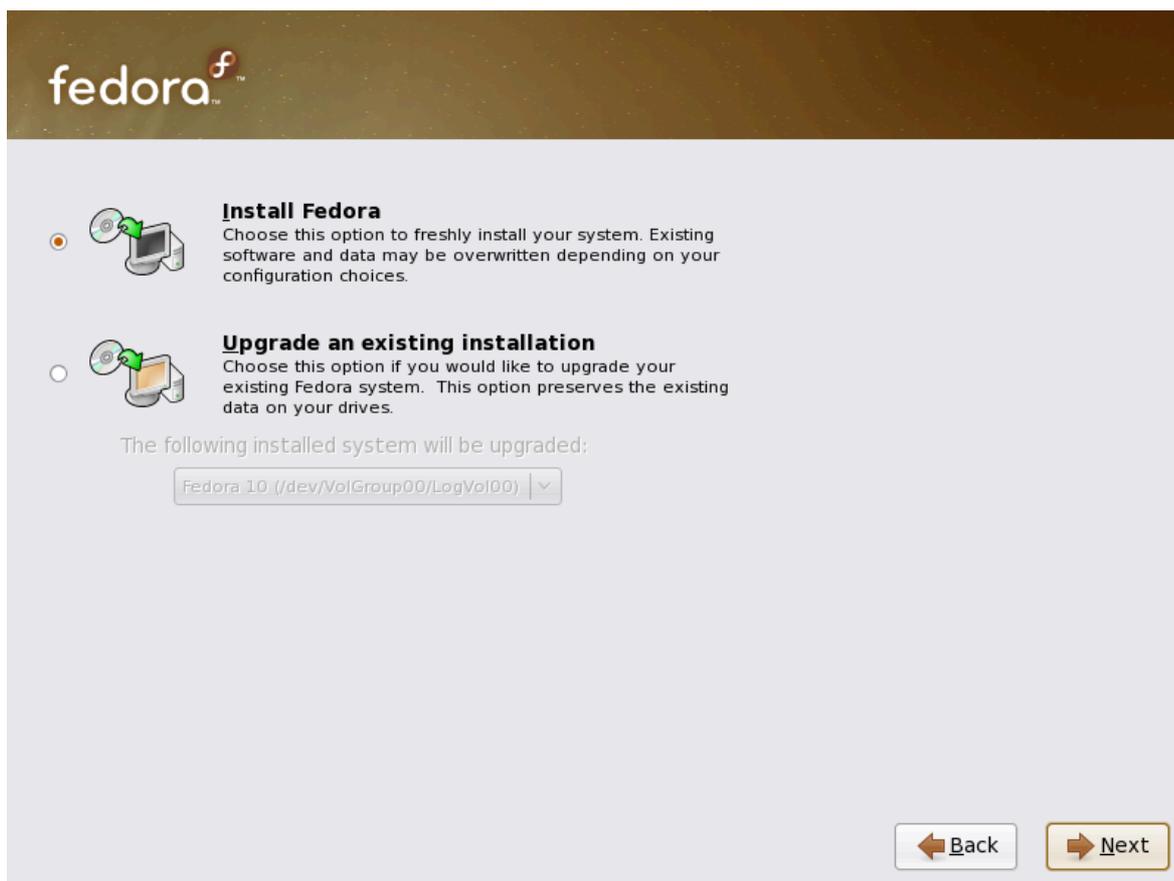
Figure 7.10. The upgrade screen

> **Manually Installed Software**
>
> Software you have installed manually on your existing Fedora or Red Hat Linux system may behave differently after an upgrade. You may need to manually reinstall or recompile this software after an upgrade to ensure it performs correctly on the updated system.

7.14.2. Upgrading Using the Installer

> **Installations are Recommended**
>
> In general, the Fedora Project recommends that you keep user data on a separate `/home` partition and perform a fresh installation. For more information on partitions and how to set them up, refer to *Section 7.18, "Disk Partitioning Setup"*.

If you choose to upgrade your system using the installation program, any software not provided by Fedora that conflicts with Fedora software is overwritten. Before you begin an upgrade this way, make a list of your system's current packages for later reference:

```
rpm -qa --qf '%{NAME} %{VERSION}-%{RELEASE} %{ARCH}\n' > ~/old-
pkglist.txt
```

After installation, consult this list to discover which packages you may need to rebuild or retrieve from non-Fedora software repositories.

Next, make a backup of any system configuration data:

```
su -c 'tar czf /tmp/etc-`date +%F`.tar.gz /etc' su -c 'mv /tmp/etc-
*.tar.gz /home'
```

Make a complete backup of any important data before performing an upgrade. Important data may include the contents of your entire **/home** directory as well as content from services such as an Apache, FTP, or SQL server, or a source code management system. Although upgrades are not destructive, if you perform one improperly there is a small possibility of data loss.

> ⚠ **Storing Backups**
>
> Note that the above examples store backup materials in a **/home** directory. If your **/home** directory is not a separate partition, *you should not follow these examples verbatim!* Store your backups on another device such as CD or DVD discs or an external hard disk.

For more information on completing the upgrade process later, refer to *Section 16.2, "Finishing an Upgrade"*.

7.14.3. Upgrading Boot Loader Configuration

Your completed Fedora installation must be registered in the *boot loader* to boot properly. A boot loader is software on your machine that locates and starts the operating system. Refer to *Section 7.22, "x86, AMD64, and Intel 64 Boot Loader Configuration"* for more information about boot loaders.

If the existing boot loader was installed by a Linux distribution, the installation system can modify it to load the new Fedora system. To update the existing Linux boot loader, select **Update boot loader configuration**. This is the default behavior when you upgrade an existing Fedora or Red Hat Linux installation.

GRUB is the standard boot loader for Fedora. If your machine uses another boot loader, such as BootMagic™, System Commander™, or the loader installed by Microsoft Windows, then the Fedora installation system cannot update it. In this case, select **Skip boot loader updating**. When the installation process completes, refer to the documentation for your product for assistance.

Install a new boot loader as part of an upgrade process only if you are certain you want to replace the existing boot loader. If you install a new boot loader, you may not be able to boot other operating systems on the same machine until you have configured the new boot loader. Select **Create new boot loader configuration** to remove the existing boot loader and install GRUB.

After you make your selection, click **Next** to continue.

7.15. Network Configuration

Fedora contains support for both IPv4 and IPv6. However, by default, the installation program configures network interfaces on your computer for IPv4, and to use DHCP via **NetworkManager**.

Currently **NetworkManager** does not support IPv6. If your network only supports IPv6 you should use **system-config-network** after installation to configure your network interfaces.

Setup prompts you to supply a host name and domain name for this computer, in the format `hostname.domainname`. Many networks have a DHCP (Dynamic Host Configuration Protocol) service that automatically supplies connected systems with a domain name, leaving the user to enter a hostname.
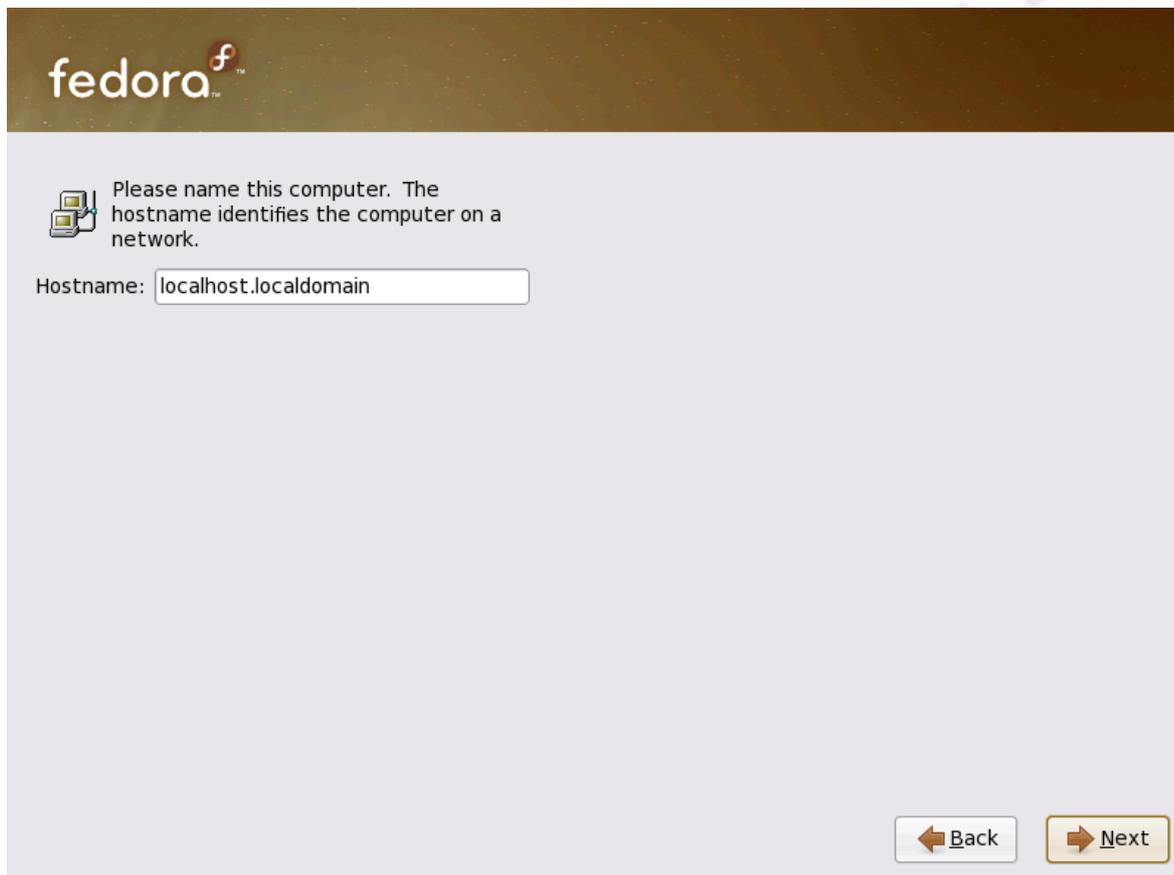
Figure 7.11. Setting the hostname

To set up a network that is behind an Internet firewall or router, you may want to use `hostname.localdomain` for your Fedora system. If you have more than one computer on this network, you should give each one a separate host name in this domain.

> **Valid Hostnames**
>
> You may give your system any name provided that the full hostname is unique. The hostname may include letters, numbers and hyphens.

On some networks, the DHCP provider also provides the name of the computer, or hostname. The complete hostname includes both the name of the machine and the name of the domain of which it is a member, such as `machine1.example.com`. The machine name (or "short hostname") is `machine1`, and the domain name is `example.com`.

If your Fedora system is connected *directly* to the Internet, you must pay attention to additional considerations to avoid service interruptions or risk action by your upstream service provider. A full discussion of these issues is beyond the scope of this document.

> **Modem Configuration**
>
> The installation program does not configure modems. Configure these devices after installation with the **Network** utility. The settings for your modem are specific to your particular Internet Service Provider (ISP).

7.15.1. Manual configuration

Installations that require certain advanced configurations cannot succeed without network connectivity during the installation process, for example, installations on systems with ISCSI drives. In situations where successful installation depends upon correct network settings, the installation program will present you with a dialog that allows you to confirm these details.

Figure 7.12. Manual network configuration

If your network does not have DHCP enabled, or if you need to override the DHCP settings, select the network interface that you plan to use from the **Interfaces** menu. Clear the checkbox for **Use dynamic IP configuration (DHCP)**. You can now enter an IPv4 address and netmask for this system in the form *address* / *netmask*, along with the gateway address and nameserver address for your network.

Click **OK** to accept these settings and continue.

7.16. Time Zone Configuration

Specify a time zone even if you plan to use NTP (Network Time Protocol) to maintain the accuracy of the system clock.

Set your time zone by selecting the city closest to your computer's physical location. Click on the map to zoom in to a particular geographical region of the world.

Specify a time zone even if you plan to use NTP (Network Time Protocol) to maintain the accuracy of the system clock.

From here there are two ways for you to select your time zone:

* Using your mouse, click on the interactive map to select a specific city (represented by a yellow dot). A red **X** appears indicating your selection.

* You can also scroll through the list at the bottom of the screen to select your time zone. Using your mouse, click on a location to highlight your selection.
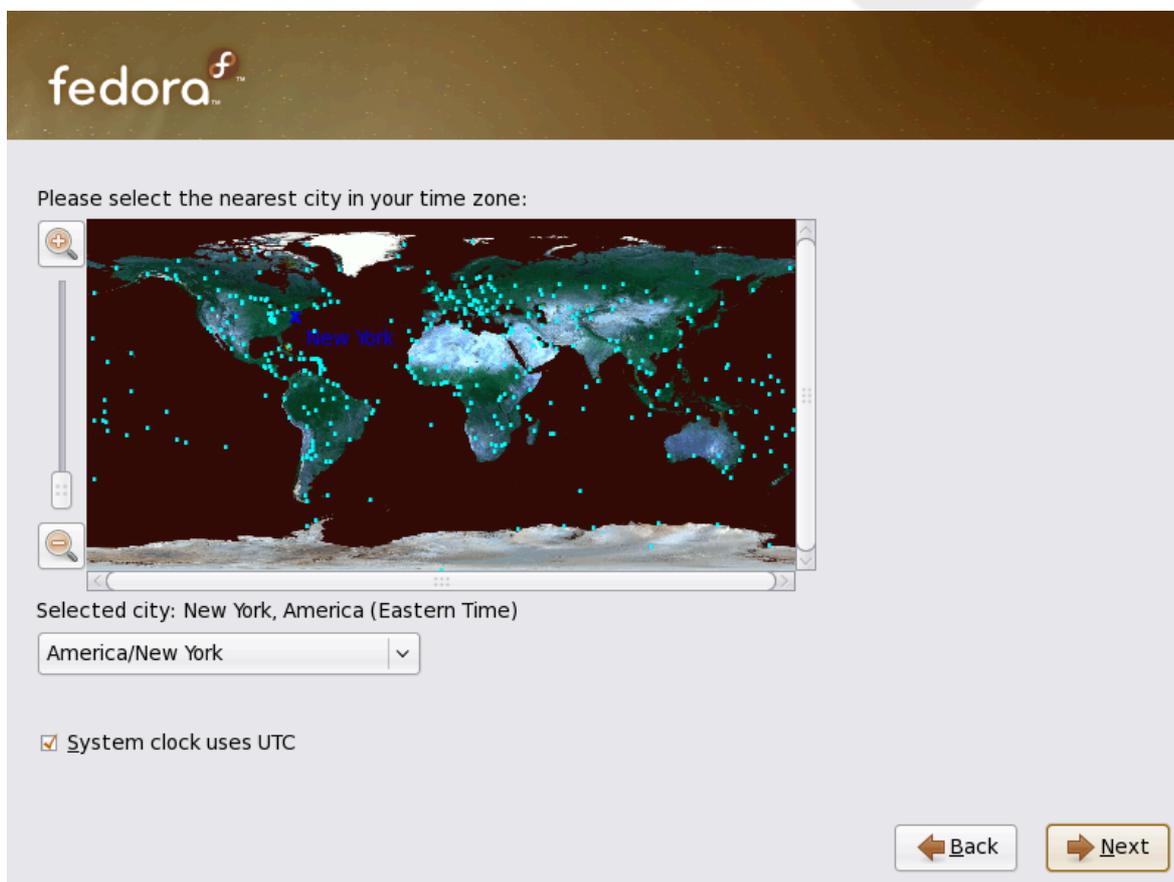
Figure 7.13. Configuring the Time Zone

If Fedora is the only operating system on your computer, select **System clock uses UTC**. The system clock is a piece of hardware on your computer system. Fedora uses the timezone setting to determine the offset between the local time and UTC on the system clock. This behavior is standard for UNIX-like operating systems.

> ⚠ **Windows and the System Clock**
>
> Do not enable the **System clock uses UTC** option if your machine also runs Microsoft Windows. Microsoft operating systems change the BIOS clock to match local time rather than UTC. This may cause unexpected behavior under Fedora.

> 📝 **Note**
>
> To change your time zone configuration after you have completed the installation, use the **Time and Date Properties Tool**.
>
> Type the `system-config-date` command in a shell prompt to launch the **Time and Date Properties Tool**. If you are not root, it prompts you for the root password to continue.
>
> To run the **Time and Date Properties Tool** as a text-based application, use the command `timeconfig`.

Select **Next** to proceed.

7.17. Set the Root Password

Setting up a root account and password is one of the most important steps during your installation. Your root account is similar to the administrator account used on Microsoft Windows machines. The root account is used to install packages, upgrade RPMs, and perform most system maintenance. Logging in as root gives you complete control over your system.

> 📝 **Note**
>
> The root user (also known as the superuser) has complete access to the entire system; for this reason, logging in as the root user is best done *only* to perform system maintenance or administration.
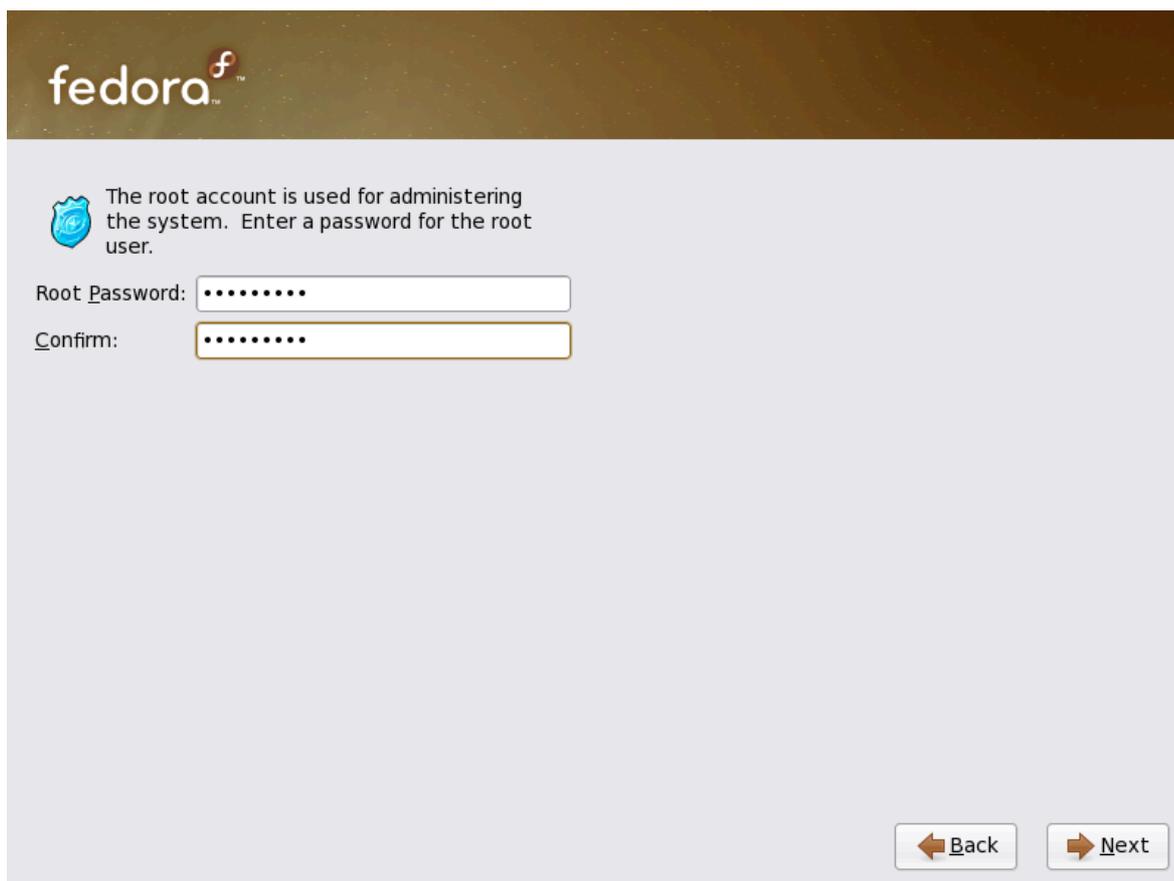
Figure 7.14. Root Password

Use the root account only for system administration. Create a non-root account for your general use and `su -` to root when you need to fix something quickly. These basic rules minimize the chances of a typo or an incorrect command doing damage to your system.

> **Note**
>
> To become root, type `su -` at the shell prompt in a terminal window and then press `Enter`. Then, enter the root password and press `Enter`.

The installation program prompts you to set a root password[1] for your system. *You cannot proceed to the next stage of the installation process without entering a root password.*

The root password must be at least six characters long; the password you type is not echoed to the screen. You must enter the password twice; if the two passwords do not match, the installation program asks you to enter them again.

You should make the root password something you can remember, but not something that is easy for someone else to guess. Your name, your phone number, *qwerty*, *password, root*, *123456*, and *anteater* are all examples of bad passwords. Good passwords mix numerals with upper and lower case letters and do not contain dictionary words: *Aard387vark* or *420BMttNT*, for example. Remember

---

[1] A root password is the administrative password for your Fedora system. You should only log in as root when needed for system maintenance. The root account does not operate within the restrictions placed on normal user accounts, so changes made as root can have implications for your entire system.

that the password is case-sensitive. If you write down your password, keep it in a secure place. However, it is recommended that you do not write down this or any password you create.

> **Note**
>
> Do not use one of the example passwords offered in this manual. Using one of these passwords could be considered a security risk.

> **Note**
>
> To change your root password after you have completed the installation, use the **Root Password Tool**.
>
> Type the **system-config-rootpassword** command in a shell prompt to launch the **Root Password Tool**. If you are not root, it prompts you for the root password to continue.

Enter the `root` password into the **Root Password** field. Fedora displays the characters as asterisks for security. Type the same password into the **Confirm** field to ensure it is set correctly. After you set the root password, select **Next** to proceed.

7.18. Disk Partitioning Setup

Partitioning allows you to divide your hard drive into isolated sections, where each section behaves as its own hard drive. Partitioning is particularly useful if you run multiple operating systems. If you are not sure how you want your system to be partitioned, read *Appendix A, An Introduction to Disk Partitions* for more information.

On this screen you can choose to create the default layout or choose to manual partition using the 'Create custom layout' option of **Disk Druid**.

The first three options allow you to perform an automated installation without having to partition your drive(s) yourself. If you do not feel comfortable with partitioning your system, it is recommended that you *do not* choose to create a custom layout and instead let the installation program partition for you.

You can configure an iSCSI target for installation, or disable a dmraid device from this screen by clicking on the 'Advanced storage configuration' button. For more information refer to *Section 7.19, " Advanced Storage Options "*.

> **Warning**
>
> The **Update Agent** downloads updated packages to `/var/cache/yum/` by default. If you partition the system manually, and create a separate `/var/` partition, be sure to create the partition large enough (3.0 GB or more) to download package updates.
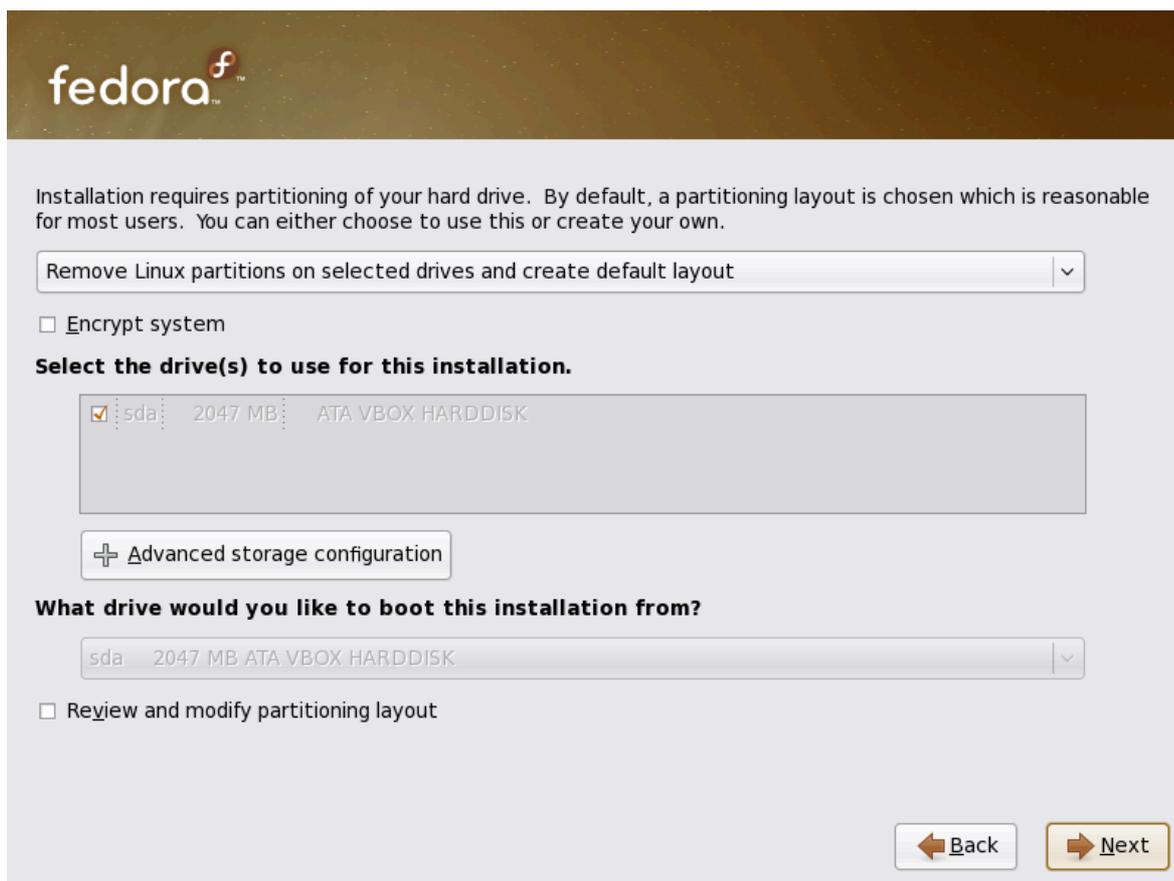
Figure 7.15. Disk Partitioning Setup

If you choose to create a custom layout using **Disk Druid**, refer to *Section 7.21, "Partitioning Your System"*.

> ⚠️ **Warning**
>
> If you receive an error after the **Disk Partitioning Setup** phase of the installation saying something similar to:
>
> "The partition table on device hda was unreadable. To create new partitions it must be initialized, causing the loss of ALL DATA on this drive."
>
> you may not have a partition table on that drive or the partition table on the drive may not be recognizable by the partitioning software used in the installation program.
>
> Users who have used programs such as **EZ-BIOS** have experienced similar problems, causing data to be lost (assuming the data was not backed up before the installation began).
>
> No matter what type of installation you are performing, backups of the existing data on your systems should always be made.

7.18.1. RAID and Other Disk Devices

7.18.1.1. Hardware RAID

RAID, or Redundant Array of Independent Disks, allows a group, or array, of drives to act as a single device. Configure any RAID functions provided by the mainboard of your computer, or attached controller cards, before you begin the installation process. Each active RAID array appears as one drive within Fedora.

On systems with more than one hard drive you may configure Fedora to operate several of the drives as a Linux RAID array without requiring any additional hardware.

7.18.1.2. Software RAID

You can use the Fedora installation program to create Linux software RAID arrays, where RAID functions are controlled by the operating system rather than dedicated hardware. These functions are explained in detail in *Section 7.21, "Partitioning Your System"*.

7.18.1.3. FireWire and USB Disks

Some FireWire and USB hard disks may not be recognized by the Fedora installation system. If configuration of these disks at installation time is not vital, disconnect them to avoid any confusion.

> **Post-installation Usage**
>
> You can connect and configure external FireWire and USB hard disks after installation. Most such devices are recognized by the kernel and available for use at that time.

7.19. Advanced Storage Options
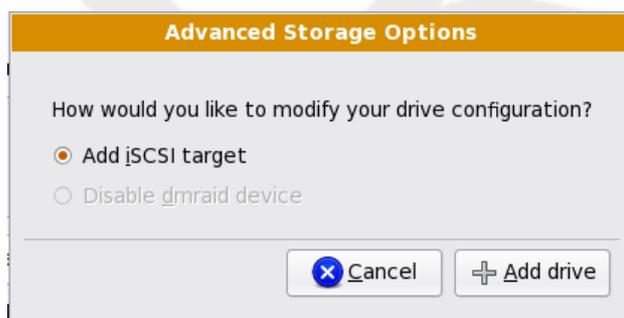
Figure 7.16. Advanced Storage Options

From this screen you can choose to disable a dmraid device, in which case the individual elements of the dmraid device will appear as separate hard drives. You can also choose to configure an iSCSI (SCSI over TCP/IP) target. See *Appendix B, ISCSI disks* for an introduction to iSCSI.

To configure an ISCSI target invoke the 'Configure ISCSI Parameters' dialog by selecting 'Add ISCSI target' and clicking on the 'Add Drive' button. Fill in the details for the ISCSI target IP and provide a unique ISCSI initiator name to identify this system. If the ISCSI target uses CHAP (Challenge

Handshake Authentication Protocol) for authentication, enter the CHAP username and password. If your enviroment uses 2-way CHAP (also called "Mutual CHAP"), also enter the reverse CHAP username and password. Click the 'Add target' button to attempt connection to the ISCSI target using this information.
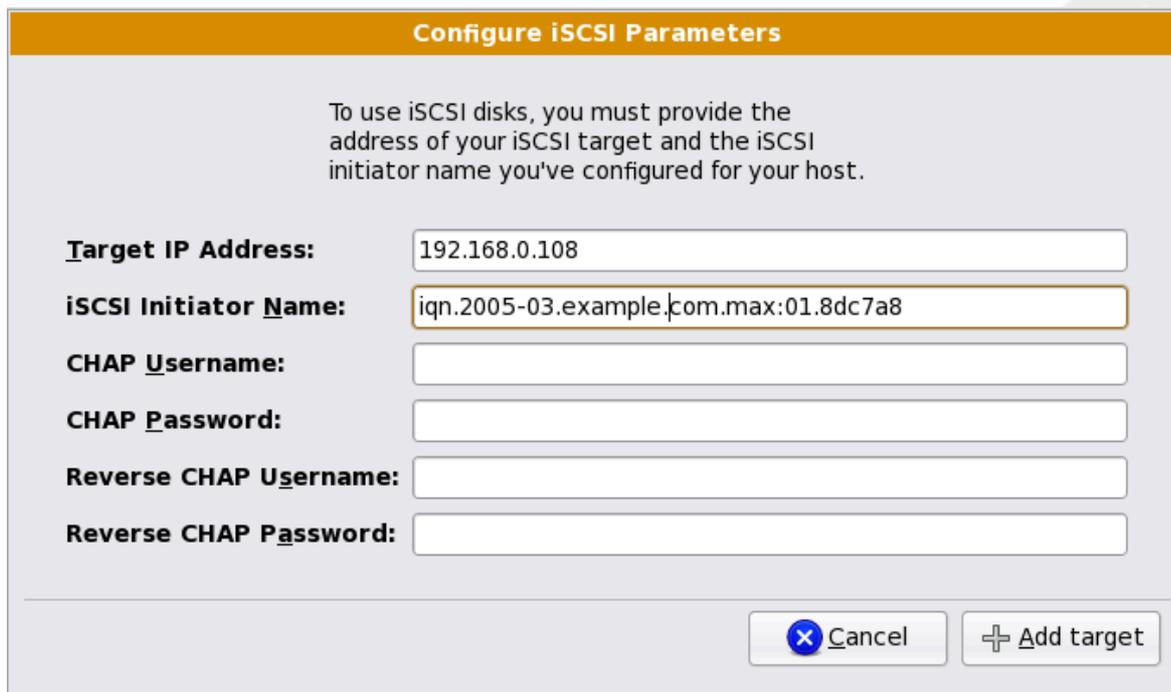
Figure 7.17. Configure ISCSI Parameters

Please note that you will be able to reattempt with a different ISCSI target IP should you enter it incorrectly, but in order to change the ISCSI initiator name you will need to restart the installation.

7.20. Create Default Layout

Create default layout allows you to have some control concerning what data is removed (if any) from your system. Your options are:

- **Remove all partitions on selected drives and create default layout** — select this option to remove all partitions on your hard drive(s) (this includes partitions created by other operating systems such as Windows VFAT or NTFS partitions).

> ⚠ **Warning**
>
> If you select this option, all data on the selected hard drive(s) is removed by the installation program. Do not select this option if you have information that you want to keep on the hard drive(s) where you are installing Fedora.

- **Remove Linux partitions on selected drives and create default layout** — select this option to remove only Linux partitions (partitions created from a previous Linux installation). This does not remove other partitions you may have on your hard drive(s) (such as VFAT or FAT32 partitions).

- **Use free space on selected drives and create default layout** — select this option to retain your current data and partitions, assuming you have enough free space available on your hard drive(s).
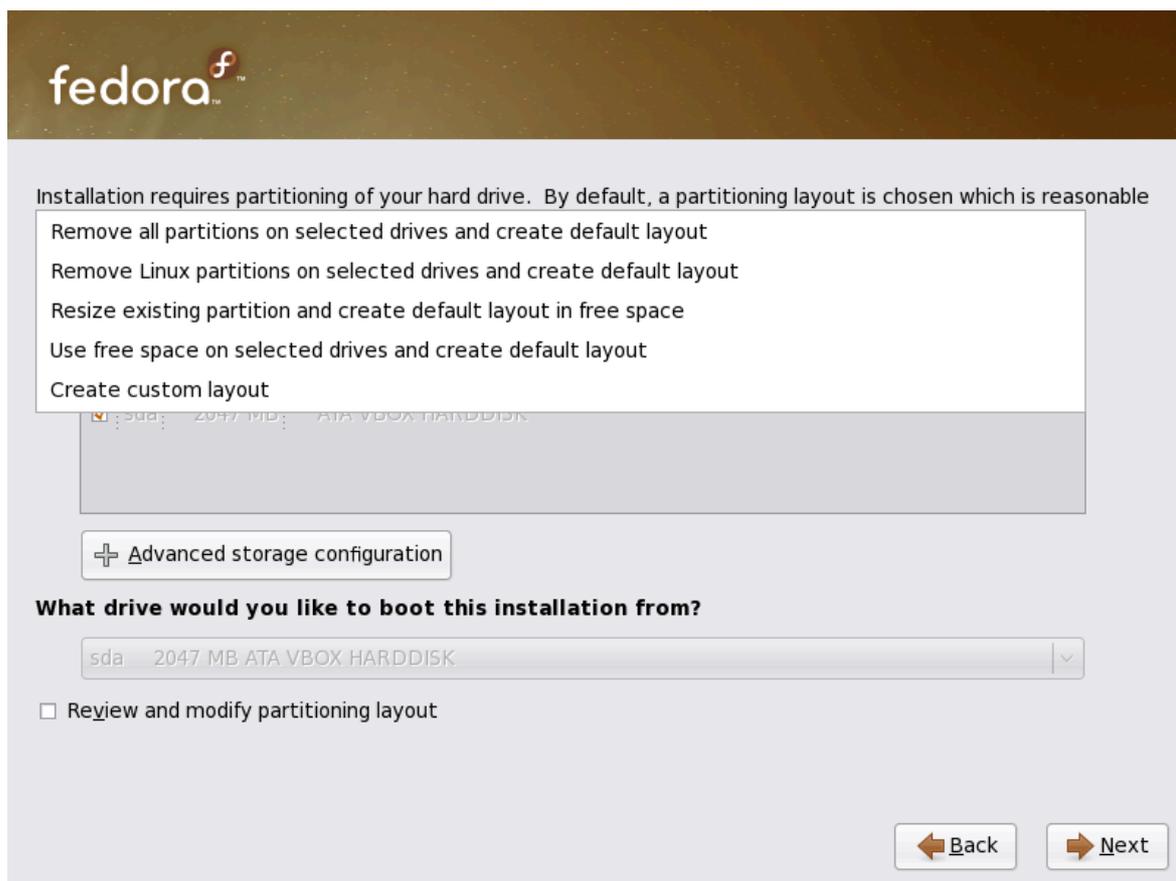
Figure 7.18. Create Default Layout

Using your mouse, choose the storage drive(s) on which you want Fedora to be installed. If you have two or more drives, you can choose which drive(s) should contain this installation. Unselected drives, and any data on them, are not touched.

> **Warning**
>
> It is always a good idea to back up any data that you have on your systems. For example, if you are upgrading or creating a dual-boot system, you should back up any data you wish to keep on your drive(s). Mistakes do happen and can result in the loss of all your data.

> **Note**
>
> If you have a RAID card, be aware that some BIOSes do not support booting from the RAID card. In cases such as these, the `/boot/` partition must be created on a partition outside of the RAID array, such as on a separate hard drive. An internal hard drive is necessary to use for partition creation with problematic RAID cards.
>
> A `/boot/` partition is also necessary for software RAID setups.
>
> If you have chosen to automatically partition your system, you should select **Review** and manually edit your `/boot/` partition.

Select **Encrypt system** to encrypt all partitions except the /boot partition.

Use the **Advanced storage options** option if:

- You want to install Fedora to a drive connected through the *iSCSI* protocol. Select **Advanced storage options**, then select **Add iSCSI target**, then select **Add drive**. Provide an IP address and the iSCSI initiator name, and select **Add drive**.

- You want to disable a *dmraid* device that was detected at boot time.

To review and make any necessary changes to the partitions created by automatic partitioning, select the **Review** option. After selecting **Review** and clicking **Next** to move forward, the partitions created for you in **Disk Druid** appear. You can make modifications to these partitions if they do not meet your needs.

> ### Installing in text mode
>
> If you install Fedora in text mode, you can only use the default partitioning schemes described in this section. Therefore, although you can choose to use a whole drive, to remove existing Linux partitions, or to use the free space on the drive, you cannot customize the partition layout. That is, you cannot add or remove partitions or file systems beyond those that the installer automatically adds or removes. If you require a customized layout at installation time, you should perform a graphical installation over a VNC connection or a kickstart installation.
>
> Furthermore, advanced options such as LVM, encrypted filesystems, and resizable filesystems are available only in graphical mode and kickstart.

Click **Next** once you have made your selections to proceed.

7.21. Partitioning Your System

If you chose one of the three automatic partitioning options and did not select **Review**, skip ahead to *Section 7.23, "Package Group Selection"*.

If you chose one of the automatic partitioning options and selected **Review**, you can either accept the current partition settings (click **Next**), or modify the setup using **Disk Druid**, the manual partitioning tool.

> ### Note
>
> Please note that in the text mode installation it is not possible to work with LVM (Logical Volumes) beyond viewing the existing setup. LVM can only be set up using the graphical Disk Druid program in a graphical installation.

If you chose to create a custom layout, you must tell the installation program where to install Fedora. This is done by defining mount points for one or more disk partitions in which Fedora is installed. You may also need to create and/or delete partitions at this time.

> ### Note
>
> If you have not yet planned how to set up your partitions, refer to *Appendix A, An Introduction to Disk Partitions* and *Section 7.21.4, "Recommended Partitioning*

Scheme". At a bare minimum, you need an appropriately-sized root partition, and a swap partition equal to twice the amount of RAM you have on the system. Itanium system users should have a `/boot/efi/` partition of approximately 100 MB and of type FAT (VFAT), a swap partition of at least 512 MB, and an appropriately-sized root (`/`) partition.
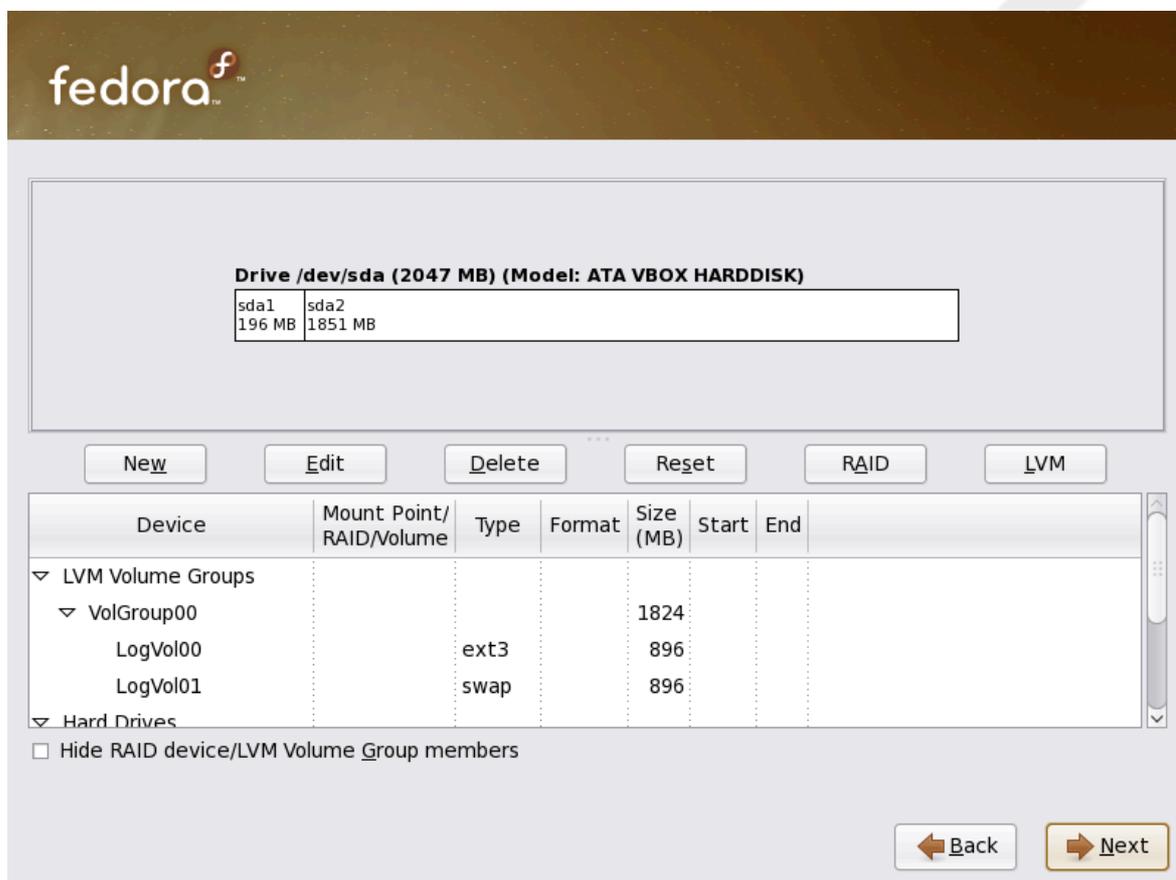
Figure 7.19. Partitioning with **Disk Druid** on x86, AMD64, and Intel® 64 Systems

The partitioning tool used by the installation program is **Disk Druid**. With the exception of certain esoteric situations, **Disk Druid** can handle the partitioning requirements for a typical installation.

7.21.1. Graphical Display of Hard Drive(s)

Disk Druid offers a graphical representation of your hard drive(s).

Using your mouse, click once to highlight a particular field in the graphical display. Double-click to edit an existing partition or to create a partition out of existing free space.

Above the display, you can review the **Drive** name (such as /dev/hda), the **Geom** (which shows the hard disk's geometry and consists of three numbers representing the number of cylinders, heads, and sectors as reported by the hard disk), and the **Model** of the hard drive as detected by the installation program.

7.21.2. Disk Druid's Buttons

These buttons control **Disk Druid's** actions. They are used to change the attributes of a partition (for example the file system type and mount point) and also to create RAID devices. Buttons on this screen are also used to accept the changes you have made, or to exit **Disk Druid**. For further explanation, take a look at each button in order:

- **New**: Select this option to add a partition or LVM physical volume to the disk. In the **Add partition** dialog, choose a mount point and a partition type. If you have more than one disk on the system, choose which disks the partition may inhabit. Indicate a size in megabytes for the partition. If you wish to encrypt the partition, select that option.

> ⭐ **Illegal Partitions**
>
> The `/bin/`, `/dev/`, `/etc/`, `/lib/`, `/proc/`, `/root/`, and `/sbin/` directories may not be used for separate partitions in **Disk Druid**. These directories reside on the `/` (root) partition.
>
> The `/boot` partition may not reside on an LVM volume group. Create the `/boot` partition before configuring any volume groups.

You may also choose from three options for sizing your partition:

Fixed size
> Use a fixed size as close to your entry as possible.

Fill all space up to
> Grow the partition to a maximum size of your choice.

Fill to maximum allowable size
> Grow the partition until it fills the remainder of the selected disks.

> 📌 **Partition Sizes**
>
> The actual partition on the disk may be slightly smaller or larger than your choice. Disk geometry issues cause this effect, not an error or bug.

Select the **Encrypt partition** option to encrypt all information on the disk partition.

After you enter the details for your partition, select **OK** to continue. If you chose to encrypt the partition, the installer prompts you to assign a passphrase by typing it twice. For hints on using good passphrases, refer to *Section 7.17, "Set the Root Password"*.

- **Edit**: Used to modify attributes of the partition currently selected in the **Partitions** section. Selecting **Edit** opens a dialog box. Some or all of the fields can be edited, depending on whether the partition information has already been written to disk.

You can also edit free space as represented in the graphical display to create a new partition within that space. Either highlight the free space and then select the **Edit** button, or double-click on the free space to edit it.

- To make a RAID device, you must first create (or reuse existing) software RAID partitions. Once you have created two or more software RAID partitions, select **Make RAID** to join the software RAID partitions into a RAID device.

- **Delete**: Used to remove the partition currently highlighted in the **Current Disk Partitions** section. You will be asked to confirm the deletion of any partition.

  To delete an LVM physical volume, first delete any volume groups of which that physical volume is a member.

  If you make a mistake, use the **Reset** option to abandon all the changes you have made.

- **Reset**: Used to restore **Disk Druid** to its original state. All changes made will be lost if you **Reset** the partitions.

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

  To make a RAID device, you must first create software RAID partitions. Once you have created two or more software RAID partitions, select **RAID** to join the software RAID partitions into a RAID device.

Figure 7.20. RAID options

Create a software RAID partition

  Choose this option to add a partition for software RAID. This option is the only choice available if your disk contains no software RAID partitions.

Figure 7.21. Create a software RAID partition

Create a RAID device

Choose this option to construct a RAID device from two or more existing software RAID partitions. This option is available if two or more software RAID partitions have been configured.

Figure 7.22. Create a RAID device

Clone a drive to create a RAID device

Choose this option to set up a RAID *mirror* of an existing disk. This option is available if two or more disks are attached to the system.

Figure 7.23. Clone a RAID device

- **LVM**: Allows you to create an LVM logical volume. The role of LVM (Logical Volume Manager) is to present a simple logical view of underlying physical storage space, such as a hard drive(s). LVM manages individual physical disks — or to be more precise, the individual partitions present on them. *It should only be used if you have experience using LVM.* Note, LVM is only available in the graphical installation program.

  To assign one or more physical volumes to a volume group, first name the volume group. Then select the physical volumes to be used in the volume group. Finally, configure logical volumes on any volume groups using the **Add**, **Edit** and **Delete** options.

  You may not remove a physical volume from a volume group if doing so would leave insufficient space for that group's logical volumes. Take for example a volume group made up of two 5 GB LVM physical volume partitions, which contains an 8 GB logical volume. The installer would not allow you to remove either of the component physical volumes, since that would leave only 5 GB in the group for an 8 GB logical volume. If you reduce the total size of any logical volumes appropriately, you may then remove a physical volume from the volume group. In the example, reducing the size of the logical volume to 4 GB would allow you to remove one of the 5 GB physical volumes.

> ⭐ **LVM Unavailable in Text Installs**
>
> LVM initial set up is not available in a text-mode installation. The installer allows you to edit pre-configured LVM volumes. If you need to create an LVM configuration from scratch, hit `Alt+F2` to use the terminal, and run the `lvm` command. To return to the text-mode installation, hit `Alt+F1`.

7.21.3. Partition Fields

Above the partition hierarchy are labels which present information about the partitions you are creating. The labels are defined as follows:

- **Device**: This field displays the partition's device name.

- **Mount Point/RAID/Volume**: A mount point is the location within the directory hierarchy at which a volume exists; the volume is "mounted" at this location. This field indicates where the partition is mounted. If a partition exists, but is not set, then you need to define its mount point. Double-click on the partition or click the **Edit** button.

- **Type**: This field shows the partition's file system type (for example, ext2, ext3, or vfat).

- **Format**: This field shows if the partition being created will be formatted.

- **Size (MB)**: This field shows the partition's size (in MB).

- **Start**: This field shows the cylinder on your hard drive where the partition begins.

- **End**: This field shows the cylinder on your hard drive where the partition ends.

Hide RAID device/LVM Volume Group members: Select this option if you do not want to view any RAID device or LVM Volume Group members that have been created.

7.21.4. Recommended Partitioning Scheme

7.21.4.1. x86, AMD64, and Intel® 64 systems

Unless you have a reason for doing otherwise, we recommend that you create the following partitions for x86, AMD64, and Intel® 64 systems:

- A `swap` partition

- A `/boot` partition

- A `/` partition

A swap partition (at least 256 MB)

Swap partitions are used to support virtual memory. In other words, data is written to a swap partition when there is not enough RAM to store the data your system is processing. In addition, certain power management features store all of the memory for a suspended system in the available swap partitions.

If you are unsure about what size swap partition to create, make it twice the amount of RAM on your machine. It must be of type swap.

Creation of the proper amount of swap space varies depending on a number of factors including the following (in descending order of importance):

- The applications running on the machine.

- The amount of physical RAM installed on the machine.

- The version of the OS.

Swap should equal 2x physical RAM for up to 2 GB of physical RAM, and then an additional 1x physical RAM for any amount above 2 GB, but never less than 32 MB.

So, if:

M = Amount of RAM in GB, and S = Amount of swap in GB, then

```
If M < 2
 S = M *2
Else
 S = M + 2
```

Using this formula, a system with 2 GB of physical RAM would have 4 GB of swap, while one with 3 GB of physical RAM would have 5 GB of swap. Creating a large swap space partition can be especially helpful if you plan to upgrade your RAM at a later time.

For systems with really large amounts of RAM (more than 32 GB) you can likely get away with a smaller swap partition (around 1x, or less, of physical RAM).

A `/boot/` partition (100 MB)

The partition mounted on `/boot/` contains the operating system kernel (which allows your system to boot Fedora), along with files used during the bootstrap process. Due to limitations, creating

a native ext3 partition to hold these files is required. For most users, a 100 MB boot partition is sufficient.

> **Note**
>
> If your hard drive is more than 1024 cylinders (and your system was manufactured more than two years ago), you may need to create a `/boot/` partition if you want the `/` (root) partition to use all of the remaining space on your hard drive.

> **Note**
>
> If you have a RAID card, be aware that some BIOSes do not support booting from the RAID card. In cases such as these, the `/boot/` partition must be created on a partition outside of the RAID array, such as on a separate hard drive.

A `root` partition (3.0 GB - 5.0 GB)

This is where "`/`" (the root directory) is located. In this setup, all files (except those stored in `/boot`) are on the root partition.

A 3.0 GB partition allows you to install a minimal installation, while a 5.0 GB root partition lets you perform a full installation, choosing all package groups.

> **Root and `/root`**
>
> The `/` (or root) partition is the top of the directory structure. The `/root` directory `/root` (sometimes pronounced "slash-root") directory is the home directory of the user account for system administration.

Many systems have more partitions than the minimum listed above. Choose partitions based on your particular system needs. For example, consider creating a separate `/home` partition on systems that store user data. Refer to *Section 7.21.4.1.1, "Advice on Partitions"* for more information.

If you create many partitions instead of one large `/` partition, upgrades become easier. Refer to the description of **Disk Druid's** Edit option in *Section 7.21.2, "Disk Druid's Buttons"* for more information.

The following table summarizes minimum partition sizes for the partitions containing the listed directories. You *do not* have to make a separate partition for each of these directories. For instance, if the partition containing `/foo` must be at least 500 MB, and you do not make a separate `/foo` partition, then the `/` (root) partition must be at least 500 MB.

Directory	Minimum size
`/`	250 MB
`/usr`	250 MB
`/tmp`	50 MB
`/var`	384 MB
`/home`	100 MB
`/boot`	75 MB

Table 7.3. Minimum partition sizes

> **Leave Excess Capacity Unallocated**
>
> Only assign storage capacity to those partitions you require immediately. You may allocate free space at any time, to meet needs as they occur. To learn about a more flexible method for storage management, refer to *Appendix D, Understanding LVM*.

If you are not sure how best to configure the partitions for your computer, accept the default partition layout.

7.21.4.1.1. Advice on Partitions

Optimal partition setup depends on the usage for the Linux system in question. The following tips may help you decide how to allocate your disk space.

- If you expect that you or other users will store data on the system, create a separate partition for the `/home` directory within a volume group. With a separate `/home` partition, you may upgrade or reinstall Fedora without erasing user data files.

- Each kernel installed on your system requires approximately 6 MB on the `/boot` partition. Unless you plan to install a great many kernels, the default partition size of 100 MB for `/boot` should suffice.

- The `/var` directory holds content for a number of applications, including the **Apache** web server. It also is used to store downloaded update packages on a temporary basis. Ensure that the partition containing the `/var` directory has enough space to download pending updates and hold your other content.

> **Pending Updates**
>
> Because Fedora is a rapidly progressing collection of software, many updates may be available late in a release cycle. You can add an update repository to the sources for installation later to minimize this issue. Refer to *Section 7.23.1, "Installing from Additional Repositories"* for more information.

- The `/usr` directory holds the majority of software content on a Fedora system. For an installation of the default set of software, allocate at least 4 GB of space. If you are a software developer or plan to use your Fedora system to learn software development skills, you may want to at least double this allocation.

- Consider leaving a portion of the space in an LVM volume group unallocated. This unallocated space gives you flexibility if your space requirements change but you do not wish to remove data from other partitions to reallocate storage.

- If you separate subdirectories into partitions, you can retain content in those subdirectories if you decide to install a new version of Fedora over your current system. For instance, if you intend to run a **MySQL** database in `/var/lib/mysql`, make a separate partition for that directory in case you need to reinstall later.

The following table is a possible partition setup for a system with a single, new 80 GB hard disk and 1 GB of RAM. Note that approximately 10 GB of the volume group is unallocated to allow for future growth.

> ### Example Usage
>
> This setup is not optimal for all use cases.

Partition	Size and type
`/boot`	100 MB `ext3` partition
`swap`	2 GB swap
LVM physical volume	Remaining space, as one LVM volume group

Table 7.4. Example partition setup

The physical volume is assigned to the default volume group and divided into the following logical volumes:

Partition	Size and type
`/`	3 GB ext3
`/usr`	8 GB ext3
`/usr/local`	2 GB ext3
`/var`	4 GB ext3
`/home`	50 GB ext3

Table 7.5. Example partition setup: LVM physical volume

Example 7.1. Example partition setup

7.21.5. Adding Partitions

To add a new partition, select the **New** button. A dialog box appears (refer to *Figure 7.24, "Creating a New Partition"*).

> ### Note
>
> You must dedicate at least one partition for this installation, and optionally more. For more information, refer to *Appendix A, An Introduction to Disk Partitions*.

Figure 7.24. Creating a New Partition

- **Mount Point**: Enter the partition's mount point. For example, if this partition should be the root partition, enter `/`; enter `/boot` for the `/boot` partition, and so on. You can also use the pull-down menu to choose the correct mount point for your partition. For a swap partition the mount point should not be set - setting the filesystem type to swap is sufficient.

- **File System Type**: Using the pull-down menu, select the appropriate file system type for this partition. For more information on file system types, refer to *Section 7.21.5.1, "File System Types"*.

- **Allowable Drives**: This field contains a list of the hard disks installed on your system. If a hard disk's box is highlighted, then a desired partition can be created on that hard disk. If the box is *not* checked, then the partition will *never* be created on that hard disk. By using different checkbox settings, you can have **Disk Druid** place partitions where you need them, or let **Disk Druid** decide where partitions should go.

- **Size (MB)**: Enter the size (in megabytes) of the partition. Note, this field starts with 100 MB; unless changed, only a 100 MB partition will be created.

- **Additional Size Options**: Choose whether to keep this partition at a fixed size, to allow it to "grow" (fill up the available hard drive space) to a certain point, or to allow it to grow to fill any remaining hard drive space available.

If you choose **Fill all space up to (MB)**, you must give size constraints in the field to the right of this option. This allows you to keep a certain amount of space free on your hard drive for future use.

- **Force to be a primary partition**: Select whether the partition you are creating should be one of the first four partitions on the hard drive. If unselected, the partition is created as a logical partition. Refer to *Section A.1.3, "Partitions within Partitions — An Overview of Extended Partitions"*, for more information.

- **OK**: Select **OK** once you are satisfied with the settings and wish to create the partition.

- **Cancel**: Select **Cancel** if you do not want to create the partition.

7.21.5.1. File System Types

Fedora allows you to create different partition types, based on the file system they will use. The following is a brief description of the different file systems available, and how they can be utilized.

- **ext2** — An ext2 file system supports standard Unix file types (regular files, directories, symbolic links, etc). It provides the ability to assign long file names, up to 255 characters.

- **ext3** — The ext3 file system is based on the ext2 file system and has one main advantage — journaling. Using a journaling file system reduces time spent recovering a file system after a crash as there is no need to `fsck`[2] the file system. The ext3 file system is selected by default and is highly recommended.

- **physical volume (LVM)** — Creating one or more physical volume (LVM) partitions allows you to create an LVM logical volume. LVM can improve performance when using physical disks.

- **software RAID** — Creating two or more software RAID partitions allows you to create a RAID device.

- **swap** — Swap partitions are used to support virtual memory. In other words, data is written to a swap partition when there is not enough RAM to store the data your system is processing.

- **vfat** — The VFAT file system is a Linux file system that is compatible with Microsoft Windows long filenames on the FAT file system. This file system must be used for the `/boot/efi/` partition on Itanium systems.

7.21.6. Editing Partitions

To edit a partition, select the **Edit** button or double-click on the existing partition.

> **Note**
>
> If the partition already exists on your disk, you can only change the partition's mount point. To make any other changes, you must delete the partition and recreate it.

7.21.7. Deleting a Partition

To delete a partition, highlight it in the **Partitions** section and click the **Delete** button. Confirm the deletion when prompted.

For further installation instructions for x86, AMD64, and Intel® 64 systems, skip to *Section 7.22, "x86, AMD64, and Intel 64 Boot Loader Configuration"*.

7.22. x86, AMD64, and Intel® 64 Boot Loader Configuration

To boot the system without boot media, you usually need to install a boot loader. A boot loader is the first software program that runs when a computer starts. It is responsible for loading and transferring control to the operating system kernel software. The kernel, in turn, initializes the rest of the operating system.

Installing in text mode

If you install Fedora in text mode, the installer configures the bootloader automatically and you cannot customize bootloader settings during the installation process.

GRUB (GRand Unified Bootloader), which is installed by default, is a very powerful boot loader. GRUB can load a variety of free operating systems, as well as proprietary operating systems with chain-loading (the mechanism for loading unsupported operating systems, such as DOS or Windows, by loading another boot loader).

The GRUB boot menu

The GRUB menu defaults to being hidden, except on dual-boot systems. To show the GRUB menu during system boot, press and hold the `shift` key before the kernel is loaded. (Any other key works as well but the `shift` key is the safest to use.)
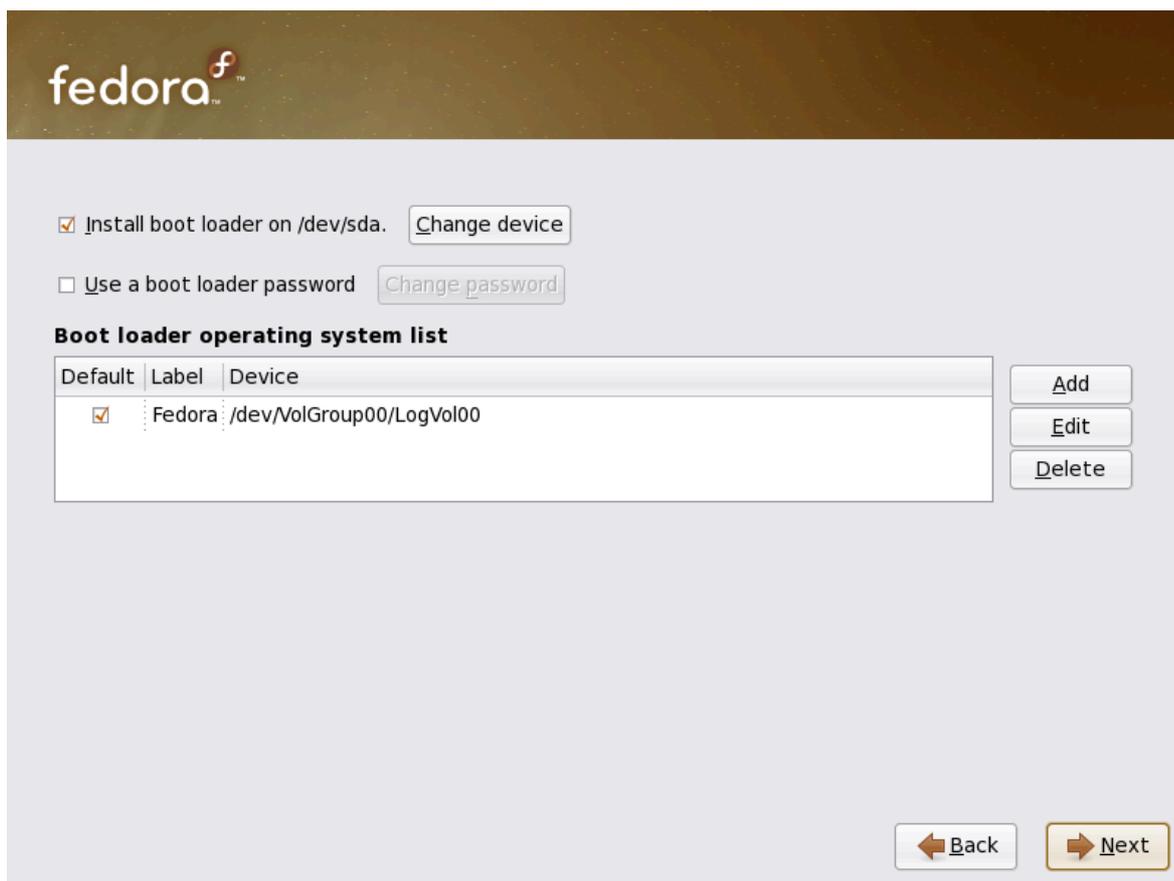
Figure 7.25. Boot Loader Configuration

If there are no other operating systems on your computer, or you are completely removing any other operating systems the installation program will install **GRUB** as your boot loader without any intervention. In that case you may continue on to *Section 7.23, "Package Group Selection"*.

You may have a boot loader installed on your system already. An operating system may install its own preferred boot loader, or you may have installed a third-party boot loader.If your boot loader does not recognize Linux partitions, you may not be able to boot Fedora. Use **GRUB** as your boot loader to boot Linux and most other operating systems. Follow the directions in this chapter to install **GRUB**.

> ## Installing GRUB
> If you install GRUB, it may overwrite your existing boot loader.

By default, the installation program installs GRUB in the master boot record or MBR, of the device for the root file system. To decline installation of a new boot loader, unselect **Install boot loader on /dev/sda**.

> ## Warning
> If you choose not to install GRUB for any reason, you will not be able to boot the system directly, and you must use another boot method (such as a commercial boot

> loader application). Use this option only if you are sure you have another way of booting the system!

If you have other operating systems already installed, Fedora attempts to automatically detect and configure **GRUB** to boot them. You may manually configure any additional operating systems if **GRUB** does not detect them.

To add, remove, or change the detected operating system settings, use the options provided.

Add

Select **Add** to include an additional operating system in GRUB.

Select the disk partition which contains the bootable operating system from the drop-down list and give the entry a label. **GRUB** displays this label in its boot menu.

Edit

To change an entry in the GRUB boot menu, select the entry and then select **Edit**.

Delete

To remove an entry from the GRUB boot menu, select the entry and then select **Delete**.

Select **Default** beside the preferred boot partition to choose your default bootable OS. You cannot move forward in the installation unless you choose a default boot image.

> **Note**
>
> The **Label** column lists what you must enter at the boot prompt, in non-graphical boot loaders, in order to boot the desired operating system.
>
> Once you have loaded the GRUB boot screen, use the arrow keys to choose a boot label or type **e** for edit. You are presented with a list of items in the configuration file for the boot label you have selected.

Boot loader passwords provide a security mechanism in an environment where physical access to your server is available.

If you are installing a boot loader, you should create a password to protect your system. Without a boot loader password, users with access to your system can pass options to the kernel which can compromise your system security. With a boot loader password in place, the password must first be entered before selecting any non-standard boot options. However, it is still possible for someone with physical access to the machine to boot from a diskette, CD-ROM, or USB media if the BIOS supports it. Security plans which include boot loader passwords should also address alternate boot methods.

> **GRUB Passwords Not Required**
>
> You may not require a **GRUB** password if your system only has trusted operators, or is physically secured with controlled console access. However, if an untrusted person can get physical access to your computer's keyboard and monitor, that person can reboot the system and access **GRUB**. A password is helpful in this case.

If you choose to use a boot loader password to enhance your system security, be sure to select the checkbox labeled **Use a boot loader password**.

Once selected, enter a password and confirm it.

GRUB stores the password in encrypted form, so it *cannot* be read or recovered. If you forget the boot password, boot the system normally and then change the password entry in the `/boot/grub/grub.conf` file. If you cannot boot, you may be able to use the "rescue" mode on the first Fedora installation disc to reset the GRUB password.

If you do need to change the **GRUB** password, use the `grub-md5-crypt` utility. For information on using this utility, use the command `man grub-md5-crypt` in a terminal window to read the manual pages.

To configure more advanced boot loader options, such as changing the drive order or passing options to the kernel, be sure **Configure advanced boot loader options** is selected before clicking **Next**.

7.22.1. Advanced Boot Loader Configuration

Now that you have chosen which boot loader to install, you can also determine where you want the boot loader to be installed. You may install the boot loader in one of two places:

- The master boot record (MBR) — This is the recommended place to install a boot loader, unless the MBR already starts another operating system loader, such as System Commander. The MBR is a special area on your hard drive that is automatically loaded by your computer's BIOS, and is the earliest point at which the boot loader can take control of the boot process. If you install it in the MBR, when your machine boots, GRUB presents a boot prompt. You can then boot Fedora or any other operating system that you have configured the boot loader to boot.

- The first sector of your boot partition — This is recommended if you are already using another boot loader on your system. In this case, your other boot loader takes control first. You can then configure that boot loader to start GRUB, which then boots Fedora.

> **GRUB as a Secondary Boot Loader**
>
> If you install GRUB as a secondary boot loader, you must reconfigure your primary boot loader whenever you install and boot from a new kernel. The kernel of an operating system such as Microsoft Windows does not boot in the same fashion. Most users therefore use GRUB as the primary boot loader on dual-boot systems.

Figure 7.26. Boot Loader Installation

> **Note**
>
> If you have a RAID card, be aware that some BIOSes do not support booting from the RAID card. In cases such as these, the boot loader *should not* be installed on the MBR of the RAID array. Rather, the boot loader should be installed on the MBR of the same drive as the `/boot/` partition was created.

If your system only uses Fedora, you should choose the MBR.

Click the **Change Drive Order** button if you would like to rearrange the drive order or if your BIOS does not return the correct drive order. Changing the drive order may be useful if you have multiple SCSI adapters, or both SCSI and IDE adapters, and you want to boot from the SCSI device.

> **Note**
>
> While partitioning your hard drive, keep in mind that the BIOS in some older systems cannot access more than the first 1024 cylinders on a hard drive. If this is the case, leave enough room for the `/boot` Linux partition on the first 1024 cylinders of your hard drive to boot Linux. The other Linux partitions can be after cylinder 1024.
>
> In `parted`, 1024 cylinders equals 528MB. For more information, refer to:
>
> ```
> http://www.pcguide.com/ref/hdd/bios/sizeMB504-c.html
> ```

7.22.2. Rescue Mode

Rescue mode provides the ability to boot a small Fedora environment entirely from boot media or some other boot method instead of the system's hard drive. There may be times when you are unable to get Fedora running completely enough to access files on your system's hard drive. Using rescue mode, you can access the files stored on your system's hard drive, even if you cannot actually run Fedora from that hard drive. If you need to use rescue mode, try the following method:

* Using the CD-ROM to boot an x86, AMD64, or Intel® 64 system, type `linux rescue` at the installation boot prompt. Itanium users should type `elilo linux rescue` to enter rescue mode.

7.22.3. Alternative Boot Loaders

If you do not wish to use a boot loader, you have several alternatives:

LOADLIN

You can load Linux from MS-DOS. Unfortunately, this requires a copy of the Linux kernel (and an initial RAM disk, if you have a SCSI adapter) to be available on an MS-DOS partition. The only way to accomplish this is to boot your Fedora system using some other method (for example, from a boot CD-ROM) and then copy the kernel to an MS-DOS partition. LOADLIN is available from

ftp://metalab.unc.edu/pub/Linux/system/boot/dualboot/

and associated mirror sites.

SYSLINUX

SYSLINUX is an MS-DOS program very similar to LOADLIN. It is also available from

ftp://metalab.unc.edu/pub/Linux/system/boot/loaders/

and associated mirror sites.

Commercial boot loaders

You can load Linux using commercial boot loaders. For example, System Commander and Partition Magic are able to boot Linux (but still require GRUB to be installed in your Linux root partition).

> **Note**
>
> Boot loaders such as LOADLIN and System Commander are considered to be third-party boot loaders and are not supported by Red Hat.

7.22.4. SMP Motherboards and GRUB

In previous versions of Fedora there were two different kernel versions, a uniprocessor version and an SMP version. In Fedora 11 the kernel is SMP-enabled by default and will take advantage of multiple core, hyperthreading, and multiple CPU capabilities when they are present. This same kernel can run on single CPUs with a single core and no hyperthreading.

7.23. Package Group Selection

Now that you have made most of the choices for your installation, you are ready to confirm the default package selection or customize packages for your system.

The **Package Installation Defaults** screen appears and details the default package set for your Fedora installation. This screen varies depending on the version of Fedora you are installing.

> **Installing from a Live Image**
>
> If you install from a Fedora Live image, you cannot make package selections. This installation method transfers a copy of the Live image rather than installing packages from a repository. To change the package selection, complete the installation, then use the **Add/Remove Software** application to make desired changes.

> **Installing in text mode**
>
> If you install Fedora in text mode, you cannot make package selections. The installer automatically selects packages only from the base and core groups. These packages are sufficient to ensure that the system is operational at the end of the installation process, ready to install updates and new packages. To change the package selection, complete the installation, then use the **Add/Remove Software** application to make desired changes.

Figure 7.27. Package Group Selection

By default, the Fedora installation process loads a selection of software that is suitable for a desktop system. To include or remove software for common tasks, select the relevant items from the list:

Office and Productivity
 This option provides the OpenOffice.org productivity suite, the Planner project management application, graphical tools such as the GIMP, and multimedia applications.

Software Development
 This option provides the necessary tools to compile software on your Fedora system.

Web server
 This option provides the Apache Web server.

If you choose to accept the current package list, skip ahead to *Section 7.24, "Preparing to Install"*.

To select a component, click on the checkbox beside it (refer to *Figure 7.27, "Package Group Selection"*).

To customize your package set further, select the **Customize now** option on the screen. Clicking **Next** takes you to the **Package Group Selection** screen.

7.23.1. Installing from Additional Repositories

You can define additional *repositories* to increase the software available to your system during installation. A repository is a network location that stores software packages along with *metadata* that

describes them. Many of the software packages used in Fedora require other software to be installed. The installer uses the metadata to ensure that these requirements are met for every piece of software you select for installation.

The basic options are:

- The **Installation Repo** repository is automatically selected for you. This represents the collection of software available on your installation CD or DVD.

- The **Fedora 11 - i386** repository contains the complete collection of software that was released as Fedora 11, with the various pieces of software in their versions that were current at the time of release. If you are installing from the Fedora 11 DVD or *set* of CDs, this option does not offer you anything extra. However, if you are installing from a Fedora Live CD, this option provides access to far more software than is included on the disk. Note that the computer must have access to the internet to use this option.

- The **Fedora 11 - i386 - Updates** repository contains the complete collection of software that was released as Fedora 11, with the various pieces of software in their most current stable versions. This option not only installs the software that you select, but makes sure that it is fully updated as well. Note that the computer must have access to the internet to use this option.
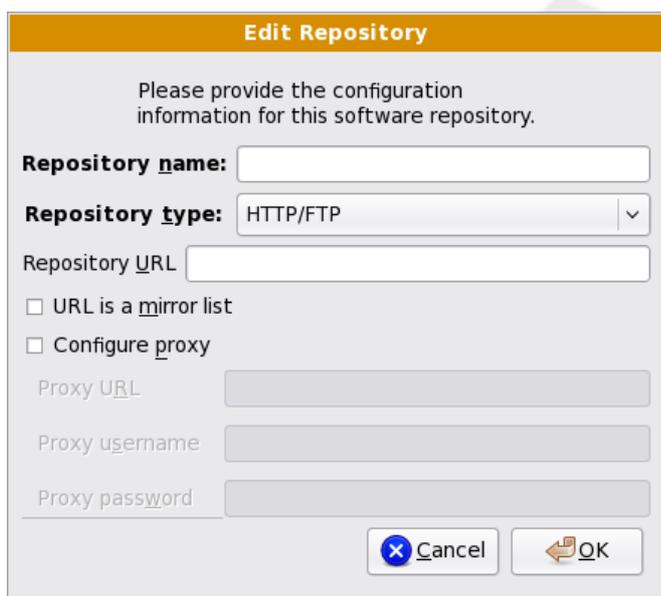
Figure 7.28. Adding a software repository

To include software from *repositories* other than the Fedora package collection, select **Add additional software repositories**. You may provide the location of a repository of third-party software. Depending on the configuration of that repository, you may be able to select non-Fedora software during installation.

To edit an existing software repository location, select the repository in the list and then select **Modify repository**.

> **Network Access Required**
>
> If you change the repository information during a non-network installation, such as from a Fedora DVD, the installer prompts you for network configuration information.

If you select **Add additional software repositories**, the **Edit repository** dialog appears. Provide a **Repository name** and the **Repository URL** for its location.

> ### Fedora Software Mirrors
> To find a Fedora software mirror near you, refer to *http://fedoraproject.org/wiki/Mirrors*.

Once you have located a mirror, to determine the URL to use, find the directory on the mirror that *contains* a directory named `repodata`. For instance, the "Everything" repository for Fedora is typically located in a directory tree `releases/11/Everything/`*`arch`*`/os`, where *arch* is a system architecture name.

Once you provide information for an additional repository, the installer reads the package metadata over the network. Software that is specially marked is then included in the package group selection system. See *Section 7.23.2, "Customizing the Software Selection"* for more information on selecting packages.

> ### Backtracking Removes Repository Metadata
> If you choose **Back** from the package selection screen, any extra repository data you may have entered is lost. This allows you to effectively cancel extra repositories. Currently there is no way to cancel only a single repository once entered.

7.23.2. Customizing the Software Selection

Select **Customize now** to specify the software packages for your final system in more detail. This option causes the installation process to display an additional customization screen when you select **Next**.

> ### Installing Support for Additional Languages
> Select **Customize now** to install support for additional languages. Refer to *Section 7.23.2.2, "Additional Language Support"* for more information on configuring language support.
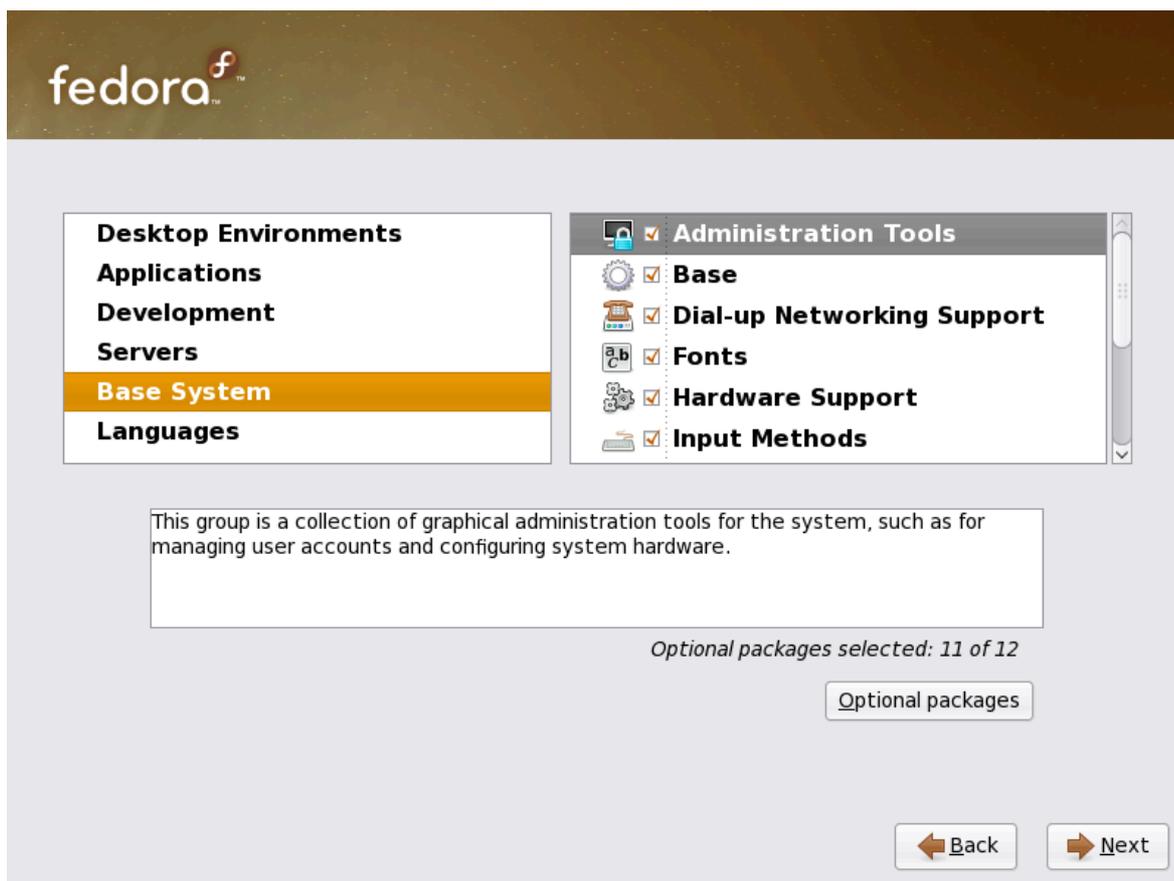
Figure 7.29. Package Group Details

Fedora divides the included software into
package groups. For ease of use, the package selection screen displays these groups as categories.

You can select package groups, which group components together according to function (for example, **X Window System** and **Editors**), individual packages, or a combination of the two.

To view the package groups for a category, select the category from the list on the left. The list on the right displays the package groups for the currently selected category.

To specify a package group for installation, select the check box next to the group. The box at the bottom of the screen displays the details of the package group that is currently highlighted. *None* of the packages from a group will be installed unless the check box for that group is selected.

If you select a package group, Fedora automatically installs the base and mandatory packages for that group. To change which optional packages within a selected group will be installed, select the **Optional Packages** button under the description of the group. Then use the check box next to an individual package name to change its selection.

After you choose the desired packages, select **Next** to proceed. Fedora checks your selection, and automatically adds any extra packages required to use the software you select. When you have finished selecting packages, click **Close** to save your optional package selections and return to the main package selection screen.

7.23.2.1. Changing Your Mind

The packages that you select are not permanent. After you boot your system, use the **Add/Remove Software** tool to either install new software or remove installed packages. To run this tool, from the main menu, select **System** → **Administration** → **Add/Remove Software**. The Fedora software management system downloads the latest packages from network servers, rather than using those on the installation discs.

7.23.2.2. Additional Language Support

Your Fedora system automatically supports the language that you selected at the start of the installation process. To include support for additional languages, select the package group for those languages from the **Languages** category.

7.23.2.3. Core Network Services

All Fedora installations include the following network services:

• centralized logging through syslog

• email through SMTP (Simple Mail Transfer Protocol)

• network file sharing through NFS (Network File System)

• remote access through SSH (Secure SHell)

• resource advertising through mDNS (multicast DNS)

The default installation also provides:

• network file transfer through HTTP (HyperText Transfer Protocol)

• printing through CUPS (Common UNIX Printing System)

• remote desktop access through VNC (Virtual Network Computing)

Some automated processes on your Fedora system use the email service to send reports and messages to the system administrator. By default, the email, logging, and printing services do not accept connections from other systems. Fedora installs the NFS sharing, HTTP, and VNC components without enabling those services.

You may configure your Fedora system after installation to offer email, file sharing, logging, printing and remote desktop access services. The SSH service is enabled by default. You may use NFS to access files on other systems without enabling the NFS sharing service.

7.24. Preparing to Install

7.24.1. Prepare to Install

A screen preparing you for the installation of Fedora now appears.

For your reference, a complete log of your installation can be found in `/root/install.log` once you reboot your system.

> ⚠️ **Warning**
>
> If, for some reason, you would rather not continue with the installation process, this is your last opportunity to safely cancel the process and reboot your machine. Once you press the **Next** button, partitions are written and packages are installed. If you wish to abort the installation, you should reboot now before any existing information on any hard drive is rewritten.

To cancel this installation process, press your computer's Reset button or use the `Control+Alt+Delete` key combination to restart your machine.

7.25. Installing Packages

At this point there is nothing left for you to do until all the packages have been installed. How quickly this happens depends on the number of packages you have selected and your computer's speed.

Fedora reports the installation progress on the screen as it writes the selected packages to your system. Network and DVD installations require no further action. If you are using CDs to install, Fedora prompts you to change discs periodically. After you insert a disc, select **OK** to resume the installation.

After installation completes, select **Reboot** to restart your computer. Fedora ejects any loaded discs before the computer reboots.

> 📌 **Installing from a Live Image**
>
> If you install from a Fedora Live image, no reboot prompt appears. You may continue to use the Live image as desired, and reboot the system at any time to enjoy the newly installed Fedora system.

7.26. Installation Complete

Congratulations! Your Fedora installation is now complete!

The installation program prompts you to prepare your system for reboot. Remember to remove any installation media if it is not ejected automatically upon reboot.

Troubleshooting Installation on an Intel or AMD System

This appendix discusses some common installation problems and their solutions.

8.1. You are unable to boot Fedora

8.1.1. Are You Unable to Boot With Your RAID Card?

If you have performed an installation and cannot boot your system properly, you may need to reinstall and create your partitions differently.

Some BIOSes do not support booting from RAID cards. At the end of an installation, a text-based screen showing the boot loader prompt (for example, `GRUB:`) and a flashing cursor may be all that appears. If this is the case, you must repartition your system.

Whether you choose automatic or manual partitioning, you must install your `/boot` partition outside of the RAID array, such as on a separate hard drive. An internal hard drive is necessary to use for partition creation with problematic RAID cards.

You must also install your preferred boot loader (GRUB or LILO) on the MBR of a drive that is outside of the RAID array. This should be the same drive that hosts the `/boot/` partition.

Once these changes have been made, you should be able to finish your installation and boot the system properly.

8.1.2. Is Your System Displaying Signal 11 Errors?

A signal 11 error, commonly know as a *segmentation fault*, means that the program accessed a memory location that was not assigned to it. A signal 11 error may be due to a bug in one of the software programs that is installed, or faulty hardware.

If you receive a fatal signal 11 error during your installation, it is probably due to a hardware error in memory on your system's bus. Like other operating systems, Fedora places its own demands on your system's hardware. Some of this hardware may not be able to meet those demands, even if they work properly under another OS.

Ensure that you have the latest installation updates and images. Review the online errata to see if newer versions are available. If the latest images still fail, it may be due to a problem with your hardware. Commonly, these errors are in your memory or CPU-cache. A possible solution for this error is turning off the CPU-cache in the BIOS, if your system supports this. You could also try to swap your memory around in the motherboard slots to check if the problem is either slot or memory related.

Another option is to perform a media check on your installation CD-ROMs. **Anaconda**, the installation program, has the ability to test the integrity of the installation media. It works with the CD, DVD, hard drive ISO, and NFS ISO installation methods. Red Hat recommends that you test all installation media before starting the installation process, and before reporting any installation-related bugs (many of the bugs reported are actually due to improperly-burned CDs). To use this test, type the following command at the `boot:` prompt:

```
linux mediacheck
```

For more information concerning signal 11 errors, refer to:

http://www.bitwizard.nl/sig11/

8.2. Trouble Beginning the Installation

8.2.1. Problems with Booting into the Graphical Installation

There are some video cards that have trouble booting into the graphical installation program. If the installation program does not run using its default settings, it tries to run in a lower resolution mode. If that still fails, the installation program attempts to run in text mode.

One possible solution is to try using the **resolution=** boot option. This option may be most helpful for laptop users. Another solution to try is the **driver=** option to specify the driver that should be loaded for your video card. If this works, it should be reported as a bug as the installer has failed to autodetect your videocard. Refer to *Chapter 9, Boot Options* for more information on boot options.

> **Note**
>
> To disable frame buffer support and allow the installation program to run in text mode, try using the **nofb** boot option. This command may be necessary for accessibility with some screen reading hardware.

8.3. Trouble During the Installation

8.3.1. `No devices found to install Fedora` Error Message

If you receive an error message stating `No devices found to install Fedora`, there is probably a SCSI controller that is not being recognized by the installation program.

Check your hardware vendor's website to determine if a driver diskette image is available that fixes your problem. For more general information on driver diskettes, refer to *Chapter 5, Driver Media for Intel and AMD Systems*.

You can also refer to the LinuxQuestions.org Hardware Compatibility List, available online at:

http://www.linuxquestions.org/hcl/index.php

8.3.2. Saving traceback messages without removeable media

If you receive a traceback error message during installation, you can usually save it to removeable media, for example a USB flash drive or a floppy disk.

If you do not have removeable media available on your system, you can **scp** the error message to a remote system.

When the traceback dialog appears, the traceback error message is automatically written to a file named **/tmp/anacdump.txt**. Once the dialog appears, switch over to a new tty (virtual console) by pressing the keys **Ctrl+Alt+F2** and **scp** the message written to **/tmp/anacdump.txt** to a known working remote system.

8.3.3. Trouble with Partition Tables

If you receive an error after the **Disk Partitioning Setup** (*Section 7.18, "Disk Partitioning Setup"*) phase of the installation saying something similar to

The partition table on device hda was unreadable. To create new partitions it must be initialized, causing the loss of ALL DATA on this drive.

you may not have a partition table on that drive or the partition table on the drive may not be recognizable by the partitioning software used in the installation program.

Users who have used programs such as **EZ-BIOS** have experienced similar problems, causing data to be lost (assuming the data was not backed up before the installation began) that could not be recovered.

No matter what type of installation you are performing, backups of the existing data on your systems should always be made.

8.3.4. Using Remaining Space

You have a **swap** and a **/** (root) partition created, and you have selected the root partition to use the remaining space, but it does not fill the hard drive.

If your hard drive is more than 1024 cylinders, you must create a **/boot** partition if you want the **/** (root) partition to use all of the remaining space on your hard drive.

8.3.5. Other Partitioning Problems

If you are using **Disk Druid** to create partitions, but cannot move to the next screen, you probably have not created all the partitions necessary for **Disk Druid**'s dependencies to be satisfied.

You must have the following partitions as a bare minimum:

- A **/** (root) partition

- A <swap> partition of type swap

> **Note**
>
> When defining a partition's type as swap, do not assign it a mount point. **Disk Druid** automatically assigns the mount point for you.

8.3.6. Are You Seeing Python Errors?

During some upgrades or installations of Fedora, the installation program (also known as **anaconda**) may fail with a Python or traceback error. This error may occur after the selection of individual packages or while trying to save the upgrade log in the **/tmp/** directory. The error may look similar to:

```
Traceback (innermost last):
File "/var/tmp/anaconda-7.1//usr/lib/anaconda/iw/progress_gui.py", line
 20, in run
rc = self.todo.doInstall ()
File "/var/tmp/anaconda-7.1//usr/lib/anaconda/todo.py", line 1468, in
 doInstall
self.fstab.savePartitions ()
File "fstab.py", line 221, in savePartitions
sys.exit(0)
SystemExit: 0
Local variables in innermost frame:
self: <fstab.GuiFstab instance at 8446fe0>
sys: <module 'sys' (built-in)>
ToDo object: (itodo  ToDo  p1  (dp2  S'method'  p3  (iimage
 CdromInstallMethod
p4  (dp5  S'progressWindow'  p6   <failed>
```

This error occurs in some systems where links to **/tmp/** are symbolic to other locations or have been changed since creation. These symbolic or changed links are invalid during the installation process, so the installation program cannot write information and fails.

If you experience such an error, first try to download any available updates for **anaconda**. Updates for **anaconda** and instructions for using them can be found at:

http://fedoraproject.org/wiki/Anaconda/Updates

The **anaconda** website may also be a useful reference and can be found online at:

http://fedoraproject.org/wiki/Anaconda

You can also search for bug reports related to this problem. To search Red Hat's bug tracking system, go to:

http://bugzilla.redhat.com/bugzilla/

8.4. Problems After Installation

8.4.1. Trouble With the Graphical GRUB Screen on an x86-based System?

If you are experiencing problems with GRUB, you may need to disable the graphical boot screen. To do this, become the root user and edit the **/boot/grub/grub.conf** file.

Within the **grub.conf** file, comment out the line which begins with splashimage by inserting the # character at the beginning of the line.

Press **Enter** to exit the editing mode.

Once the boot loader screen has returned, type **b** to boot the system.

Once you reboot, the **grub.conf** file is reread and any changes you have made take effect.

You may re-enable the graphical boot screen by uncommenting (or adding) the above line back into the **grub.conf** file.

8.4.2. Booting into a Graphical Environment

If you have installed the X Window System but are not seeing a graphical desktop environment once you log into your Red Hat Enterprise Linux system, you can start the X Window System graphical interface using the command **startx**.

Once you enter this command and press **Enter**, the graphical desktop environment is displayed.

Note, however, that this is just a one-time fix and does not change the log in process for future log ins.

To set up your system so that you can log in at a graphical login screen, you must edit one file, **/etc/inittab**, by changing just one number in the runlevel section. When you are finished, reboot the computer. The next time you log in, you are presented with a graphical login prompt.

Open a shell prompt. If you are in your user account, become root by typing the **su** command.

Now, type **gedit /etc/inittab** to edit the file with **gedit**. The file **/etc/inittab** opens. Within the first screen, a section of the file which looks like the following appears:

```
# Default runlevel. The runlevels used are:
#   0 - halt (Do NOT set initdefault to this)
#   1 - Single user mode
#   2 - Multiuser, without NFS (The same as 3, if you do not have
 networking)
#   3 - Full multiuser mode
#   4 - unused
#   5 - X11
#   6 - reboot (Do NOT set initdefault to this)
#  id:3:initdefault:
```

To change from a console to a graphical login, you should change the number in the line id:3:initdefault: from a 3 to a 5.

> ⚠️ **Warning**
>
> Change *only* the number of the default runlevel from 3 to 5.

Your changed line should look like the following:

```
   id:5:initdefault:
```

When you are satisfied with your change, save and exit the file using the **Ctrl+Q** keys. A window appears and asks if you would like to save the changes. Click **Save**.

The next time you log in after rebooting your system, you are presented with a graphical login prompt.

8.4.3. Problems with the X Window System (GUI)

If you are having trouble getting X (the X Window System) to start, you may not have installed it during your installation.

If you want X, you can either install the packages from the Fedora installation media or perform an upgrade.

If you elect to upgrade, select the X Window System packages, and choose GNOME, KDE, or both, during the upgrade package selection process.

8.4.4. Problems with the X Server Crashing and Non-Root Users

If you are having trouble with the X server crashing when anyone other than root logs in, you may have a full file system (or, a lack of available hard drive space).

To verify that this is the problem you are experiencing, run the following command:

```
df -h
```

The **df** command should help you diagnose which partition is full. For additional information about **df** and an explanation of the options available (such as the -h option used in this example), refer to the **df** man page by typing **man df** at a shell prompt.

A key indicator is 100% full or a percentage above 90% or 95% on a partition. The **/home/** and **/tmp/** partitions can sometimes fill up quickly with user files. You can make some room on that partition by removing old files. After you free up some disk space, try running X as the user that was unsuccessful before.

8.4.5. Problems When You Try to Log In

If you did not create a user account in the **Setup Agent**, log in as root and use the password you assigned to root.

If you cannot remember your root password, boot your system as **linux single**.

If you are using an x86-based system and GRUB is your installed boot loader, type **e** for edit when the GRUB boot screen has loaded. You are presented with a list of items in the configuration file for the boot label you have selected.

Choose the line that starts with **kernel** and type **e** to edit this boot entry.

At the end of the **kernel** line, add:

```
single
```

Press **Enter** to exit edit mode.

Once the boot loader screen has returned, type **b** to boot the system.

Once you have booted into single user mode and have access to the # prompt, you must type **passwd root**, which allows you to enter a new password for root. At this point you can type **shutdown -r now** to reboot the system with the new root password.

If you cannot remember your user account password, you must become root. To become root, type **su -** and enter your root password when prompted. Then, type **passwd <username>**. This allows you to enter a new password for the specified user account.

If the graphical login screen does not appear, check your hardware for compatibility issues. Linuxquestions.org maintains a Hardware Compatibility List at:

```
http://www.linuxquestions.org/hcl/index.php
```

8.4.6. Is Your RAM Not Being Recognized?

Sometimes, the kernel does not recognize all of your memory (RAM). You can check this with the **cat /proc/meminfo** command.

Verify that the displayed quantity is the same as the known amount of RAM in your system. If they are not equal, add the following line to the **/boot/grub/grub.conf**:

```
mem=xxM
```

Replace xx with the amount of RAM you have in megabytes.

In **/boot/grub/grub.conf**, the above example would look similar to the following:

```
# NOTICE: You have a /boot partition. This means that
#  all kernel paths are relative to /boot/
default=0
timeout=30
splashimage=(hd0,0)/grub/splash.xpm.gz
 title Fedora (2.6.27.19-170.2.35.fc10.i686)
root (hd0,1)
kernel /vmlinuz-2.6.27.19-170.2.35.fc10.i686 ro root=UUID=04a07c13-
e6bf-6d5a-b207-002689545705 mem=1024M
initrd /initrd-2.6.27.19-170.2.35.fc10.i686.img
```

Once you reboot, the changes made to **grub.conf** are reflected on your system.

Once you have loaded the GRUB boot screen, type **e** for edit. You are presented with a list of items in the configuration file for the boot label you have selected.

Choose the line that starts with kernel and type **e** to edit this boot entry.

At the end of the kernel line, add

```
mem=xxM
```

where xx equals the amount of RAM in your system.

Press **Enter** to exit edit mode.

Once the boot loader screen has returned, type **b** to boot the system.

Itanium users must enter boot commands with **elilo** followed by the boot command.

Remember to replace xx with the amount of RAM in your system. Press **Enter** to boot.

8.4.7. Your Printer Does Not Work

If you are not sure how to set up your printer or are having trouble getting it to work properly, try using the **Printer Configuration Tool**.

Type the **system-config-printer** command at a shell prompt to launch the **Printer Configuration Tool**. If you are not root, it prompts you for the root password to continue.

8.4.8. Problems with Sound Configuration

If, for some reason, you do not hear sound and know that you do have a sound card installed, you can run the **Sound Card Configuration Tool** (**system-config-soundcard**) utility.

To use the **Sound Card Configuration Tool**, choose **Main Menu** => **System** => **Administration** => **Soundcard Detection** in GNOME, or **Main Menu** => **Computer** => **System Settings** => **Multimedia** in KDE. A small text box pops up prompting you for your root password.

You can also type the **system-config-soundcard** command at a shell prompt to launch the **Sound Card Configuration Tool**. If you are not root, it prompts you for the root password to continue.

If the **Sound Card Configuration Tool** does not work (if the sample does not play and you still do not have audio sounds), it is likely that your sound card is not yet supported in Fedora.

8.4.9. Apache-based `httpd` service/Sendmail Hangs During Startup

If you are having trouble with the Apache-based **httpd** service or Sendmail hanging at startup, make sure the following line is in the **/etc/hosts** file:

```
127.0.0.1  localhost.localdomain  localhost
```

Part III. Advanced installation options

This part of the *Fedora Installation Guide* covers more complex and uncommon methods of installing Fedora, including:

- boot options.

- installing without media.

- installing through VNC.

- using **kickstart** to automate the installation process.

Boot Options

The Fedora installation system includes a range of functions and options for administrators. To use boot options, enter **linux** *option* at the `boot:` prompt.

If you specify more than one option, separate each of the options by a single space. For example:

```
linux option1 option2 option3
```

Anaconda Boot Options

The anaconda installer has many boot options, most are listed on the wiki *http://fedoraproject.org/wiki/Anaconda/Options*.

Kernel Boot Options

The *http://fedoraproject.org/wiki/KernelCommonProblems* page lists many common kernel boot options. The full list of kernel options is in the file /usr/share/doc/kernel-doc-*version*/Documentation/kernel-parameters.txt, which is installed with the kernel-doc package.

Rescue Mode

The Fedora installation and *rescue discs* may either boot with *rescue mode*, or load the installation system. For more information on rescue discs and rescue mode, refer to *Section 9.6.3, "Booting Your Computer with the Rescue Mode"*.

9.1. Configuring the Installation System at the Boot Menu

You can use the boot menu to specify a number of settings for the installation system, including:

- language

- display resolution

- interface type

- Installation method

- network settings

9.1.1. Specifying the Language

To set the language for both the installation process and the final system, specify the ISO code for that language with the `lang` option. Use the `keymap` option to configure the correct keyboard layout.

For example, the ISO codes **el_GR** and **gr** identify the Greek language and the Greek keyboard layout:

```
linux lang=el_GR keymap=gr
```

9.1.2. Configuring the Interface

You may force the installation system to use the lowest possible screen resolution (640x480) with the `lowres` option. To use a specific display resolution, enter `resolution=setting` as a boot option. For example, to set the display resolution to 1024x768, enter:

```
linux resolution=1024x768
```

To run the installation process in `text` mode, enter:

```
linux text
```

To enable support for a serial console, enter `serial` as an additional option.

Use `display=ip:0` to allow remote display forwarding. In this command, *ip* should be replaced with the IP address of the system on which you want the display to appear.

On the system you want the display to appear on, you must execute the command **xhost +remotehostname**, where *remotehostname* is the name of the host from which you are running the original display. Using the command **xhost +remotehostname** limits access to the remote display terminal and does not allow access from anyone or any system not specifically authorized for remote access.

9.1.3. Updating anaconda

You can install Fedora with a newer version of the **anaconda** installation program than the one supplied on your installation media.

The boot option

```
linux updates
```

presents you with a prompt that asks you for a floppy disk containing **anaconda** updates. You do not need to specify this option if you are performing a network installation and have already placed the updates image contents in **rhupdates/** on the server.

To load the **anaconda** updates from a network location instead, use:

```
linux updates=
```

followed by the URL for the location where the updates are stored.

9.1.4. Specifying the Installation Method

Use the `askmethod` option to display additional menus that enable you to specify the installation method and network settings. You may also configure the installation method and network settings at the `boot:` prompt itself.

To specify the installation method from the `boot:` prompt, use the `method` option. Refer to *Table 9.1, "Installation methods"* for the supported installation methods.

Installation method	Option format
CD or DVD drive	`method=cdrom`
Hard Drive	`method=hd://device/`
HTTP Server	`method=http://server.mydomain.com/directory/`
FTP Server	`method=ftp://server.mydomain.com/directory/`
NFS Server	`method=nfs:server.mydomain.com:/directory/`

Table 9.1. Installation methods

9.1.5. Manually Configuring the Network Settings

By default, the installation system uses DHCP to automatically obtain the correct network settings. To manually configure the network settings yourself, either enter them in the **Configure TCP/IP** screen, or at the `boot:` prompt. You may specify the `ip` address, `netmask`, `gateway`, and `dns` server settings for the installation system at the prompt. If you specify the network configuration at the `boot:` prompt, these settings are used for the installation process, and the **Configure TCP/IP** screen does not appear.

This example configures the network settings for an installation system that uses the IP address `192.168.1.10`:

```
linux ip=192.168.1.10 netmask=255.255.255.0 gateway=192.168.1.1
 dns=192.168.1.2,192.168.1.3
```

> **Configuring the Installed System**
>
> Use the Network Configuration screen to specify the network settings for the new system. Refer to *Section 7.15.1, "Manual configuration"* for more information on configuring the network settings for the installed system.

9.2. Enabling Remote Access to the Installation System

You may access either graphical or text interfaces for the installation system from any other system. Access to a text mode display requires **telnet**, which is installed by default on Fedora systems. To remotely access the graphical display of an installation system, use client software that supports the VNC (Virtual Network Computing) display protocol. A number of providers offer VNC clients for Microsoft Windows and Mac OS, as well as UNIX-based systems.

> ### Installing a VNC Client on Fedora
>
> Fedora includes **vncviewer**, the client provided by the developers of VNC. To obtain **vncviewer**, install the **vnc** package.

The installation system supports two methods of establishing a VNC connection. You may start the installation, and manually login to the graphical display with a VNC client on another system. Alternatively, you may configure the installation system to automatically connect to a VNC client on the network that is running in *listening mode*.

9.2.1. Enabling Remote Access with VNC

To enable remote graphical access to the installation system, enter two options at the prompt:

```
linux vnc vncpassword=qwerty
```

The vnc option enables the VNC service. The vncpassword option sets a password for remote access. The example shown above sets the password as **qwerty**.

> ### VNC Passwords
>
> The VNC password must be at least six characters long.

Specify the language, keyboard layout and network settings for the installation system with the screens that follow. You may then access the graphical interface through a VNC client. The installation system displays the correct connection setting for the VNC client:

```
Starting VNC...
The VNC server is now running.
Please connect to computer.mydomain.com:1 to begin the install...
Starting graphical installation...
Press <enter> for a shell
```

You may then login to the installation system with a VNC client. To run the **vncviewer** client on Fedora, choose **Applications** → **Accessories** → **VNC Viewer**, or type the command **vncviewer** in a terminal window. Enter the server and display number in the **VNC Server** dialog. For the example above, the **VNC Server** is **computer.mydomain.com:1**.

9.2.2. Connecting the Installation System to a VNC Listener

To have the installation system automatically connect to a VNC client, first start the client in listening mode. On Fedora systems, use the -listen option to run **vncviewer** as a listener. In a terminal window, enter the command:

```
vncviewer -listen
```

Firewall Reconfiguration Required

By default, **vncviewer** uses TCP port 5500 when in listening mode. To permit connections to this port from other systems, choose **System → Administration → Firewall**. Select **Other ports**, and **Add**. Enter **5500** in the **Port(s)** field, and specify **tcp** as the **Protocol**.

Once the listening client is active, start the installation system and set the VNC options at the `boot:` prompt. In addition to `vnc` and `vncpassword` options, use the `vncconnect` option to specify the name or IP address of the system that has the listening client. To specify the TCP port for the listener, add a colon and the port number to the name of the system.

For example, to connect to a VNC client on the system `desktop.mydomain.com` on the port 5500, enter the following at the `boot:` prompt:

```
linux vnc vncpassword=qwerty vncconnect=desktop.mydomain.com:5500
```

9.2.3. Enabling Remote Access with Telnet

To enable remote access to a text mode installation, use the `telnet` option at the `boot:` prompt:

```
linux text telnet
```

You may then connect to the installation system with the **telnet** utility. The **telnet** command requires the name or IP address of the installation system:

```
telnet computer.mydomain.com
```

Telnet Access Requires No Password

To ensure the security of the installation process, only use the `telnet` option to install systems on networks with restricted access.

9.3. Logging to a Remote System During the Installation

By default, the installation process sends log messages to the console as they are generated. You may specify that these messages go to a remote system that runs a *syslog* service.

To configure remote logging, add the `syslog` option. Specify the IP address of the logging system, and the UDP port number of the log service on that system. By default, syslog services that accept remote messages listen on UDP port 514.

For example, to connect to a syslog service on the system `192.168.1.20`, enter the following at the `boot:` prompt:

```
linux syslog=192.168.1.20:514
```

9.3.1. Configuring a Log Server

Fedora uses **rsyslog** to provide a syslog service. The default configuration of **rsyslog** rejects messages from remote systems.

> **⚠ Only Enable Remote Syslog Access on Secured Networks**
>
> The **rsyslog** configuration detailed below does not make use of any of the security measures available in **rsyslog** Crackers may slow or crash systems that permit access to the logging service, by sending large quantities of false log messages. In addition, hostile users may intercept or falsify messages sent to the logging service over the network.

To configure a Fedora system to accept log messages from other systems on the network, edit the file **/etc/rsyslog.conf**. You must use root privileges to edit the file **/etc/rsyslog.conf**. Uncomment the following lines by removing the hash preceding them:

```
$ModLoad imudp.so
  $UDPServerRun 514
```

Restart the **rsyslog** service to apply the change:

```
su -c '/sbin/service rsyslog restart'
```

Enter the root password when prompted.

> **Firewall Reconfiguration Required**
>
> By default, the syslog service listens on UDP port 514. To permit connections to this port from other systems, choose **System → Administration → Firewall**. Select **Other ports**, and **Add**. Enter **514** in the **Port(s)** field, and specify **udp** as the **Protocol**.

9.4. Automating the Installation with Kickstart

A
Kickstart file specifies settings for an installation. Once the installation system boots, it can read a Kickstart file and carry out the installation process without any further input from a user.

> **Every Installation Produces a Kickstart File**
>
> The Fedora installation process automatically writes a Kickstart file that contains the settings for the installed system. This file is always saved as **/root/anaconda-ks.cfg**. You may use this file to repeat the installation with identical settings, or modify copies to specify settings for other systems.

Fedora includes a graphical application to create and modify Kickstart files by selecting the options that you require. Use the package **system-config-kickstart** to install this utility. To load the Fedora Kickstart editor, choose **Applications** → **System Tools** → **Kickstart**.

Kickstart files list installation settings in plain text, with one option per line. This format lets you modify your Kickstart files with any text editor, and write scripts or applications that generate custom Kickstart files for your systems.

To automate the installation process with a Kickstart file, use the `ks` option to specify the name and location of the file:

```
linux ks=location/kickstart-file.cfg
```

You may use Kickstart files that are held on either removable storage, a hard drive, or a network server. Refer to *Table 9.2, "Kickstart sources"* for the supported Kickstart sources.

Kickstart source	Option format
CD or DVD drive	`ks=cdrom:/directory/ks.cfg`
Hard Drive	`ks=hd:/device/directory/ks.cfg`
Other Device	`ks=file:/device/directory/ks.cfg`
HTTP Server	`ks=http://server.mydomain.com/directory/ks.cfg`
FTP Server	`ks=ftp://server.mydomain.com/directory/ks.cfg`
NFS Server	`ks=nfs:server.mydomain.com:/directory/ks.cfg`

Table 9.2. Kickstart sources

To obtain a Kickstart file from a script or application on a Web server, specify the URL of the application with the `ks=` option. If you add the option `kssendmac`, the request also sends HTTP headers to the Web application. Your application can use these headers to identify the computer. This line sends a request with headers to the application *http://server.mydomain.com/kickstart.cgi*:

```
linux ks=http://server.mydomain.com/kickstart.cgi kssendmac
```

9.5. Enhancing Hardware Support

By default, Fedora attempts to automatically detect and configure support for all of the components of your computer. Fedora supports the majority of hardware in common use with the software *drivers* that are included with the operating system. To support other devices you may supply additional drivers during the installation process, or at a later time.

9.5.1. Adding Hardware Support with Driver Disks

The installation system can load drivers from disks, pen drives, or network servers to configure support for new devices. After the installation is complete, remove any driver disks and store them for later use.

Hardware manufacturers may supply

driver disks for Fedora with the device, or provide image files to prepare the disks. To obtain the latest drivers, download the correct file from the website of the manufacturer.

> **Driver Disks Supplied as Zipped Files**
>
> Driver disk images may be distributed as compressed archives, or zip files. For identification, the names of zip files include the extensions **.zip**, or **.tar.gz**. To extract the contents of a zipped file with a Fedora system, choose **Applications** → **Accessories** → **Archive Manager**.

To format a disk or pen drive with an image file, use the **dd** utility. For example, to prepare a diskette with the image file **drivers.img**, enter this command in a terminal window:

```
dd if=drivers.img of=/dev/fd0
```

To use a driver disk in the installation process, specify the dd option at the boot: prompt:

```
linux dd
```

Note that the dd option is interchangeable with the driverdisk option.

When prompted, select **Yes** to provide a driver disk. Choose the drive that holds the driver disk from the list on the **Driver Disk Source** text screen.

The installation system can also read drivers from disk images that are held on network servers. Refer to *Table 9.3, "Driver disk image sources"* for the supported sources of driver disk image files.

Image source	Option format
Select a drive or device	dd
HTTP Server	dd=*http://server.mydomain.com/directory/drivers.img*
FTP Server	dd=*ftp://server.mydomain.com/directory/drivers.img*
NFS Server	dd=*nfs:server.mydomain.com:/directory/drivers.img*

Table 9.3. Driver disk image sources

9.5.2. Overriding Automatic Hardware Detection

For some models of device automatic hardware configuration may fail, or cause instability. In these cases, you may need to disable automatic configuration for that type of device, and take additional steps to manually configure the device after the installation process is complete.

> **Check the Release Notes**
>
> Refer to the Release Notes for information on known issues with specific devices.

To override the automatic hardware detection, use one or more of the following options:

Compatibility	Option
Disable all hardware detection	`noprobe`
Disable graphics, keyboard, and mouse detection	`headless`
Disable passing keyboard and mouse information to stage 2 of the installation program	`nopass`
Disable automatic monitor detection (DDC)	`skipddc`
Use basic VESA driver for video	`xdriver=vesa`
Disable shell access on virtual console 2 during installation	`noshell`
Enable advanced programmable interrupt controller (APIC) support on Intel symmetic multi-processor (SMP) motherboards with this feature (for example, thos with the 440GX chipset)	`apic`
Disable mainboard APIC	`noapic`
Disable advanced configuration and power interface (ACPI)	`acpi=off`
Disable machine check exception (MCE) CPU self-diagnosis.	`nomce`
Disable non-uniform menory access on the AMD64 architecture	`numa-off`
Disable Direct Memory Access (DMA) for IDE, SATA, and ATAPI drives	`libata.dma=0`
Force kernel to detect a specific amount of memory, where xxx is a value in megabytes	`mem=xxxm`
Enable DMA only for IDE and SATA drives	`libata.dma=1`
Disable BIOS-assisted RAID	`nodmraid`
Force Firewire device detection	`firewire`
Disable Firewire device detection	`nofirewire`
Disable parallel port detection	`noparport`
Disable PC Card (PCMCIA) device detection	`nopcmcia`
Disable USB storage device detection	`nousbstorage`
Disable all USB device detection	`nousb`
Disable all probing of network hardware	`nonet`
Prompt user for ISA device configuration	`isa`

Table 9.4. Hardware Options

Additional Screen

The `isa` option causes the system to display an additional text screen at the beginning of the installation process. Use this screen to configure the ISA devices on your computer.

> **Important**
>
> Other kernel boot options have no particular meaning for **anaconda** and do not affect the installation process. However, if you use these options to boot the installation system, **anaconda** will preserve them in the bootloader configuration.

9.6. Using the Maintenance Boot Modes

9.6.1. Loading the Memory (RAM) Testing Mode

Faults in memory modules may cause your system to freeze or crash unpredictably. In some cases, memory faults may only cause errors with particular combinations of software. For this reason, you should test the memory of a computer before you install Fedora for the first time, even if it has previously run other operating systems.

Fedora includes the **Memtest86** memory testing application. To boot your computer in memory testing mode, choose **Memory test** at the boot menu. The first test starts immediately. By default, **Memtest86** carries out a total of ten tests.

To halt the tests and reboot your computer, enter `Esc` at any time.

9.6.2. Verifying boot media

You can test the integrity of an ISO-based installation source before using it to install Fedora. These sources include CD, DVD, and ISO images stored on a local hard drive or on an NFS server. Verifying that the ISO images are instace before you attempt an installation helps to avoid problems that are often encountered during installation.

Fedora offers you three ways to test installation ISOs:

- select the **Verify and Boot** option on the Fedora Live CD. To access the Live CD boot menu, press any key within ten seconds of the splash screen appearing.

- select **OK** at the prompt to test the media before installation when booting from the Fedora distribution CD set or DVD

- boot Fedora with the option `mediacheck` option.

9.6.3. Booting Your Computer with the Rescue Mode

You may boot a command-line Linux system from either a rescue disc or an installation disc, without installing Fedora on the computer. This enables you to use the utilities and functions of a running Linux system to modify or repair systems that are already installed on your computer.

The rescue disc starts the rescue mode system by default. To load the rescue system with the installation disc, choose **Rescue installed system** from the boot menu.

Specify the language, keyboard layout and network settings for the rescue system with the screens that follow. The final setup screen configures access to the existing system on your computer.

By default, rescue mode attaches an existing operating system to the rescue system under the directory **/mnt/sysimage/**.

9.6.4. Upgrading your computer

A previous boot option, `upgrade`, has been superceded by a stage in the installation process where the installation program prompts you to upgrade or reinstall earlier versions of Fedora that it detects on your system.

However, the installation program may not correctly detect a previous version of Fedora if the contents of the **/etc/fedora-release** file have changed. The boot option `upgradeany` relaxes the test that the installation program performs and allows you to upgrade a Fedora installation that the installation program has not correctly identified.

Installing Without Media

This section discusses how to install Fedora to your system without making any additional physical media. Instead, you can use your existing **GRUB** boot loader to start the installation program.

> ### Linux Required
> This procedure assumes you are already using Fedora or another relatively modern Linux distribution, and the **GRUB** boot loader. It also assumes you are a somewhat experienced Linux user.

10.1. Retrieving Boot Files

To perform an installation without media or a PXE server, your system must have two files stored locally, a kernel and an initial RAM disk.

1. Download a Live image or DVD distribution, or to locate an installation mirror, visit *http://mirrors.fedoraproject.org/publiclist/Fedora/11/*.

2. Locate the `isolinux/` folder using one of the following methods:

   - If you chose to download an image, open it with the appropriate desktop tool. If you are using Fedora, double-click the file to open it with the **Archive Manager**. Open the `isolinux/` folder.

   - If you chose not to download a whole image because you wish to install via the network, locate the desired release. In general, once you find a suitable mirror, browse to the `releases/11/Fedora/`*arch*`/os/isolinux/` folder.

   > ### Installation Types Available
   > If you download an image, you may then choose a hard disk-based installation or a network installation. If you only download selected files from a mirror, you may only perform a network installation.

3. Copy the `vmlinuz` and `initrd.img` files from the chosen source to the `/boot/` directory, renaming them to `vmlinuz-install` and `initrd.img-install`You must have root privileges to write files into the `/boot/` directory.

10.2. Editing the GRUB Configuration

The **GRUB** boot loader uses the configuration file `/boot/grub/grub.conf`. To configure **GRUB** to boot from the new files, add a boot stanza to `/boot/grub/grub.conf`that refers to them.

A minimal boot stanza looks like the following listing:

```
title Installation
        root (hd0,0)
        kernel /vmlinuz-install
        initrd /initrd.img-install
```

You may wish to add options to the end of the `kernel` line of the boot stanza. These options set preliminary options in **Anaconda** which the user normally sets interactively. For a list of available installer boot options, refer to *Chapter 9, Boot Options*.

The following options are generally useful for medialess installations:

- `ip=`

- `method=`

- `lang=`

- `keymap=`

- `ksdevice=` (if installation requires an interface other than eth0)

- `vnc` and `vncpassword=` for a remote installation

When you are finished, change the `default` option in **/boot/grub/grub.conf** to point to the new first stanza you added:

```
default 0
```

10.3. Booting to Installation

Reboot the system. **GRUB** boots the installation kernel and RAM disk, including any options you set. You may now refer to the appropriate chapter in this guide for the next step. If you chose to install remotely using VNC, refer to *Section 9.2, "Enabling Remote Access to the Installation System"* for assistance in connecting to the remote system.

Setting Up an Installation Server

> **Experience Required**
>
> This appendix is intended for users with previous Linux experience. If you are a new user, you may want to install using minimal boot media or the distribution DVD instead.

> **Warning**
>
> The instructions in this appendix configures an automated install server. The default configuration includes destruction of all existing data on all disks for hosts that install using this method. This is often different from other network install server configurations which may provide for an interactive installation experience.

Fedora allows for installation over a network using the NFS, FTP, or HTTP protocols. A network installation can be started from a boot CD-ROM, a bootable flash memory drive, or by using the **askmethod** boot option with the Fedora CD #1 or DVD. Alternatively, if the system to be installed contains a network interface card (NIC) with Pre-Execution Environment (PXE) support, it can be configured to boot from files on another networked system rather than local media such as a CD-ROM.

For a PXE network installation, the client's NIC with PXE support sends out a broadcast request for DHCP information. The DHCP server provides the client with an IP address, other network information such as name server, the IP address or hostname of the **tftp** server (which provides the files necessary to start the installation program), and the location of the files on the **tftp** server. This is possible because of PXELINUX, which is part of the **syslinux** package.

In the past, administrators needed to perform a great deal of manual configuration to produce an installation server. However, if you have a Red Hat Enterprise Linux, CentOS, or Fedora server on your local network, you can use *cobbler* to perform these tasks. To configure a PXE server manually, see *Section 11.5, "Manually configure a PXE server"*.

To perform the tasks in this section, switch to the root account with the command **su -**. As an alternative, you can run a command with the -c option, using the form **su -c 'command'**.

11.1. Setting Up *cobbler*

To install *cobbler* use the following command:

```
yum -y install cobbler
```

The **cobbler** command can check its own settings for validity and report the results. Run the following command to check the settings:

```
cobbler check
```

Change the settings in the **/var/lib/cobbler/settings** file to reflect the IP address information for the server. You must change at least the server and next_server options, although these options may point to the same IP address.

If you are not already running a DHCP server, you should also change the `manage_dhcp` option to `1`. If you are running a DHCP server, configure it according to the instructions found in the *syslinux* package documentation. For more information, refer to your local files **/usr/share/doc/syslinux-*version*/syslinux.doc** and **/usr/share/doc/syslinux-*version*/pxelinux.doc**.

11.2. Setting Up the Distribution

To set up a distribution from a full Fedora DVD or ISO image, use this procedure.

> **Network Locations**
>
> To create a local mirror from an existing network source, skip this section and refer instead to *Section 11.3, "Mirroring a Network Location"*.

1. If you are using a DVD disc or ISO image, Create a directory mount point:

    ```
    mkdir /mnt/dvd
    ```

    To mount a physical DVD disc, use the following command:

    ```
    mount -o context=system_u:object_r:httpd_sys_content_t:s0 /dev/dvd /
    mnt/dvd
    ```

    To mount a DVD ISO image, use the following command:

    ```
    mount -ro loop,context=system_u:object_r:httpd_sys_content_t:s0 /path/
    to/image.iso /mnt/dvd
    ```

2. To support NFS installation, create a file **/etc/exports** and add the following line to it:

    ```
    /mnt/dvd *(ro,async)
    ```

    Start the NFS server using the following commands:

    ```
    /sbin/service rpcbind start /sbin/service nfs start
    ```

3. To support HTTP installation, use **yum** to install the Apache web server if it is not already installed:

    ```
    yum -y install httpd
    ```

    Make a link to the mounted disc in the Apache public content area:

    ```
    ln -s /mnt/dvd /var/www/html/distro
    ```

11.3. Mirroring a Network Location

If you do not have discs or ISO images for a distribution, you can use **cobbler** to create an installation server. The **cobbler** command can fetch the distribution over the network as part of the import process.

Locate the distribution on the network. The location may be on the local network or reached at a remote site via FTP, HTTP, or rsync protocols. Note the URI, which will be in one of the following forms:

- `http://mirror.example.com/pub/fedora/linux/releases/11/Fedora/`*arch*`/os`

- `ftp://mirror.example.com/pub/fedora/linux/releases/11/Fedora/`*arch*`/os`

- `rsync://mirror.example.com/fedora/linux/releases/11/Fedora/`*arch*`/os`

11.4. Importing the Distribution

To offer a distribution through more than one installation method, perform additional **cobbler import** tasks using a different name for each method. For best results, use the installation method as part of the name, so it appears in the client's boot menu.

1. To import the DVD disc or ISO distribution into **cobbler**, run this command:

   ```
   cobbler import --path=/mnt/dvd --name=distro_name
   ```

   For `distro_name`, substitute a meaningful name for the distribution.

   To import a local or remote network distribution into **cobbler**, run this command. Replace `network_URI` with the URI you found in *Section 11.3, "Mirroring a Network Location"*, and `distro_name` as above:

   ```
   cobbler import --mirror=network_URI --name=distro_name
   ```

   > ⭐ **Importing a Source**
   >
   > When **cobbler** imports a distribution with the commands above, it copies all the files to the server's local storage, which may take some time.

   If you do not want to make local copies of the distribution because clients can already reach its location, use the `--available-as` option.

   ```
   cobbler import --path=/mnt/dvd --name=distro_name --available-
   as=network_URI
   cobbler import --mirror=network_URI --name=distro_name --available-
   as=network_URI
   ```

   For `nework_URI`, substitute the appropriate network location of the distribution. This URI indicates how the server makes the distribution available to its clients. The examples above assume that your **cobbler** server reaches the mirror location at the same URI as the clients. If not, substitute an appropriate URI for the `--mirror` option. The following examples are URI

locations that work if you have been following the procedures in this section, and your server's IP address is `192.168.1.1`:

- `nfs://`*`192.168.1.1`*`:/mnt/dvd`

- `http://`*`192.168.1.1`*`:/distro`

If necessary, replace *`192.168.1.1`* with the IP address for your **cobbler** server.

2. Run the command **cobbler sync** to apply the changes. To check that your **cobbler** server is listening on the correct ports, use the **netstat -lp** command.

> ### Firewall Considerations
>
> Depending on your server's configuration, you may need to use the **system-config-securitylevel** command to permit access to some or all of these network services:
>
> - 67 or bootps, for the DHCP/BOOTP server
>
> - 69 or tftp, for providing the PXE loader
>
> - 80 or http, if the **cobbler** server is to provide HTTP installation service
>
> - 20 and 21 or ftp, if the **cobbler** server is to provide FTP installation service
>
> - 111 or sunrpc, if the **cobbler** server is to provide NFS installation service

11.5. Manually configure a PXE server

The following steps must be performed to prepare for a PXE installation:

1. Configure the network (NFS, FTP, HTTP) server to export the installation tree.

2. Configure the files on the **tftp** server necessary for PXE booting.

3. Configure which hosts are allowed to boot from the PXE configuration.

4. Start the **tftp** service.

5. Configure DHCP.

6. Boot the client, and start the installation.

11.5.1. Setting up the Network Server

First, configure an NFS, FTP, or HTTP server to export the entire installation tree for the version and variant of Fedora to be installed. Refer to *Section 3.5, "Preparing for a Network Installation"* for detailed instructions.

11.5.2. PXE Boot Configuration

The next step is to copy the files necessary to start the installation to the **tftp** server so they can be found when the client requests them. The **tftp** server is usually the same server as the network server exporting the installation tree.

To copy these files, run the **Network Booting Tool** on the NFS, FTP, or HTTP server. A separate PXE server is not necessary.

11.5.2.1. Command Line Configuration

If the network server is not running X, the **pxeos** command line utility, which is part of the **system-config-netboot** package, can be used to configure the **tftp** server files as described in *Section 11.5.4, "TFTPD"*:

```
pxeos -a -i "<description>" -p <NFS|HTTP|FTP> -D 0 -s client.example.com
 \
-L <net-location> -k <kernel> -K <kickstart> <os-identifer>
```

The following list explains the options:

- -a — Specifies that an OS instance is being added to the PXE configuration.

- -i "*<description>*" — Replace "*<description>*" with a description of the OS instance.

- -p <NFS|HTTP|FTP> — Specify which of the NFS, FTP, or HTTP protocols to use for installation. Only one may be specified.

- -D <0|1> — Specify "**0**" which indicates that it is *not* a diskless configuration since **pxeos** can be used to configure a diskless environment as well.

- -s *client.example.com* — Provide the name of the NFS, FTP, or HTTP server after the -s option.

- -L *<net-location>* — Provide the location of the installation tree on that server after the -L option.

- -k *<kernel>* — Provide the specific kernel version of the server installation tree for booting.

- -K *<kickstart>* — Provide the location of the kickstart file, if available.

- *<os-identifer>* — Specify the OS identifier, which is used as the directory name in the **/tftpboot/linux-install/** directory.

If FTP is selected as the installation protocol and anonymous login is not available, specify a username and password for login, with the following options before *<os-identifer>* in the previous command:

```
-A 0 -u <username> -p <password>
```

For more information on command line options available for the **pxeos** command, refer to the **pxeos** man page.

11.5.3. Adding PXE Hosts

After configuring the network server, the interface as shown in *Figure 11.1, "Add Hosts"* is displayed.
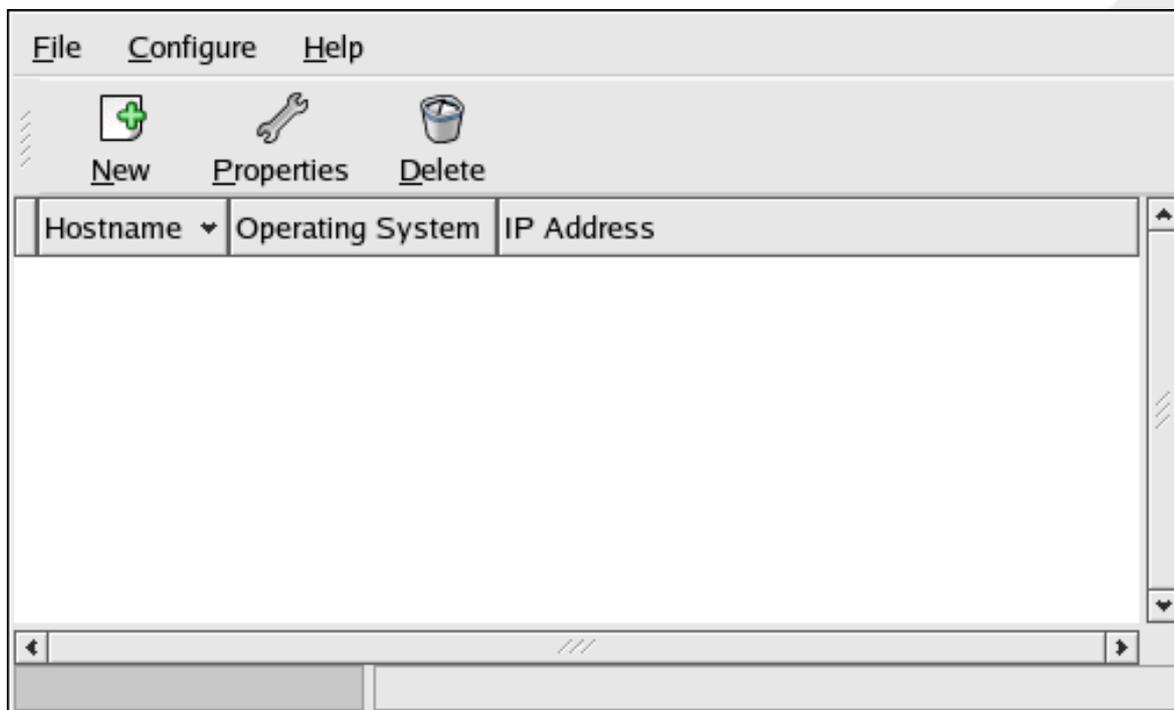
Figure 11.1. Add Hosts

The next step is to configure which hosts are allowed to connect to the PXE boot server. For the command line version of this step, refer to *Section 11.5.3.1, "Command Line Configuration"*.

To add hosts, click the **New** button.

Figure 11.2. Add a Host

Enter the following information:

- **Hostname or IP Address/Subnet** — The IP address, fully qualified hostname, or a subnet of systems that should be allowed to connect to the PXE server for installations.

- **Operating System** — The operating system identifier to install on this client. The list is populated from the network install instances created from the **Network Installation Dialog**.

- **Serial Console** — This option allows use of a serial console.

- **Kickstart File** — The location of a kickstart file to use, such as **http://server.example.com/kickstart/ks.cfg**. This file can be created with the **Kickstart Configurator**. Refer to *Chapter 14, Kickstart Configurator* for details.

Ignore the **Snapshot name** and **Ethernet** options. They are only used for diskless environments.

11.5.3.1. Command Line Configuration

If the network server is not running X, the **pxeboot** utility, a part of the **system-config-netboot** package, can be used to add hosts which are allowed to connect to the PXE server:

```
pxeboot -a -K <kickstart> -O <os-identifier> -r <value> <host>
```

The following list explains the options:

- `-a` — Specifies that a host is to be added.

- `-K <kickstart>` — The location of the kickstart file, if available.

- `-O <os-identifier>` — Specifies the operating system identifier as defined in *Section 11.5.2, "PXE Boot Configuration"*.

- `-r <value>` — Specifies the ram disk size.

- `<host>` — Specifies the IP address or hostname of the host to add.

For more information on command line options available for the **pxeboot** command, refer to the **pxeboot** man page.

11.5.4. TFTPD

11.5.4.1. Starting the **tftp** Server

On the DHCP server, verify that the **tftp-server** package is installed with the command **rpm -q tftp-server**.

tftp is an xinetd-based service; start it with the following commands:

```
/sbin/chkconfig --level 345 xinetd on /sbin/chkconfig --level 345 tftp on
```

These commands configure the **tftp** and **xinetd** services to immediately turn on and also configure them to start at boot time in runlevels 3, 4, and 5.

11.5.5. Configuring the DHCP Server

If a DHCP server does not already exist on the network, configure one. Refer to the Red Hat Enterprise Linux Deployment Guide for details. Make sure the configuration file contains the following so that PXE booting is enabled for systems which support it:

```
allow booting; allow bootp; class "pxeclients" { match if substring(option
vendor-class-identifier, 0, 9) = "PXEClient"; next-server <server-ip>;
filename "linux-install/pxelinux.0"; }
```

where the next-server `<server-ip>` should be replaced with the IP address of the `tftp` server.

11.5.6. Adding a Custom Boot Message

Optionally, modify `/tftpboot/linux-install/msgs/boot.msg` to use a custom boot message.

11.5.7. Performing the PXE Installation

For instructions on how to configure the network interface card with PXE support to boot from the network, consult the documentation for the NIC. It varies slightly per card.

After the system boots the installation program, refer to the *Chapter 7, Installing on Intel and AMD Systems*.

Installing Through VNC

Many enterprise customers perform interactive installations on systems in datacenters. These systems are often, but not always, installed in a rack environment and do not have a display, keyboard, or mouse. Additionally, a lot of these systems even lack the ability to connect a graphical display. Given that enterprise hardware rarely needs that ability at the physical system, this hardware configuration is acceptable.

The **Red Hat Enterprise Linux** (RHEL) and **Fedora Linux** installer (**anaconda**) offers users two interactive modes of operation. The original mode is a text-based interface. The newer mode uses GTK+ and runs in the X Window environment. The purpose of this document is to explain how the graphical installation mode can be used in enterprise environments, even when the system lacks a proper display and input devices typically associated with a workstation.

The primary driver behind the document is to encourage use of the graphical installer, even in enterprise environments. The text mode environment lacks a lot of capabilities found in the graphical mode. Many users still feel that the text mode interface provides them with additional power or configuration ability not found in the graphical version. The opposite is true. Much less development effort is put in to the text mode environment and specific things (e.g., LVM configuration) are deliberately left out of the text mode environment. The reasons for this are:

- Less screen real estate for creating user interfaces similar to those found in the graphical mode.

- Difficult internationalization support.

- Desire to maintain a single interactive installation code path.

All of these reasons and more are why you, the enterprise customer, should be making using of the **Virtual Network Computing** (VNC) mode offered in **anaconda**. VNC allows the graphical mode of the installer to run locally, but display on a system connected to the network.

12.1. VNC Viewer

Performing a VNC installation requires a VNC viewer running on your workstation or other terminal computer. Locations where you might want a VNC viewer installed:

- Your workstation

- Laptop on a datacenter crash cart

VNC is open source software licensed under the GNU General Public License. Versions exist for Linux, Windows, and MacOS X. Here are some recommended VNC viewers:

- `vncviewer` is available on **Red Hat Enterprise Linux** and **Fedora Linux** by installing the `vnc` package:

  ```
  # yum install vnc
  ```

- TightVNC is available for Windows at *http://www.tightvnc.com/*

- MacOS X includes built-in VNC support as of version 10.5. In the **Finder**, click the **Go** menu and choose **Connect to Server**. In the server address field, you can type `vnc://SERVER:DISPLAY`,

where SERVER is the IP address or DNS host name of the VNC server you wish to connect to and DISPLAY is the VNC display number (usually 1), and click **Connect**.

Once you have verified you have a VNC viewer available, it's time to start the installation.

12.2. VNC Modes in Anaconda

Anaconda offers two modes for VNC installation. The mode you select will depend on the network configuration in your environment.

12.2.1. Direct Mode

Direct mode VNC in anaconda is when the client initiates a connection to the VNC server running in anaconda. Anaconda will tell you when to initiate this connection in the VNC viewer. Direct mode can be activated by either of the following commands:

- Specify `vnc` as a boot argument.

- Specify the `vnc` command in the kickstart file used for installation.

When you activate VNC mode, anaconda will complete the first stage of the installer and then start VNC to run the graphical installer. The installer will display a message on the console in the following format:

```
Running anaconda VERSION, the PRODUCT system installer - please wait...
```

Anaconda will also tell you the IP address and display number to use in your VNC viewer. At this point, you need to start the VNC viewer and connect to the target system to continue the installation. The VNC viewer will present anaconda to you in graphical mode.

There are some disadvantages to direct mode, including:

- Requires visual access to the system console to see the IP address and port to connect the VNC viewer to.

- Requires interactive access to the system console to complete the first stage of the installer.

If either of these disadvantages would prevent you from using direct mode VNC in anaconda, then connect mode is probably more suited to your environment.

12.2.2. Connect Mode

Certain firewall configurations or instances where the target system is configured to obtain a dynamic IP address may cause trouble with the direct VNC mode in anaconda. In addition, if you lack a console on the target system to see the message that tells you the IP address to connect to, then you will not be able to continue the installation.

The VNC connect mode changes how VNC is started. Rather than anaconda starting up and waiting for you to connect, the VNC connect mode allows anaconda to automatically connect to your view. You won't need to know the IP address of the target system in this case.

To activate the VNC connect mode, pass the `vncconnect` boot parameter:

```
boot: linux vncconnect=HOST
```

Replace HOST with your VNC viewer's IP address or DNS host name. Before starting the installation process on the target system, start up your VNC viewer and have it wait for an incoming connection.

Start the installation and when your VNC viewer displays the graphical installer, you are ready to go.

12.3. Installation Using VNC

Now that you have installed a VNC viewer application and selected a VNC mode for use in anaconda, you are ready to begin the installation.

12.3.1. Installation Example

The easiest way to perform an installation using VNC is to connect another computer directly to the network port on the target system. The laptop on a datacenter crash cart usually fills this role. If you are performing your installation this way, make sure you follow these steps:

1. Connect the laptop or other workstation to the target system using a crossover cable. If you are using regular patch cables, make sure you connect the two systems using a small hub or switch. Most recent Ethernet interfaces will automatically detect if they need to be crossover or not, so it may be possible to connect the two systems directly using a regular patch cable.

2. Configure the VNC viewer system to use a RFC 1918 address with no gateway. This private network connection will only be used for the purpose of installation. Configure the VNC viewer system to be 192.168.100.1/24. If that address is in use, just pick something else in the RFC 1918 address space that is available to you.

3. Start the RHEL installation on the target system.

   a. Booting the installation DVD or CD.

      If booting the installation media (CD or DVD), make sure **vnc** is passed as a boot parameter. To add the **vnc** parameter, you will need a console attached to the target system that allows you to interact with the boot process. Enter the following at the prompt:

      ```
      boot: linux vnc
      ```

   b. Boot over the network.

      If the target system is configured with a static IP address, add the **vnc** command to the kickstart file. If the target system is using DHCP, add **vncconnect=HOST** to the boot arguments for the target system. HOST is the IP address or DNS host name of the VNC viewer system. Enter the following at the prompt:

      ```
      boot: linux vncconnect=HOST
      ```

4. When prompted for the network configuration on the target system, assign it an available RFC 1918 address in the same network you used for the VNC viewer system. For example, 192.168.100.2/24.

> **Note**
>
> This IP address is only used during installation. You will have an opportunity to configure the final network settings, if any, later in the installer.

5. Once the installer indicates it is starting anaconda, you will be instructed to connect to the system using the VNC viewer. Connect to the viewer and follow the graphical installation mode instructions found in the product documentation.

12.3.2. Kickstart Considerations

If your target system will be booting over the network, VNC is still available. Just add the `vnc` command to the kickstart file for the system. You will be able to connect to the target system using your VNC viewer and monitor the installation progress. The address to use is the one the system is configured with via the kickstart file.

If you are using DHCP for the target system, the reverse `vncconnect` method may work better for you. Rather than adding the `vnc` boot parameter to the kickstart file, add the `vncconnect=HOST` parameter to the list of boot arguments for the target system. For HOST, put the IP address or DNS host name of the VNC viewer system. See the next section for more details on using the vncconnect mode.

12.3.3. Firewall Considerations

If you are performing the installation where the VNC viewer system is a workstation on a different subnet from the target system, you may run in to network routing problems. VNC works fine so long as your viewer system has a route to the target system and ports 5900 and 5901 are open. If your environment has a firewall, make sure ports 5900 and 5901 are open between your workstation and the target system.

In addition to passing the `vnc` boot parameter, you may also want to pass the `vncpassword` parameter in these scenarios. While the password is sent in plain text over the network, it does provide an extra step before a viewer can connect to a system. Once the viewer connects to the target system over VNC, no other connections are permitted. These limitations are usually sufficient for installation purposes.

> **Important**
>
> Be sure to use a temporary password for the `vncpassword` option. It should not be a password you use on any systems, especially a real root password.

If you continue to have trouble, consider using the `vncconnect` parameter. In this mode of operation, you start the viewer on your system first telling it to listen for an incoming connection. Pass `vncconnect=HOST` at the boot prompt and the installer will attempt to connect to the specified HOST (either a hostname or IP address).

12.4. References

- VNC description at Wikipedia: *http://en.wikipedia.org/wiki/Vnc*

- TightVNC: *http://www.tightvnc.com/*

- RFC 1918 - Address Allocation for Private Networks: *http://www.ietf.org/rfc/rfc1918.txt*

- Anaconda boot options: *http://fedoraproject.org/wiki/Anaconda/Options*

- Kickstart documentation: *http://fedoraproject.org/wiki/Anaconda/Kickstart*

Kickstart Installations

13.1. What are Kickstart Installations?

Many system administrators would prefer to use an automated installation method to install Fedora on their machines. To answer this need, Red Hat created the kickstart installation method. Using kickstart, a system administrator can create a single file containing the answers to all the questions that would normally be asked during a typical installation.

Kickstart files can be kept on a single server system and read by individual computers during the installation. This installation method can support the use of a single kickstart file to install Fedora on multiple machines, making it ideal for network and system administrators.

Kickstart provides a way for users to automate a Fedora installation.

13.2. How Do You Perform a Kickstart Installation?

Kickstart installations can be performed using a local CD-ROM, a local hard drive, or via NFS, FTP, or HTTP.

To use kickstart, you must:

1. Create a kickstart file.

2. Create a boot media with the kickstart file or make the kickstart file available on the network.

3. Make the installation tree available.

4. Start the kickstart installation.

This chapter explains these steps in detail.

13.3. Creating the Kickstart File

The kickstart file is a simple text file, containing a list of items, each identified by a keyword. You can create it by using the **Kickstart Configurator** application, or writing it from scratch. The Fedora installation program also creates a sample kickstart file based on the options that you selected during installation. It is written to the file `/root/anaconda-ks.cfg`. You should be able to edit it with any text editor or word processor that can save files as ASCII text.

First, be aware of the following issues when you are creating your kickstart file:

- Sections must be specified *in order*. Items within the sections do not have to be in a specific order unless otherwise specified. The section order is:

  - Command section — Refer to *Section 13.4, "Kickstart Options"* for a list of kickstart options. You must include the required options.

  - The `%packages` section — Refer to *Section 13.5, "Package Selection"* for details.

  - The `%pre` and `%post` sections — These two sections can be in any order and are not required. Refer to *Section 13.6, "Pre-installation Script"* and *Section 13.7, "Post-installation Script"* for details.

- Items that are not required can be omitted.

- Omitting any required item results in the installation program prompting the user for an answer to the related item, just as the user would be prompted during a typical installation. Once the answer is given, the installation continues unattended (unless it finds another missing item).

- Lines starting with a pound (also known as hash) sign (#) are treated as comments and are ignored.

- For kickstart *upgrades*, the following items are required:

  - Language

  - Installation method

  - Device specification (if device is needed to perform the installation)

  - Keyboard setup

  - The **upgrade** keyword

  - Boot loader configuration

  If any other items are specified for an upgrade, those items are ignored (note that this includes package selection).

13.4. Kickstart Options

The following options can be placed in a kickstart file. If you prefer to use a graphical interface for creating your kickstart file, use the **Kickstart Configurator** application. Refer to *Chapter 14, Kickstart Configurator* for details.

> **Note**
>
> If the option is followed by an equals mark (=), a value must be specified after it. In the example commands, options in brackets ([]) are optional arguments for the command.

autopart (optional)

Automatically create partitions — 1 GB or more root (`/`) partition, a swap partition, and an appropriate boot partition for the architecture. One or more of the default partition sizes can be redefined with the **part** directive.

- **--encrypted** — Should all devices with support be encrypted by default? This is equivalent to checking the **Encrypt** checkbox on the initial partitioning screen.

- **--passphrase=** — Provide a default system-wide passphrase for all encrypted devices.

ignoredisk (optional)

Causes the installer to ignore the specified disks. This is useful if you use autopartition and want to be sure that some disks are ignored. For example, without **ignoredisk**, attempting to deploy on a SAN-cluster the kickstart would fail, as the installer detects passive paths to the SAN that return no partition table.

The **ignoredisk** option is also useful if you have multiple paths to your disks.

The syntax is:

```
ignoredisk --drives=drive1,drive2,...
```

where *driveN* is one of **sda**, **sdb**,..., **hda**,... etc.

autostep (optional)

Similar to **interactive** except it goes to the next screen for you. It is used mostly for debugging.

- **--autoscreenshot** — Take a screenshot at every step during installation and copy the images over to **/root/anaconda-screenshots** after installation is complete. This is most useful for documentation.

auth or **authconfig** (required)

Sets up the authentication options for the system. It is similar to the **authconfig** command, which can be run after the install. By default, passwords are normally encrypted and are not shadowed.

- **--enablemd5** — Use md5 encryption for user passwords.

- **--enablenis** — Turns on NIS support. By default, **--enablenis** uses whatever domain it finds on the network. A domain should almost always be set by hand with the **--nisdomain=** option.

- **--nisdomain=** — NIS domain name to use for NIS services.

- **--nisserver=** — Server to use for NIS services (broadcasts by default).

- **--useshadow** or **--enableshadow** — Use shadow passwords.

- **--enableldap** — Turns on LDAP support in **/etc/nsswitch.conf**, allowing your system to retrieve information about users (UIDs, home directories, shells, etc.) from an LDAP directory. To use this option, you must install the **nss_ldap** package. You must also specify a server and a base DN (distinguished name) with **--ldapserver=** and **--ldapbasedn=**.

- **--enableldapauth** — Use LDAP as an authentication method. This enables the **pam_ldap** module for authentication and changing passwords, using an LDAP directory. To use this option, you must have the **nss_ldap** package installed. You must also specify a server and a base DN with **--ldapserver=** and **--ldapbasedn=**.

- **--ldapserver=** — If you specified either **--enableldap** or **--enableldapauth**, use this option to specify the name of the LDAP server to use. This option is set in the **/etc/ldap.conf** file.

- **--ldapbasedn=** — If you specified either **--enableldap** or **--enableldapauth**, use this option to specify the DN in your LDAP directory tree under which user information is stored. This option is set in the **/etc/ldap.conf** file.

- **--enableldaptls** — Use TLS (Transport Layer Security) lookups. This option allows LDAP to send encrypted usernames and passwords to an LDAP server before authentication.

- **--enablekrb5** — Use Kerberos 5 for authenticating users. Kerberos itself does not know about home directories, UIDs, or shells. If you enable Kerberos, you must make users' accounts

known to this workstation by enabling LDAP, NIS, or Hesiod or by using the **/usr/sbin/ useradd** command. If you use this option, you must have the **pam_krb5** package installed.

- **--krb5realm=** — The Kerberos 5 realm to which your workstation belongs.

- **--krb5kdc=** — The KDC (or KDCs) that serve requests for the realm. If you have multiple KDCs in your realm, separate their names with commas (,).

- **--krb5adminserver=** — The KDC in your realm that is also running kadmind. This server handles password changing and other administrative requests. This server must be run on the master KDC if you have more than one KDC.

- **--enablehesiod** — Enable Hesiod support for looking up user home directories, UIDs, and shells. More information on setting up and using Hesiod on your network is in **/usr/share/ doc/glibc-2.x.x/README.hesiod**, which is included in the **glibc** package. Hesiod is an extension of DNS that uses DNS records to store information about users, groups, and various other items.

- **--hesiodlhs** — The Hesiod LHS ("left-hand side") option, set in **/etc/hesiod.conf**. This option is used by the Hesiod library to determine the name to search DNS for when looking up information, similar to LDAP's use of a base DN.

- **--hesiodrhs** — The Hesiod RHS ("right-hand side") option, set in **/etc/hesiod.conf**. This option is used by the Hesiod library to determine the name to search DNS for when looking up information, similar to LDAP's use of a base DN.

> **Note**
>
> To look up user information for "jim", the Hesiod library looks up *jim.passwd<LHS><RHS>*, which should resolve to a TXT record that looks like what his passwd entry would look like (`jim:*:501:501:Jungle Jim:/home/jim:/bin/bash`). For groups, the situation is identical, except *jim.group<LHS><RHS>* would be used.
>
> Looking up users and groups by number is handled by making "501.uid" a CNAME for "jim.passwd", and "501.gid" a CNAME for "jim.group". Note that the library does not place a period `.` in front of the LHS and RHS values when performing a search. Therefore the LHS and RHS values need to have a period placed in front of them in order if they require this.

- **--enablesmbauth** — Enables authentication of users against an SMB server (typically a Samba or Windows server). SMB authentication support does not know about home directories, UIDs, or shells. If you enable SMB, you must make users' accounts known to the workstation by enabling LDAP, NIS, or Hesiod or by using the **/usr/sbin/useradd** command to make their accounts known to the workstation. To use this option, you must have the **pam_smb** package installed.

- **--smbservers=** — The name of the server(s) to use for SMB authentication. To specify more than one server, separate the names with commas (,).

- **--smbworkgroup=** — The name of the workgroup for the SMB servers.

- **--enablecache** — Enables the **nscd** service. The **nscd** service caches information about users, groups, and various other types of information. Caching is especially helpful if you choose to distribute information about users and groups over your network using NIS, LDAP, or hesiod.

bootloader (required)

Specifies how the boot loader should be installed. This option is required for both installations and upgrades.

> **Important**
>
> If you select text mode for a kickstart installation, make sure that you specify choices for the partitioning, bootloader, and package selection options. These steps are automated in text mode, and **anaconda** cannot prompt you for missing information. If you do not provide choices for these options, **anaconda** will stop the installation process.

- **--append=** — Specifies kernel parameters. To specify multiple parameters, separate them with spaces. For example:

```
bootloader --location=mbr --append="hdd=ide-scsi ide=nodma"
```

- **--driveorder** — Specify which drive is first in the BIOS boot order. For example:

```
bootloader --driveorder=sda,hda
```

- **--location=** — Specifies where the boot record is written. Valid values are the following: **mbr** (the default), **partition** (installs the boot loader on the first sector of the partition containing the kernel), or **none** (do not install the boot loader).

- **--password=** — If using GRUB, sets the GRUB boot loader password to the one specified with this option. This should be used to restrict access to the GRUB shell, where arbitrary kernel options can be passed.

- **--md5pass=** — If using GRUB, similar to **--password=** except the password should already be encrypted.

- **--upgrade** — Upgrade the existing boot loader configuration, preserving the old entries. This option is only available for upgrades.

clearpart (optional)

Removes partitions from the system, prior to creation of new partitions. By default, no partitions are removed.

> **Note**
>
> If the **clearpart** command is used, then the **--onpart** command cannot be used on a logical partition.

- **--all** — Erases all partitions from the system.

- **--drives=** — Specifies which drives to clear partitions from. For example, the following clears all the partitions on the first two drives on the primary IDE controller:

```
clearpart --drives=hda,hdb --all
```

- **--initlabel** — Initializes the disk label to the default for your architecture (for example **msdos** for x86 and **gpt** for Itanium). It is useful so that the installation program does not ask if it should initialize the disk label if installing to a brand new hard drive.

- **--linux** — Erases all Linux partitions.

- **--none** (default) — Do not remove any partitions.

cmdline (optional)

Perform the installation in a completely non-interactive command line mode. Any prompts for interaction halts the install. This mode is useful on IBM System z systems with the x3270 console.

device (optional)

On most PCI systems, the installation program autoprobes for Ethernet and SCSI cards properly. On older systems and some PCI systems, however, kickstart needs a hint to find the proper devices. The **device** command, which tells the installation program to install extra modules, is in this format:

```
device <type> <moduleName> --opts=<options>
```

- *<type>* — Replace with either **scsi** or **eth**.

- *<moduleName>* — Replace with the name of the kernel module which should be installed.

- **--opts=** — Mount options to use for mounting the NFS export. Any options that can be specified in **/etc/fstab** for an NFS mount are allowed. The options are listed in the **nfs(5)** man page. Multiple options are separated with a comma.

driverdisk (optional)

Driver diskettes can be used during kickstart installations. You must copy the driver diskettes's contents to the root directory of a partition on the system's hard drive. Then you must use the **driverdisk** command to tell the installation program where to look for the driver disk.

```
driverdisk <partition> [--type=<fstype>]
```

Alternatively, a network location can be specified for the driver diskette:

```
driverdisk --source=ftp://path/to/dd.img
driverdisk --source=http://path/to/dd.img
driverdisk --source=nfs:host:/path/to/img
```

- *<partition>* — Partition containing the driver disk.

- **--type=** — File system type (for example, vfat or ext2).

firewall (optional)

This option corresponds to the **Firewall Configuration** screen in the installation program:

```
firewall --enabled|--disabled [--trust=] <device> [--port=]
```

- **--enabled** or **--enable** — Reject incoming connections that are not in response to outbound requests, such as DNS replies or DHCP requests. If access to services running on this machine is needed, you can choose to allow specific services through the firewall.

- **--disabled** or **--disable** — Do not configure any iptables rules.

- **--trust=** — Listing a device here, such as eth0, allows all traffic coming from that device to go through the firewall. To list more than one device, use **--trust eth0 --trust eth1**. Do NOT use a comma-separated format such as **--trust eth0, eth1**.

- *<incoming>* — Replace with one or more of the following to allow the specified services through the firewall.

    - **--ssh**

    - **--telnet**

    - **--smtp**

    - **--http**

    - **--ftp**

- **--port=** — You can specify that ports be allowed through the firewall using the port:protocol format. For example, to allow IMAP access through your firewall, specify **imap:tcp**. Numeric ports can also be specified explicitly; for example, to allow UDP packets on port 1234 through, specify **1234:udp**. To specify multiple ports, separate them by commas.

firstboot (optional)

Determine whether the **Setup Agent** starts the first time the system is booted. If enabled, the **firstboot** package must be installed. If not specified, this option is disabled by default.

- **--enable** or **--enabled** — The **Setup Agent** is started the first time the system boots.

- **--disable** or **--disabled** — The **Setup Agent** is not started the first time the system boots.

- **--reconfig** — Enable the **Setup Agent** to start at boot time in reconfiguration mode. This mode enables the language, mouse, keyboard, root password, security level, time zone, and networking configuration options in addition to the default ones.

halt (optional)

Halt the system after the installation has successfully completed. This is similar to a manual installation, where anaconda displays a message and waits for the user to press a key before rebooting. During a kickstart installation, if no completion method is specified, the **reboot** option is used as default.

The **halt** option is roughly equivalent to the **shutdown -h** command.

For other completion methods, refer to the **poweroff**, **reboot**, and **shutdown** kickstart options.

graphical (optional)

Perform the kickstart installation in graphical mode. This is the default.

install (optional)

Tells the system to install a fresh system rather than upgrade an existing system. This is the default mode. For installation, you must specify the type of installation from **cdrom**, **harddrive**, **nfs**, or **url** (for FTP or HTTP installations). The **install** command and the installation method command must be on separate lines.

- **cdrom** — Install from the first CD-ROM drive on the system.

- **harddrive** — Install from a Red Hat installation tree on a local drive, which must be either vfat or ext2.

  - **--biospart=**

    BIOS partition to install from (such as 82).

  - **--partition=**

    Partition to install from (such as sdb2).

  - **--dir=**

    Directory containing the *variant* directory of the installation tree.

    For example:

    ```
    harddrive --partition=hdb2 --dir=/tmp/install-tree
    ```

- **nfs** — Install from the NFS server specified.

  - **--server=**

    Server from which to install (hostname or IP).

  - **--dir=**

    Directory containing the *variant* directory of the installation tree.

  - **--opts=**

    Mount options to use for mounting the NFS export. (optional)

    For example:

    ```
    nfs --server=nfsserver.example.com --dir=/tmp/install-tree
    ```

- **url** — Install from an installation tree on a remote server via FTP or HTTP.

  For example:

  ```
  url --url http://<server>/<dir>
  ```

  or:

```
url --url ftp://<username>:<password>@<server>/<dir>
```

interactive (optional)

Uses the information provided in the kickstart file during the installation, but allow for inspection and modification of the values given. You are presented with each screen of the installation program with the values from the kickstart file. Either accept the values by clicking **Next** or change the values and click **Next** to continue. Refer to the **autostep** command.

iscsi (optional)

issci --ipaddr= [options].

- **--target** —

- **--port=** —

- **--user=** —

- **--password=** —

key (optional)

Specify an installation key, which is needed to aid in package selection and identify your system for support purposes. This command is specific to Red Hat Enterprise Linux; it has no meaning for Fedora and will be ignored.

- **--skip** — Skip entering a key. Usually if the key command is not given, anaconda will pause at this step to prompt for a key. This option allows automated installation to continue if you do not have a key or do not want to provide one.

keyboard (required)

Sets system keyboard type. Here is the list of available keyboards on i386, Itanium, and Alpha machines:

```
be-latin1, bg, br-abnt2, cf, cz-lat2, cz-us-qwertz, de, de-latin1,
de-latin1-nodeadkeys, dk, dk-latin1, dvorak, es, et, fi, fi-latin1,
fr, fr-latin0, fr-latin1, fr-pc, fr_CH, fr_CH-latin1, gr, hu, hu101,
is-latin1, it, it-ibm, it2, jp106, la-latin1, mk-utf, no, no-latin1,
pl, pt-latin1, ro_win, ru, ru-cp1251, ru-ms, ru1, ru2,  ru_win,
se-latin1, sg, sg-latin1, sk-qwerty, slovene, speakup,  speakup-lt,
sv-latin1, sg, sg-latin1, sk-querty, slovene, trq, ua,  uk, us, us-
acentos
```

The file **/usr/lib/python2.2/site-packages/rhpl/keyboard_models.py** also contains this list and is part of the **rhpl** package.

lang (required)

Sets the language to use during installation and the default language to use on the installed system. For example, to set the language to English, the kickstart file should contain the following line:

```
lang en_US
```

The file **/usr/share/system-config-language/locale-list** provides a list of the valid language codes in the first column of each line and is part of the **system-config-language** package.

Certain languages (mainly Chinese, Japanese, Korean, and Indic languages) are not supported during text mode installation. If one of these languages is specified using the lang command, installation will continue in English though the running system will have the specified langauge by default.

langsupport (deprecated)

The langsupport keyword is deprecated and its use will cause an error message to be printed to the screen and installation to halt. Instead of using the langsupport keyword, you should now list the support package groups for all languages you want supported in the **%packages** section of your kickstart file. For instance, adding support for French means you should add the following to **%packages**:

```
@french-support
```

logvol (optional)

Create a logical volume for Logical Volume Management (LVM) with the syntax:

```
logvol <mntpoint> --vgname=<name> --size=<size> --
name=<name> <options>
```

The options are as follows:

- **--noformat** — Use an existing logical volume and do not format it.

- **--useexisting** — Use an existing logical volume and reformat it.

- **--fstype=** — Sets the file system type for the logical volume. Valid values are ext2, ext3, swap, and vfat.

- **--fsoptions=** — Specifies a free form string of options to be used when mounting the filesystem. This string will be copied into the **/etc/fstab** file of the installed system and should be enclosed in quotes.

- **--bytes-per-inode=** — Specifies the size of inodes on the filesystem to be made on the logical volume. Not all filesystems support this option, so it is silently ignored for those cases.

- **--grow=** — Tells the logical volume to grow to fill available space (if any), or up to the maximum size setting.

- **--maxsize=** — The maximum size in megabytes when the logical volume is set to grow. Specify an integer value here, and do not append the number with MB.

- **--recommended=** — Determine the size of the logical volume automatically.

- **--percent=** — Specify the size of the logical volume as a percentage of available space in the volume group.

Create the partition first, create the logical volume group, and then create the logical volume. For example:

```
part pv.01 --size 3000
volgroup myvg pv.01
logvol / --vgname=myvg --size=2000 --name=rootvol
```

logging (optional)

This command controls the error logging of anaconda during installation. It has no effect on the installed system.

- **--host=** — Send logging information to the given remote host, which must be running a syslogd process configured to accept remote logging.

- **--port=** — If the remote syslogd process uses a port other than the default, it may be specified with this option.

- **--level=** — One of debug, info, warning, error, or critical.

  Specify the minimum level of messages that appear on tty3. All messages will still be sent to the log file regardless of this level, however.

mediacheck (optional)

If given, this will force anaconda to run mediacheck on the installation media. This command requires that installs be attended, so it is disabled by default.

monitor (optional)

If the monitor command is not given, anaconda will use X to automatically detect your monitor settings. Please try this before manually configuring your monitor.

- **--hsync=** — Specifies the horizontal sync frequency of the monitor.

- **--monitor=** — Use specified monitor; monitor name should be from the list of monitors in /usr/share/hwdata/MonitorsDB from the hwdata package. The list of monitors can also be found on the X Configuration screen of the Kickstart Configurator. This is ignored if --hsync or --vsync is provided. If no monitor information is provided, the installation program tries to probe for it automatically.

- **--noprobe=** — Do not try to probe the monitor.

- **--vsync=** — Specifies the vertical sync frequency of the monitor.

mouse (deprecated)

The mouse keyword is deprecated.

network (optional)

Configures network information for the system. If the kickstart installation does not require networking (in other words, it is not installed over NFS, HTTP, or FTP), networking is not configured for the system. If the installation does require networking and network information is not provided in the kickstart file, the installation program assumes that the installation should be done over eth0 via a dynamic IP address (BOOTP/DHCP), and configures the final, installed system to determine its IP address dynamically. The **network** option configures networking information for kickstart installations via a network as well as for the installed system.

- **--bootproto=** — One of **dhcp**, **bootp**, or **static**.

  It defaults to **dhcp**. **bootp** and **dhcp** are treated the same.

  The DHCP method uses a DHCP server system to obtain its networking configuration. As you might guess, the BOOTP method is similar, requiring a BOOTP server to supply the networking configuration. To direct a system to use DHCP:

  ```
  network --bootproto=dhcp
  ```

  To direct a machine to use BOOTP to obtain its networking configuration, use the following line in the kickstart file:

  ```
  network --bootproto=bootp
  ```

  The static method requires that you enter all the required networking information in the kickstart file. As the name implies, this information is static and are used during and after the installation. The line for static networking is more complex, as you must include all network configuration information on one line. You must specify the IP address, netmask, gateway, and nameserver. For example: (the "\" indicates that this should be read as one continuous line):

  ```
  network --bootproto=static --ip=10.0.2.15 --netmask=255.255.255.0 \
  --gateway=10.0.2.254 --nameserver=10.0.2.1
  ```

  If you use the static method, be aware of the following two restrictions:

  - All static networking configuration information must be specified on *one* line; you cannot wrap lines using a backslash, for example.

  - You can also configure multiple nameservers here. To do so, specify them as a comma-delimited list in the command line. For example:

    ```
    network --bootproto=static --ip=10.0.2.15 --netmask=255.255.255.0 \
    --gateway=10.0.2.254 --nameserver 192.168.2.1,192.168.3.1
    ```

- **--device=** — Used to select a specific Ethernet device for installation. Note that using **--device=** is not effective unless the kickstart file is a local file (such as **ks=floppy**), since the installation program configures the network to find the kickstart file. For example:

  ```
  network --bootproto=dhcp --device=eth0
  ```

- **`--ip=`** — IP address for the machine to be installed.

- **`--gateway=`** — Default gateway as an IP address.

- **`--nameserver=`** — Primary nameserver, as an IP address.

- **`--nodns`** — Do not configure any DNS server.

- **`--netmask=`** — Netmask for the installed system.

- **`--hostname=`** — Hostname for the installed system.

- **`--ethtool=`** — Specifies additional low-level settings for the network device which will be passed to the ethtool program.

- **`--essid=`** — The network ID for wireless networks.

- **`--wepkey=`** — The encryption key for wireless networks.

- **`--onboot=`** — Whether or not to enable the device at boot time.

- **`--class=`** — The DHCP class.

- **`--mtu=`** — The MTU of the device.

- **`--noipv4`** — Disable IPv4 on this device.

- **`--noipv6`** — Disable IPv6 on this device.

`multipath` (optional)

    multipath --name= --device= --rule=

`part` or **`partition`** (required for installs, ignored for upgrades)
Creates a partition on the system.

If more than one Fedora installation exists on the system on different partitions, the installation program prompts the user and asks which installation to upgrade.

> ⚠️ **Warning**
>
> All partitions created are formatted as part of the installation process unless **`--noformat`** and **`--onpart`** are used.

> ⭐ **Important**
>
> If you select text mode for a kickstart installation, make sure that you specify choices for the partitioning, bootloader, and package selection options. These steps are automated in text mode, and **anaconda** cannot prompt you for missing information. If you do not provide choices for these options, **anaconda** will stop the installation process.

For a detailed example of **part** in action, refer to *Section 13.4.1, "Advanced Partitioning Example"*.

- *<mntpoint>* — The *<mntpoint>* is where the partition is mounted and must be of one of the following forms:

  - */<path>*

    For example, **/**, **/usr**, **/home**

  - **swap**

    The partition is used as swap space.

    To determine the size of the swap partition automatically, use the **--recommended** option:

    ```
    swap --recommended
    ```

    The recommended maximum swap size for machines with less than 2GB of RAM is twice the amount of RAM. For machines with 2GB or more, this recommendation changes to 2GB plus the amount of RAM.

  - **raid.<id>**

    The partition is used for software RAID (refer to **raid**).

  - **pv.<id>**

    The partition is used for LVM (refer to **logvol**).

- **--size=** — The minimum partition size in megabytes. Specify an integer value here such as 500. Do not append the number with MB.

- **--grow** — Tells the partition to grow to fill available space (if any), or up to the maximum size setting.

> **Note**
>
> If you use **--grow=** without setting **--maxsize=** on a swap partition, **Anaconda** will limit the maximum size of the swap partition. For systems that have less than 2GB of physical memory, the imposed limit is twice the amount of physical memory. For systems with more than 2GB, the imposed limit is the size of physical memory plus 2GB.

- **--maxsize=** — The maximum partition size in megabytes when the partition is set to grow. Specify an integer value here, and do not append the number with MB.

- **--noformat** — Tells the installation program not to format the partition, for use with the **--onpart** command.

- **--onpart=** or **--usepart=** — Put the partition on the *already existing* device. For example:

  ```
  partition /home --onpart=hda1
  ```

puts `/home` on `/dev/hda1`, which must already exist.

- `--ondisk=` or `--ondrive=` — Forces the partition to be created on a particular disk. For example, `--ondisk=sdb` puts the partition on the second SCSI disk on the system.

- `--asprimary` — Forces automatic allocation of the partition as a primary partition, or the partitioning fails.

- `--type=` (replaced by `fstype`) — This option is no longer available. Use `fstype`.

- `--fstype=` — Sets the file system type for the partition. Valid values are `ext2`, `ext3`, `swap`, and `vfat`.

- `--start=` — Specifies the starting cylinder for the partition. It requires that a drive be specified with `--ondisk=` or `ondrive=`. It also requires that the ending cylinder be specified with `--end=` or the partition size be specified with `--size=`.

- `--end=` — Specifies the ending cylinder for the partition. It requires that the starting cylinder be specified with `--start=`.

- `--bytes-per-inode=` — Specifies the size of inodes on the filesystem to be made on the partition. Not all filesystems support this option, so it is silently ignored for those cases.

- `--recommended` — Determine the size of the partition automatically.

- `--onbiosdisk` — Forces the partition to be created on a particular disk as discovered by the BIOS.

- `--encrypted` — Specifies that this partition should be encrypted.

- `--passphrase=` — Specifies the passphrase to use when encrypting this partition. Without the above `--encrypted` option, this option does nothing. If no passphrase is specified, the default system-wide one is used, or the installer will stop and prompt if there is no default.

> **Note**
>
> If partitioning fails for any reason, diagnostic messages appear on virtual console 3.

poweroff (optional)

Shut down and power off the system after the installation has successfully completed. Normally during a manual installation, anaconda displays a message and waits for the user to press a key before rebooting. During a kickstart installation, if no completion method is specified, the `reboot` option is used as default.

The `poweroff` option is roughly equivalent to the `shutdown -p` command.

> **Note**
>
> The `poweroff` option is highly dependent on the system hardware in use. Specifically, certain hardware components such as the BIOS, APM (advanced power management), and ACPI (advanced configuration and power interface) must

> be able to interact with the system kernel. Contact your manufacturer for more information on you system's APM/ACPI abilities.

For other completion methods, refer to the **halt**, **reboot**, and **shutdown** kickstart options.

raid (optional)

Assembles a software RAID device. This command is of the form:

```
raid <mntpoint> --level=<level> --device=<mddevice> <partitions*>
```

- *<mntpoint>* — Location where the RAID file system is mounted. If it is **/**, the RAID level must be 1 unless a boot partition (**/boot**) is present. If a boot partition is present, the **/boot** partition must be level 1 and the root (**/**) partition can be any of the available types. The *<partitions*>* (which denotes that multiple partitions can be listed) lists the RAID identifiers to add to the RAID array.

- **--level=** — RAID level to use (0, 1, or 5).

- **--device=** — Name of the RAID device to use (such as md0 or md1). RAID devices range from md0 to md15, and each may only be used once.

- **--bytes-per-inode=** — Specifies the size of inodes on the filesystem to be made on the RAID device. Not all filesystems support this option, so it is silently ignored for those cases.

- **--spares=** — Specifies the number of spare drives allocated for the RAID array. Spare drives are used to rebuild the array in case of drive failure.

- **--fstype=** — Sets the file system type for the RAID array. Valid values are ext2, ext3, swap, and vfat.

- **--fsoptions=** — Specifies a free form string of options to be used when mounting the filesystem. This string will be copied into the /etc/fstab file of the installed system and should be enclosed in quotes.

- **--noformat** — Use an existing RAID device and do not format the RAID array.

- **--useexisting** — Use an existing RAID device and reformat it.

- **--encrypted** — Specifies that this RAID device should be encrypted.

- **--passphrase=** — Specifies the passphrase to use when encrypting this RAID device. Without the above **--encrypted** option, this option does nothing. If no passphrase is specified, the default system-wide one is used, or the installer will stop and prompt if there is no default.

The following example shows how to create a RAID level 1 partition for **/**, and a RAID level 5 for **/usr**, assuming there are three SCSI disks on the system. It also creates three swap partitions, one on each drive.

```
part raid.01 --size=60 --ondisk=sda
part raid.02 --size=60 --ondisk=sdb
part raid.03 --size=60 --ondisk=sdc
```

```
part swap --size=128 --ondisk=sda
part swap --size=128 --ondisk=sdb
part swap --size=128 --ondisk=sdc
```

```
part raid.11 --size=1 --grow --ondisk=sda
part raid.12 --size=1 --grow --ondisk=sdb
part raid.13 --size=1 --grow --ondisk=sdc
```

```
raid / --level=1 --device=md0 raid.01 raid.02 raid.03
raid /usr --level=5 --device=md1 raid.11 raid.12 raid.13
```

For a detailed example of **raid** in action, refer to *Section 13.4.1, "Advanced Partitioning Example"*.

reboot (optional)

Reboot after the installation is successfully completed (no arguments). Normally, kickstart displays a message and waits for the user to press a key before rebooting.

The **reboot** option is roughly equivalent to the **shutdown -r** command.

> **Note**
>
> Use of the **reboot** option *may* result in an endless installation loop, depending on the installation media and method.
>
> The **reboot** option is the default completion method if no other methods are explicitly specified in the kickstart file.

For other completion methods, refer to the **halt**, **poweroff**, and **shutdown** kickstart options.

repo (optional)

Configures additional yum repositories that may be used as sources for package installation. Multiple repo lines may be specified.

```
repo --name=<repoid> [--baseurl=<url>| --mirrorlist=<url>]
```

- **--name=** — The repo id. This option is required.

- **--baseurl=** — The URL for the repository. The variables that may be used in yum repo config files are not supported here. You may use one of either this option or --mirrorlist, not both.

- **--mirrorlist=** — The URL pointing at a list of mirrors for the repository. The variables that may be used in yum repo config files are not supported here. You may use one of either this option or --baseurl, not both.

rootpw (required)

Sets the system's root password to the *<password>* argument.

```
rootpw [--iscrypted] <password>
```

- **--iscrypted** — If this is present, the password argument is assumed to already be encrypted.

selinux (optional)

Sets the state of SELinux on the installed system. SELinux defaults to enforcing in anaconda.

```
selinux [--disabled|--enforcing|--permissive]
```

- **--enforcing** — Enables SELinux with the default targeted policy being enforced.

> **Note**
>
> If the **selinux** option is not present in the kickstart file, SELinux is enabled and set to **--enforcing** by default.

- **--permissive** — Outputs warnings based on the SELinux policy, but does not actually enforce the policy.

- **--disabled** — Disables SELinux completely on the system.

For complete information regarding SELinux for Fedora, refer to the *Fedora 11 Security-Enhanced Linux User Guide*.

services (optional)

Modifies the default set of services that will run under the default runlevel. The services listed in the disabled list will be disabled before the services listed in the enabled list are enabled.

- **--disabled** — Disable the services given in the comma separated list.

- **--enabled** — Enable the services given in the comma separated list.

shutdown (optional)

Shut down the system after the installation has successfully completed. During a kickstart installation, if no completion method is specified, the **reboot** option is used as default.

The **shutdown** option is roughly equivalent to the **shutdown** command.

For other completion methods, refer to the **halt**, **poweroff**, and **reboot** kickstart options.

skipx (optional)

If present, X is not configured on the installed system.

text (optional)

Perform the kickstart installation in text mode. Kickstart installations are performed in graphical mode by default.

> **Important**
>
> If you select text mode for a kickstart installation, make sure that you specify choices for the partitioning, bootloader, and package selection options. These steps are automated in text mode, and **anaconda** cannot prompt you for missing

information. If you do not provide choices for these options, **anaconda** will stop the installation process.

timezone (required)

Sets the system time zone to *<timezone>* which may be any of the time zones listed by **timeconfig**.

```
timezone [--utc] <timezone>
```

- **--utc** — If present, the system assumes the hardware clock is set to UTC (Greenwich Mean) time.

upgrade (optional)

Tells the system to upgrade an existing system rather than install a fresh system. You must specify one of **cdrom**, **harddrive**, **nfs**, or **url** (for FTP and HTTP) as the location of the installation tree. Refer to **install** for details.

user (optional)

Creates a new user on the system.

```
user --name=<username> [--groups=<list>] [--homedir=<homedir>] [--
password=<password>] [--iscrypted] [--shell=<shell>] [--uid=<uid>]
```

- **--name=** — Provides the name of the user. This option is required.

- **--groups=** — In addition to the default group, a comma separated list of group names the user should belong to.

- **--homedir=** — The home directory for the user. If not provided, this defaults to / home/*<username>*.

- **--password=** — The new user's password. If not provided, the account will be locked by default.

- **--iscrypted=** — Is the password provided by --password already encrypted or not?

- **--shell=** — The user's login shell. If not provided, this defaults to the system default.

- **--uid=** — The user's UID. If not provided, this defaults to the next available non-system UID.

vnc (optional)

Allows the graphical installation to be viewed remotely via VNC. This method is usually preferred over text mode, as there are some size and language limitations in text installs. With no options, this command will start a VNC server on the machine with no password and will print out the command that needs to be run to connect a remote machine.

```
vnc [--host=<hostname>] [--port=<port>] [--password=<password>]
```

- **--host=** — Instead of starting a VNC server on the install machine, connect to the VNC viewer process listening on the given hostname.

- **--port=** — Provide a port that the remote VNC viewer process is listening on. If not provided, anaconda will use the VNC default.

- **--password=** — Set a password which must be provided to connect to the VNC session. This is optional, but recommended.

volgroup (optional)

Use to create a Logical Volume Management (LVM) group with the syntax:

```
volgroup <name> <partition> <options>
```

The options are as follows:

- **--noformat** — Use an existing volume group and do not format it.

- **--useexisting** — Use an existing volume group and reformat it.

- **--pesize=** — Set the size of the physical extents.

Create the partition first, create the logical volume group, and then create the logical volume. For example:

```
part pv.01 --size 3000
volgroup myvg pv.01
logvol / --vgname=myvg --size=2000 --name=rootvol
```

For a detailed example of **volgroup** in action, refer to *Section 13.4.1, "Advanced Partitioning Example"*.

xconfig (optional)

Configures the X Window System. If this option is not given, the user must configure X manually during the installation, if X was installed; this option should not be used if X is not installed on the final system.

- **--driver** — Specify the X driver to use for the video hardware.

- **--videoram=** — Specifies the amount of video RAM the video card has.

- **--defaultdesktop=** — Specify either GNOME or KDE to set the default desktop (assumes that GNOME Desktop Environment and/or KDE Desktop Environment has been installed through **%packages**).

- **--startxonboot** — Use a graphical login on the installed system.

- **--resolution=** — Specify the default resolution for the X Window System on the installed system. Valid values are 640x480, 800x600, 1024x768, 1152x864, 1280x1024, 1400x1050, 1600x1200. Be sure to specify a resolution that is compatible with the video card and monitor.

- **--depth=** — Specify the default color depth for the X Window System on the installed system. Valid values are 8, 16, 24, and 32. Be sure to specify a color depth that is compatible with the video card and monitor.

zerombr (optional)

Ifd **zerombr** is specified any invalid partition tables found on disks are initialized. This destroys all of the contents of disks with invalid partition tables.

Note that in previous versions of Red Hat Enterprise Linux, this command was specified as **zerombr yes**. This form is now deprecated; you should now simply specify **zerombr** in your kickstart file instead.

zfcp (optional)

Define a Fiber channel device (IBM System z).

zfcp [--devnum=<*devnum*>] [--fcplun=<*fcplun*>] [--scsiid=<*scsiid*>] [-- scsilun=<*scsilun*>] [--wwpn=<*wwpn*>]

%include (optional)

Use the **%include /path/to/file** command to include the contents of another file in the kickstart file as though the contents were at the location of the **%include** command in the kickstart file.

13.4.1. Advanced Partitioning Example

The following is a single, integrated example showing the **clearpart**, **raid**, **part**, **volgroup**, and **logvol** kickstart options in action:

```
clearpart --drives=hda,hdc --initlabel
# Raid 1 IDE config
part raid.11    --size 1000     --asprimary     --ondrive=hda
part raid.12    --size 1000     --asprimary     --ondrive=hda
part raid.13    --size 2000     --asprimary     --ondrive=hda
part raid.14    --size 8000                     --ondrive=hda
part raid.15    --size 1 --grow                 --ondrive=hda

part raid.21    --size 1000     --asprimary     --ondrive=hdc
part raid.22    --size 1000     --asprimary     --ondrive=hdc
part raid.23    --size 2000     --asprimary     --ondrive=hdc
part raid.24    --size 8000                     --ondrive=hdc
part raid.25    --size 1 --grow                 --ondrive=hdc

# You can add --spares=x
raid /          --fstype ext3 --device md0 --level=RAID1 raid.11 raid.21
raid /safe      --fstype ext3 --device md1 --level=RAID1 raid.12 raid.22
raid swap       --fstype swap --device md2 --level=RAID1 raid.13 raid.23
raid /usr       --fstype ext3 --device md3 --level=RAID1 raid.14 raid.24
raid pv.01      --fstype ext3 --device md4 --level=RAID1 raid.15 raid.25


# LVM configuration so that we can resize /var and /usr/local later
volgroup sysvg pv.01
logvol /var             --vgname=sysvg  --size=8000     --name=var
logvol /var/freespace   --vgname=sysvg  --size=8000     --
name=freespacetouse
```

```
logvol /usr/local        --vgname=sysvg  --size=1 --grow --name=usrlocal
```

This advanced example implements LVM over RAID, as well as the ability to resize various directories for future growth.

13.5. Package Selection

Use the **%packages** command to begin a kickstart file section that lists the packages you would like to install (this is for installations only, as package selection during upgrades is not supported).

Packages can be specified by group or by individual package name, including with globs using the asterisk. The installation program defines several groups that contain related packages. Refer to the **variant/repodata/comps-*.xml** file on the first Fedora CD-ROM for a list of groups. Each group has an id, user visibility value, name, description, and package list. In the package list, the packages marked as mandatory are always installed if the group is selected, the packages marked default are selected by default if the group is selected, and the packages marked optional must be specifically selected even if the group is selected to be installed.

In most cases, it is only necessary to list the desired groups and not individual packages. Note that the **Core** and **Base** groups are always selected by default, so it is not necessary to specify them in the **%packages** section.

Here is an example **%packages** selection:

```
%packages
@ X Window System
@ GNOME Desktop Environment
@ Graphical Internet
@ Sound and Video dhcp
```

As you can see, groups are specified, one to a line, starting with an **@** symbol, a space, and then the full group name as given in the **comps.xml** file. Groups can also be specified using the id for the group, such as gnome-desktop. Specify individual packages with no additional characters (the **dhcp** line in the example above is an individual package).

You can also specify which packages not to install from the default package list:

```
-autofs
```

The following options are available for the **%packages** option:

--nobase

 Do not install the @Base group. Use this option if you are trying to create a very small system.

--resolvedeps

 The --resolvedeps option has been deprecated. Dependencies are resolved automatically every time now.

--ignoredeps

 The --ignoredeps option has been deprecated. Dependencies are resolved automatically every time now.

--ignoremissing

Ignore the missing packages and groups instead of halting the installation to ask if the installation should be aborted or continued. For example:

```
%packages --ignoremissing
```

13.6. Pre-installation Script

You can add commands to run on the system immediately after the **ks.cfg** has been parsed. This section must be at the end of the kickstart file (after the commands) and must start with the **%pre** command. You can access the network in the **%pre** section; however, *name service* has not been configured at this point, so only IP addresses work.

> **Note**
>
> Note that the pre-install script is not run in the change root environment.

--interpreter */usr/bin/python*

Allows you to specify a different scripting language, such as Python. Replace */usr/bin/python* with the scripting language of your choice.

13.6.1. Example

Here is an example **%pre** section:

```
%pre
#!/bin/sh
hds=""
mymedia=""
for file in /proc/ide/h* do
 mymedia=`cat $file/media`
 if [ $mymedia == "disk" ] ; then
  hds="$hds `basename $file`"
 fi
done
set $hds
numhd=`echo $#`
drive1=`echo $hds | cut -d' ' -f1`
drive2=`echo $hds | cut -d' ' -f2`
#Write out partition scheme based on whether there are 1 or 2 hard drives

if [ $numhd == "2" ] ; then
 #2 drives
 echo "#partitioning scheme generated in %pre for 2 drives" > /tmp/part-
include
 echo "clearpart --all" >> /tmp/part-include
 echo "part /boot --fstype ext3 --size 75 --ondisk hda" >> /tmp/part-
include
```

```
 echo "part / --fstype ext3 --size 1 --grow --ondisk hda" >> /tmp/part-
include
 echo "part swap --recommended --ondisk $drive1" >> /tmp/part-include
 echo "part /home --fstype ext3 --size 1 --grow --ondisk hdb" >> /tmp/
part-include
else
 #1 drive
 echo "#partitioning scheme generated in %pre for 1 drive" > /tmp/part-
include
 echo "clearpart --all" >> /tmp/part-include
 echo "part /boot --fstype ext3 --size 75" >> /tmp/part-includ
 echo "part swap --recommended" >> /tmp/part-include
 echo "part / --fstype ext3 --size 2048" >> /tmp/part-include
 echo "part /home --fstype ext3 --size 2048 --grow" >> /tmp/part-include
fi
```

This script determines the number of hard drives in the system and writes a text file with a different partitioning scheme depending on whether it has one or two drives. Instead of having a set of partitioning commands in the kickstart file, include the line:

```
%include /tmp/part-include
```

The partitioning commands selected in the script are used.

> **Note**
>
> The pre-installation script section of kickstart *cannot* manage multiple install trees or source media. This information must be included for each created ks.cfg file, as the pre-installation script occurs during the second stage of the installation process.

13.7. Post-installation Script

You have the option of adding commands to run on the system once the installation is complete. This section must be at the end of the kickstart file and must start with the **%post** command. This section is useful for functions such as installing additional software and configuring an additional nameserver.

> **Note**
>
> If you configured the network with static IP information, including a nameserver, you can access the network and resolve IP addresses in the **%post** section. If you configured the network for DHCP, the **/etc/resolv.conf** file has not been completed when the installation executes the **%post** section. You can access the network, but you can not resolve IP addresses. Thus, if you are using DHCP, you must specify IP addresses in the **%post** section.

> **Note**
>
> The post-install script is run in a chroot environment; therefore, performing tasks such as copying scripts or RPMs from the installation media do not work.

`--nochroot`

Allows you to specify commands that you would like to run outside of the chroot environment.

The following example copies the file **`/etc/resolv.conf`** to the file system that was just installed.

```
%post --nochroot cp /etc/resolv.conf /mnt/sysimage/etc/resolv.conf
```

`--interpreter /usr/bin/python`

Allows you to specify a different scripting language, such as Python. Replace */usr/bin/python* with the scripting language of your choice.

13.7.1. Examples

Register the system to a Red Hat Network Satellite:

```
%post
( # Note that in this example we run the entire %post section as a
 subshell for logging.
wget -O- http://proxy-or-sat.example.com/pub/bootstrap_script | /bin/bash
/usr/sbin/rhnreg_ks --activationkey=<activationkey>
# End the subshell and capture any output to a post-install log file.
) 1>/root/post_install.log 2>&1
```

Run a script named **`runme`** from an NFS share:

```
mkdir /mnt/temp
mount -o nolock 10.10.0.2:/usr/new-machines /mnt/temp open -s -w --
/mnt/temp/runme
umount /mnt/temp
```

> **Note**
>
> NFS file locking is *not* supported while in kickstart mode, therefore **`-o nolock`** is required when mounting an NFS mount.

13.8. Making the Kickstart File Available

A kickstart file must be placed in one of the following locations:

- On a boot diskette

- On a boot CD-ROM

- On a network

Normally a kickstart file is copied to the boot diskette, or made available on the network. The network-based approach is most commonly used, as most kickstart installations tend to be performed on networked computers.

Let us take a more in-depth look at where the kickstart file may be placed.

13.8.1. Creating Kickstart Boot Media

Diskette-based booting is no longer supported in Fedora. Installations must use CD-ROM or flash memory products for booting. However, the kickstart file may still reside on a diskette's top-level directory, and must be named `ks.cfg`.

To perform a CD-ROM-based kickstart installation, the kickstart file must be named `ks.cfg` and must be located in the boot CD-ROM's top-level directory. Since a CD-ROM is read-only, the file must be added to the directory used to create the image that is written to the CD-ROM. Refer to *Section 3.4.2, "Making an Installation Boot CD-ROM"* for instructions on creating boot media; however, before making the `file.iso` image file, copy the `ks.cfg` kickstart file to the `isolinux/` directory.

To perform a pen-based flash memory kickstart installation, the kickstart file must be named `ks.cfg` and must be located in the flash memory's top-level directory. Create the boot image first, and then copy the `ks.cfg` file.

For example, the following transfers a boot image to the pen drive (`/dev/sda`) using the `dd` command:

```
dd if=diskboot.img of=/dev/sda bs=1M
```

> **Note**
>
> Creation of USB flash memory pen drives for booting is possible, but is heavily dependent on system hardware BIOS settings. Refer to your hardware manufacturer to see if your system supports booting to alternate devices.

13.8.2. Making the Kickstart File Available on the Network

Network installations using kickstart are quite common, because system administrators can easily automate the installation on many networked computers quickly and painlessly. In general, the approach most commonly used is for the administrator to have both a BOOTP/DHCP server and an NFS server on the local network. The BOOTP/DHCP server is used to give the client system its networking information, while the actual files used during the installation are served by the NFS server. Often, these two servers run on the same physical machine, but they are not required to.

To perform a network-based kickstart installation, you must have a BOOTP/DHCP server on your network, and it must include configuration information for the machine on which you are attempting to install Fedora. The BOOTP/DHCP server provides the client with its networking information as well as the location of the kickstart file.

If a kickstart file is specified by the BOOTP/DHCP server, the client system attempts an NFS mount of the file's path, and copies the specified file to the client, using it as the kickstart file. The exact settings required vary depending on the BOOTP/DHCP server you use.

Here is an example of a line from the **dhcpd.conf** file for the DHCP server:

```
filename "/usr/new-machine/kickstart/"; next-server blarg.redhat.com;
```

Note that you should replace the value after filename with the name of the kickstart file (or the directory in which the kickstart file resides) and the value after next-server with the NFS server name.

If the file name returned by the BOOTP/DHCP server ends with a slash ("/"), then it is interpreted as a path only. In this case, the client system mounts that path using NFS, and searches for a particular file. The file name the client searches for is:

```
<ip-addr>-kickstart
```

The **<ip-addr>** section of the file name should be replaced with the client's IP address in dotted decimal notation. For example, the file name for a computer with an IP address of 10.10.0.1 would be **10.10.0.1-kickstart**.

Note that if you do not specify a server name, then the client system attempts to use the server that answered the BOOTP/DHCP request as its NFS server. If you do not specify a path or file name, the client system tries to mount **/kickstart** from the BOOTP/DHCP server and tries to find the kickstart file using the same **<ip-addr>-kickstart** file name as described above.

13.9. Making the Installation Tree Available

The kickstart installation must access an *installation tree*. An installation tree is a copy of the binary Fedora CD-ROMs with the same directory structure.

If you are performing a CD-based installation, insert the Fedora CD-ROM #1 into the computer before starting the kickstart installation.

If you are performing a hard drive installation, make sure the ISO images of the binary Fedora CD-ROMs are on a hard drive in the computer.

If you are performing a network-based (NFS, FTP, or HTTP) installation, you must make the installation tree available over the network. Refer to *Section 3.5, "Preparing for a Network Installation"* for details.

13.10. Starting a Kickstart Installation

To begin a kickstart installation, you must boot the system from boot media you have made or the Fedora CD-ROM #1, and enter a special boot command at the boot prompt. The installation program looks for a kickstart file if the **ks** command line argument is passed to the kernel.

CD-ROM #1 and Diskette

> The **linux ks=floppy** command also works if the **ks.cfg** file is located on a vfat or ext2 file system on a diskette and you boot from the Fedora CD-ROM #1.
>
> An alternate boot command is to boot off the Fedora CD-ROM #1 and have the kickstart file on a vfat or ext2 file system on a diskette. To do so, enter the following command at the boot: prompt:

```
linux ks=hd:fd0:/ks.cfg
```

With Driver Disk

If you need to use a driver disk with kickstart, specify the **dd** option as well. For example, to boot off a boot diskette and use a driver disk, enter the following command at the `boot:` prompt:

```
linux ks=floppy dd
```

Boot CD-ROM

If the kickstart file is on a boot CD-ROM as described in *Section 13.8.1, "Creating Kickstart Boot Media"*, insert the CD-ROM into the system, boot the system, and enter the following command at the `boot:` prompt (where **ks.cfg** is the name of the kickstart file):

```
linux ks=cdrom:/ks.cfg
```

Other options to start a kickstart installation are as follows:

askmethod

Do not automatically use the CD-ROM as the install source if we detect a Fedora CD in your CD-ROM drive.

autostep

Make kickstart non-interactive.

debug

Start up pdb immediately.

dd

Use a driver disk.

dhcpclass=*<class>*

Sends a custom DHCP vendor class identifier. ISC's dhcpcd can inspect this value using "option vendor-class-identifier".

dns=*<dns>*

Comma separated list of nameservers to use for a network installation.

driverdisk

Same as 'dd'.

expert

Turns on special features:
- allows partitioning of removable media

- prompts for a driver disk

gateway=*<gw>*

Gateway to use for a network installation.

graphical

Force graphical install. Required to have ftp/http use GUI.

isa

Prompt user for ISA devices configuration.

ip=<*ip*>

   IP to use for a network installation, use 'dhcp' for DHCP.

keymap=<*keymap*>

   Keyboard layout to use. Valid values are those which can be used for the 'keyboard' kickstart
   command.

ks=nfs:<*server*>:/<*path*>

   The installation program looks for the kickstart file on the NFS server *<server>*, as file *<path>*.
   The installation program uses DHCP to configure the Ethernet card. For example, if your NFS
   server is server.example.com and the kickstart file is in the NFS share **/mydir/ks.cfg**, the
   correct boot command would be **ks=nfs:server.example.com:/mydir/ks.cfg**.

ks=http://<*server*>/<*path*>

   The installation program looks for the kickstart file on the HTTP server *<server>*, as file *<path>*.
   The installation program uses DHCP to configure the Ethernet card. For example, if your HTTP
   server is server.example.com and the kickstart file is in the HTTP directory **/mydir/ks.cfg**, the
   correct boot command would be **ks=http://server.example.com/mydir/ks.cfg**.

ks=floppy

   The installation program looks for the file **ks.cfg** on a vfat or ext2 file system on the diskette in **/
   dev/fd0**.

ks=floppy:/<*path*>

   The installation program looks for the kickstart file on the diskette in **/dev/fd0**, as file *<path>*.

ks=hd:<*device*>:/<*file*>

   The installation program mounts the file system on *<device>* (which must be vfat or ext2), and
   look for the kickstart configuration file as *<file>* in that file system (for example, **ks=hd:sda3:/
   mydir/ks.cfg**).

ks=file:/<*file*>

   The installation program tries to read the file *<file>* from the file system; no mounts are done.
   This is normally used if the kickstart file is already on the **initrd** image.

ks=cdrom:/<*path*>

   The installation program looks for the kickstart file on CD-ROM, as file *<path>*.

ks

   If **ks** is used alone, the installation program configures the Ethernet card to use DHCP. The
   kickstart file is read from the "bootServer" from the DHCP response as if it is an NFS server
   sharing the kickstart file. By default, the bootServer is the same as the DHCP server. The name of
   the kickstart file is one of the following:

   • If DHCP is specified and the boot file begins with a **/**, the boot file provided by DHCP is looked
     for on the NFS server.

   • If DHCP is specified and the boot file begins with something other than a **/**, the boot file
     provided by DHCP is looked for in the **/kickstart** directory on the NFS server.

   • If DHCP did not specify a boot file, then the installation program tries to read the file **/
     kickstart/1.2.3.4-kickstart**, where *1.2.3.4* is the numeric IP address of the machine
     being installed.

ksdevice=*<device>*

The installation program uses this network device to connect to the network. For example, consider a system connected to an NFS server through the eth1 device. To perform a kickstart installation on this system using a kickstart file from the NFS server, you would use the command **ks=nfs:*<server>*:/*<path>* ksdevice=eth1** at the boot: prompt.

kssendmac

Adds HTTP headers to ks=http:// request that can be helpful for provisioning systems. Includes MAC address of all nics in CGI environment variables of the form: "X-RHN-Provisioning-MAC-0: eth0 01:23:45:67:89:ab".

lang=*<lang>*

Language to use for the installation. This should be a language which is valid to be used with the 'lang' kickstart command.

loglevel=*<level>*

Set the minimum level required for messages to be logged. Values for <level> are debug, info, warning, error, and critical. The default value is info.

lowres

Force GUI installer to run at 640x480.

mediacheck

Activates loader code to give user option of testing integrity of install source (if an ISO-based method).

method=cdrom

Do a CDROM based installation.

method=ftp://*<path>*

Use <path> for an FTP installation.

method=hd:*<dev>*:*<path>*

Use <path> on <dev> for a hard drive installation.

method=http://*<path>*

Use <path> for an HTTP installation.

method=nfs:*<path>*

Use <path> for an NFS installation.

netmask=*<nm>*

Netmask to use for a network installation.

nofallback

If GUI fails exit.

nofb

Do not load the VGA16 framebuffer required for doing text-mode installation in some languages.

nofirewire

Do not load support for firewire devices.

noipv6

Disable IPv6 networking during installation.

nokill

A debugging option that prevents anaconda from terminating all running programs when a fatal error occurs.

nomount

Don't automatically mount any installed Linux partitions in rescue mode.

nonet

Do not auto-probe network devices.

noparport

Do not attempt to load support for parallel ports.

nopass

Don't pass keyboard/mouse info to stage 2 installer, good for testing keyboard and mouse config screens in stage2 installer during network installs.

nopcmcia

Ignore PCMCIA controller in system.

noprobe

Do not attempt to detect hw, prompts user instead.

noshell

Do not put a shell on tty2 during install.

nostorage

Do not auto-probe storage devices (SCSI, IDE, RAID).

nousb

Do not load USB support (helps if install hangs early sometimes).

nousbstorage

Do not load usbstorage module in loader. May help with device ordering on SCSI systems.

rescue

Run rescue environment.

resolution=*<mode>*

Run installer in mode specified, '1024x768' for example.

serial

Turns on serial console support.

skipddc

Skips DDC probe of monitor, may help if it's hanging system.

syslog=*<host>*[:*<port>*]

Once installation is up and running, send log messages to the syslog process on *<host>*, and optionally, on port *<port>*. Requires the remote syslog process to accept connections (the -r option).

text

Force text mode install.

> ⭐ **Important**
>
> If you select text mode for a kickstart installation, make sure that you specify choices for the partitioning, bootloader, and package selection options. These steps are automated in text mode, and **anaconda** cannot prompt you for missing information. If you do not provide choices for these options, **anaconda** will stop the installation process.

updates

Prompt for floppy containing updates (bug fixes).

updates=ftp://<path>

Image containing updates over FTP.

updates=http://<path>

Image containing updates over HTTP.

upgradeany

Don't require an /etc/redhat-release that matches the expected syntax to upgrade.

vnc

Enable vnc-based installation. You will need to connect to the machine using a vnc client application.

vncconnect=<host>[:<port>]

Once installation is up and running, connect to the vnc client named *<host>*, and optionally use port *<port>*.

Requires 'vnc' option to be specified as well.

vncpassword=<password>

Enable a password for the vnc connection. This will prevent someone from inadvertently connecting to the vnc-based installation.

Requires 'vnc' option to be specified as well.

Kickstart Configurator

Kickstart Configurator allows you to create or modify a kickstart file using a graphical user interface, so that you do not have to remember the correct syntax of the file.

To use **Kickstart Configurator**, you must be running the X Window System. To start **Kickstart Configurator**, select **Applications** (the main menu on the panel) => **System Tools** => **Kickstart**, or type the command `/usr/sbin/system-config-kickstart`.

As you are creating a kickstart file, you can select **File** => **Preview** at any time to review your current selections.

To start with an existing kickstart file, select **File** => **Open** and select the existing file.

14.1. Basic Configuration
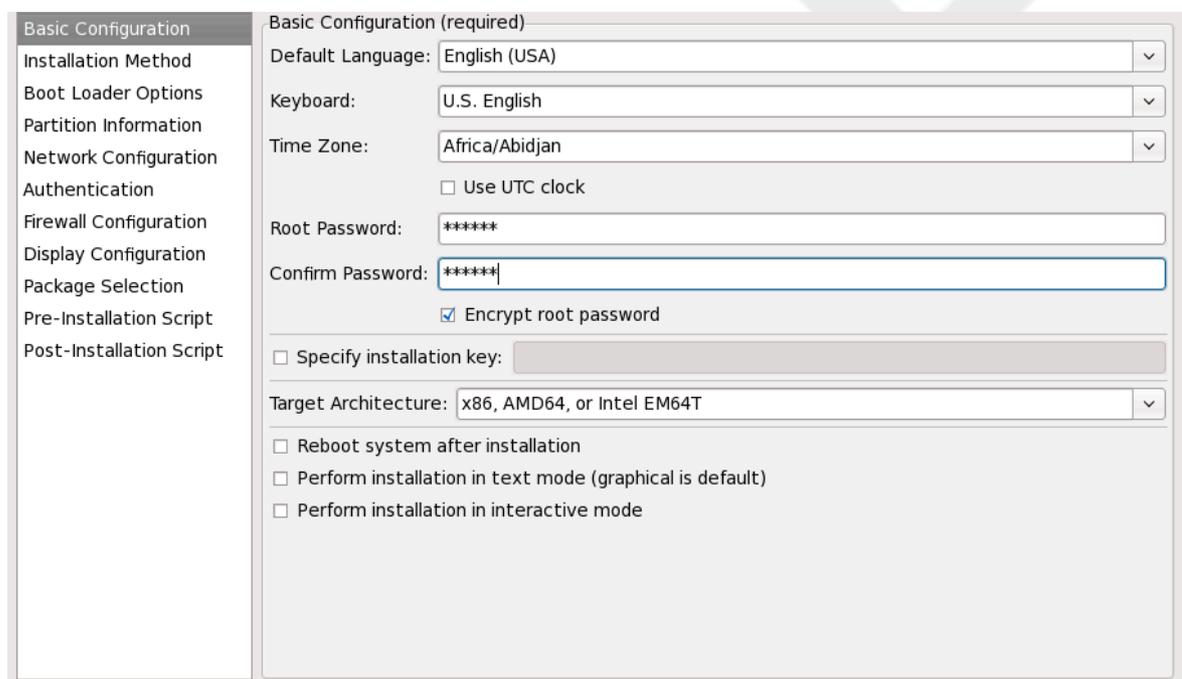
Figure 14.1. Basic Configuration

Choose the language to use during the installation and as the default language to be used after installation from the **Default Language** menu.

Select the system keyboard type from the **Keyboard** menu.

From the **Time Zone** menu, choose the time zone to use for the system. To configure the system to use UTC, select **Use UTC clock**.

Enter the desired root password for the system in the **Root Password** text entry box. Type the same password in the **Confirm Password** text box. The second field is to make sure you do not mistype the password and then realize you do not know what it is after you have completed the installation. To save the password as an encrypted password in the file, select **Encrypt root password**. If the encryption option is selected, when the file is saved, the plain text password that you typed is encrypted and written to the kickstart file. Do not type an already encrypted password and select to

encrypt it. Because a kickstart file is a plain text file that can be easily read, it is recommended that an encrypted password be used.

Choosing **Target Architecture** specifies which specific hardware architecture distribution is used during installation.

Choosing **Reboot system after installation** reboots your system automatically after the installation is finished.

Kickstart installations are performed in graphical mode by default. To override this default and use text mode instead, select the **Perform installation in text mode** option.

You can perform a kickstart installation in interactive mode. This means that the installation program uses all the options pre-configured in the kickstart file, but it allows you to preview the options in each screen before continuing to the next screen. To continue to the next screen, click the **Next** button after you have approved the settings or change them before continuing the installation. To select this type of installation, select the **Perform installation in interactive mode** option.

14.2. Installation Method

Figure 14.2. Installation Method

The **Installation Method** screen allows you to choose whether to perform a new installation or an upgrade. If you choose upgrade, the **Partition Information** and **Package Selection** options are disabled. They are not supported for kickstart upgrades.

Choose the type of kickstart installation or upgrade from the following options:

• **CD-ROM** — Choose this option to install or upgrade from the Fedora CD-ROMs.

• **NFS** — Choose this option to install or upgrade from an NFS shared directory. In the text field for the the NFS server, enter a fully-qualified domain name or IP address. For the NFS directory, enter

the name of the NFS directory that contains the `variant` directory of the installation tree. For example, if the NFS server contains the directory `/mirrors/redhat/i386/Server/`, enter `/mirrors/redhat/i386/` for the NFS directory.

- **FTP** — Choose this option to install or upgrade from an FTP server. In the FTP server text field, enter a fully-qualified domain name or IP address. For the FTP directory, enter the name of the FTP directory that contains the `variant` directory. For example, if the FTP server contains the directory `/mirrors/redhat/i386/Server/`, enter `/mirrors/redhat/i386/Server/` for the FTP directory. If the FTP server requires a username and password, specify them as well.

- **HTTP** — Choose this option to install or upgrade from an HTTP server. In the text field for the HTTP server, enter the fully-qualified domain name or IP address. For the HTTP directory, enter the name of the HTTP directory that contains the `variant` directory. For example, if the HTTP server contains the directory `/mirrors/redhat/i386/Server/`, enter `/mirrors/redhat/i386/Server/` for the HTTP directory.

- **Hard Drive** — Choose this option to install or upgrade from a hard drive. Hard drive installations require the use of ISO (or CD-ROM) images. Be sure to verify that the ISO images are intact before you start the installation. To verify them, use an `md5sum` program as well as the `linux mediacheck` boot option as discussed in *Section 6.3, "Verifying Media"*. Enter the hard drive partition that contains the ISO images (for example, `/dev/hda1`) in the **Hard Drive Partition** text box. Enter the directory that contains the ISO images in the **Hard Drive Directory** text box.

14.3. Boot Loader Options

Figure 14.3. Boot Loader Options

Please note that this screen will be disabled if you have specified a target architecture other than x86 / x86_64.

GRUB is the default boot loader for Fedora on x86 / x86_64 architectures. If you do not want to install a boot loader, select **Do not install a boot loader**. If you choose not to install a boot loader, make

sure you create a boot diskette or have another way to boot your system, such as a third-party boot loader.

You must choose where to install the boot loader (the Master Boot Record or the first sector of the `/boot` partition). Install the boot loader on the MBR if you plan to use it as your boot loader.

To pass any special parameters to the kernel to be used when the system boots, enter them in the **Kernel parameters** text field. For example, if you have an IDE CD-ROM Writer, you can tell the kernel to use the SCSI emulation driver that must be loaded before using `cdrecord` by configuring `hdd=ide-scsi` as a kernel parameter (where `hdd` is the CD-ROM device).

You can password protect the GRUB boot loader by configuring a GRUB password. Select **Use GRUB password**, and enter a password in the **Password** field. Type the same password in the **Confirm Password** text field. To save the password as an encrypted password in the file, select **Encrypt GRUB password**. If the encryption option is selected, when the file is saved, the plain text password that you typed is encrypted and written to the kickstart file. If the password you typed was already encrypted, unselect the encryption option.

If **Upgrade an existing installation** is selected on the **Installation Method** page, select **Upgrade existing boot loader** to upgrade the existing boot loader configuration, while preserving the old entries.

14.4. Partition Information

Figure 14.4. Partition Information

Select whether or not to clear the Master Boot Record (MBR). Choose to remove all existing partitions, remove all existing Linux partitions, or preserve existing partitions.

To initialize the disk label to the default for the architecture of the system (for example, `msdos` for x86 and `gpt` for Itanium), select **Initialize the disk label** if you are installing on a brand new hard drive.

> **Note**
>
> Although **anaconda** and **kickstart** support Logical Volume Management (LVM), at present there is no mechanism for configuring this using the **Kickstart Configurator**.

14.4.1. Creating Partitions

To create a partition, click the **Add** button. The **Partition Options** window shown in *Figure 14.5, "Creating Partitions"* appears. Choose the mount point, file system type, and partition size for the new partition. Optionally, you can also choose from the following:

- In the **Additional Size Options** section, choose to make the partition a fixed size, up to a chosen size, or fill the remaining space on the hard drive. If you selected swap as the file system type, you can select to have the installation program create the swap partition with the recommended size instead of specifying a size.

- Force the partition to be created as a primary partition.

- Create the partition on a specific hard drive. For example, to make the partition on the first IDE hard disk (**/dev/hda**), specify **hda** as the drive. Do not include **/dev** in the drive name.

- Use an existing partition. For example, to make the partition on the first partition on the first IDE hard disk (**/dev/hda1**), specify **hda1** as the partition. Do not include **/dev** in the partition name.

- Format the partition as the chosen file system type.

Figure 14.5. Creating Partitions

To edit an existing partition, select the partition from the list and click the **Edit** button. The same **Partition Options** window appears as when you chose to add a partition as shown in *Figure 14.5, "Creating Partitions"*, except it reflects the values for the selected partition. Modify the partition options and click **OK**.

To delete an existing partition, select the partition from the list and click the **Delete** button.

14.4.1.1. Creating Software RAID Partitions

To create a software RAID partition, use the following steps:

1. Click the **RAID** button.

2. Select **Create a software RAID partition**.

3. Configure the partitions as previously described, except select **Software RAID** as the file system type. Also, you must specify a hard drive on which to make the partition or specify an existing partition to use.

Figure 14.6. Creating a Software RAID Partition

Repeat these steps to create as many partitions as needed for your RAID setup. All of your partitions do not have to be RAID partitions.

After creating all the partitions needed to form a RAID device, follow these steps:

1. Click the **RAID** button.

2. Select **Create a RAID device**.

3. Select a mount point, file system type, RAID device name, RAID level, RAID members, number of spares for the software RAID device, and whether to format the RAID device.
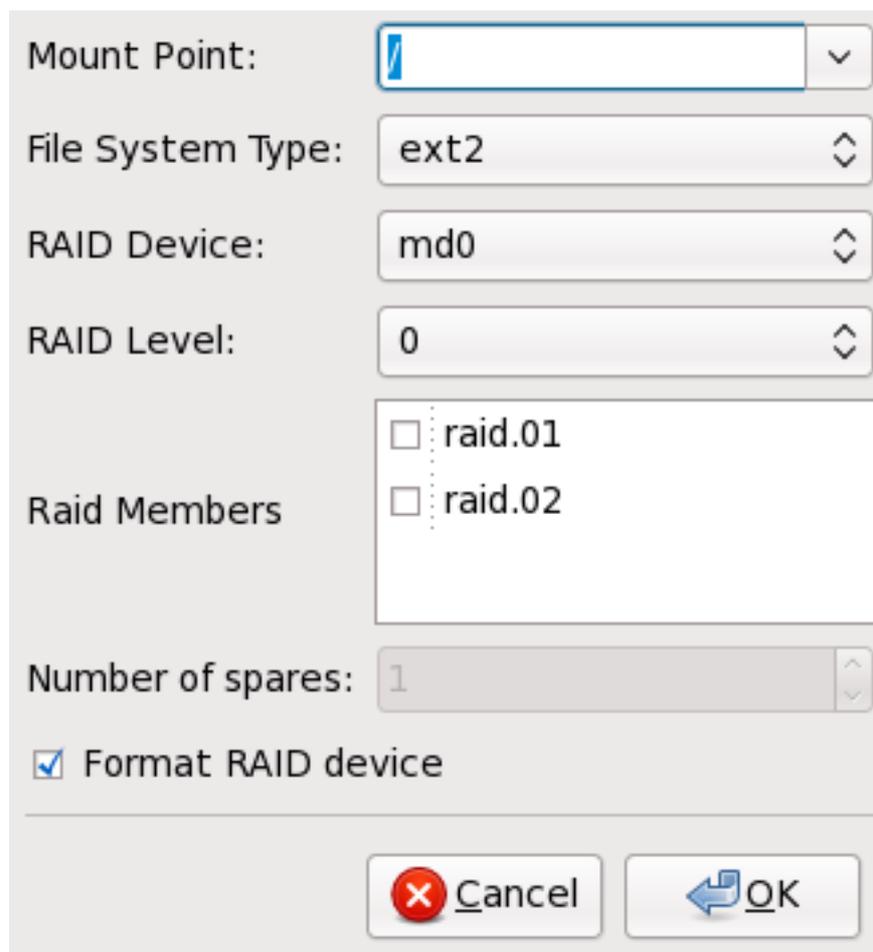
Figure 14.7. Creating a Software RAID Device

4. Click **OK** to add the device to the list.

14.5. Network Configuration

Figure 14.8. Network Configuration

If the system to be installed via kickstart does not have an Ethernet card, do not configure one on the **Network Configuration** page.

Networking is only required if you choose a networking-based installation method (NFS, FTP, or HTTP). Networking can always be configured after installation with the **Network Administration Tool** (`system-config-network`). Refer to the Red Hat Enterprise Linux Deployment Guide for details.

For each Ethernet card on the system, click **Add Network Device** and select the network device and network type for the device. Select **eth0** to configure the first Ethernet card, **eth1** for the second Ethernet card, and so on.
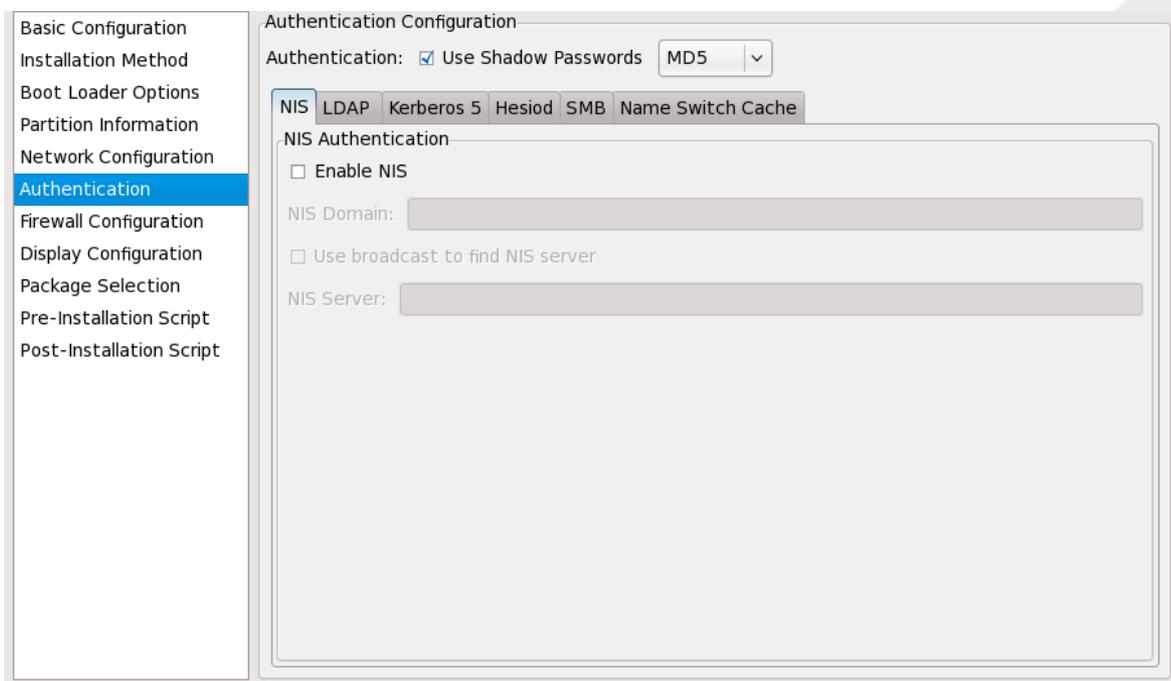
14.6. Authentication

Figure 14.9. Authentication

In the **Authentication** section, select whether to use shadow passwords and MD5 encryption for user passwords. These options are highly recommended and chosen by default.

The **Authentication Configuration** options allow you to configure the following methods of authentication:

- NIS

- LDAP

- Kerberos 5

- Hesiod

- SMB

- Name Switch Cache

These methods are not enabled by default. To enable one or more of these methods, click the appropriate tab, click the checkbox next to **Enable**, and enter the appropriate information for the authentication method. Refer to the Red Hat Enterprise Linux Deployment Guide for more information about the options.

14.7. Firewall Configuration

The **Firewall Configuration** window is similar to the screen in the installation program and the **Security Level Configuration Tool**.
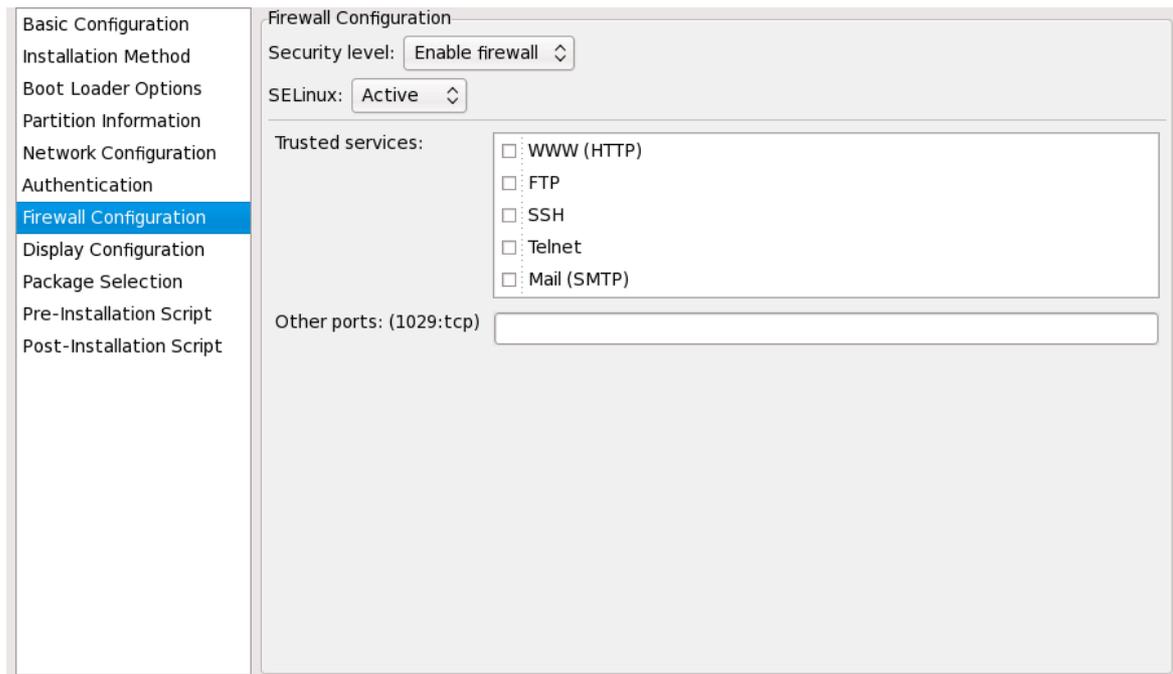
Figure 14.10. Firewall Configuration

If **Disable firewall** is selected, the system allows complete access to any active services and ports. No connections to the system are refused or denied.

Selecting **Enable firewall** configures the system to reject incoming connections that are not in response to outbound requests, such as DNS replies or DHCP requests. If access to services running on this machine is required, you can choose to allow specific services through the firewall.

Only devices configured in the **Network Configuration** section are listed as available **Trusted devices**. Connections from any devices selected in the list are accepted by the system. For example, if **eth1** only receives connections from internal system, you might want to allow connections from it.

If a service is selected in the **Trusted services** list, connections for the service are accepted and processed by the system.

In the **Other ports** text field, list any additional ports that should be opened for remote access. Use the following format: `port:protocol`. For example, to allow IMAP access through the firewall, specify `imap:tcp`. Numeric ports can also be specified explicitly; to allow UDP packets on port 1234 through the firewall, enter `1234:udp`. To specify multiple ports, separate them with commas.

14.7.1. SELinux Configuration

Kickstart can set SELinux to `enforcing`, `permissive` or `disabled` mode. Finer grained configuration is not possible at this time.

14.8. Display Configuration

If you are installing the X Window System, you can configure it during the kickstart installation by checking the **Configure the X Window System** option on the **Display Configuration** window as shown in *Figure 14.11, "X Configuration"*. If this option is not chosen, the X configuration options are disabled and the `skipx` option is written to the kickstart file.

Figure 14.11. X Configuration

Also select whether to start the **Setup Agent** the first time the system is rebooted. It is disabled by default, but the setting can be changed to enabled or enabled in reconfiguration mode. Reconfiguration mode enables the language, mouse, keyboard, root password, security level, time zone, and networking configuration options in addition to the default ones.
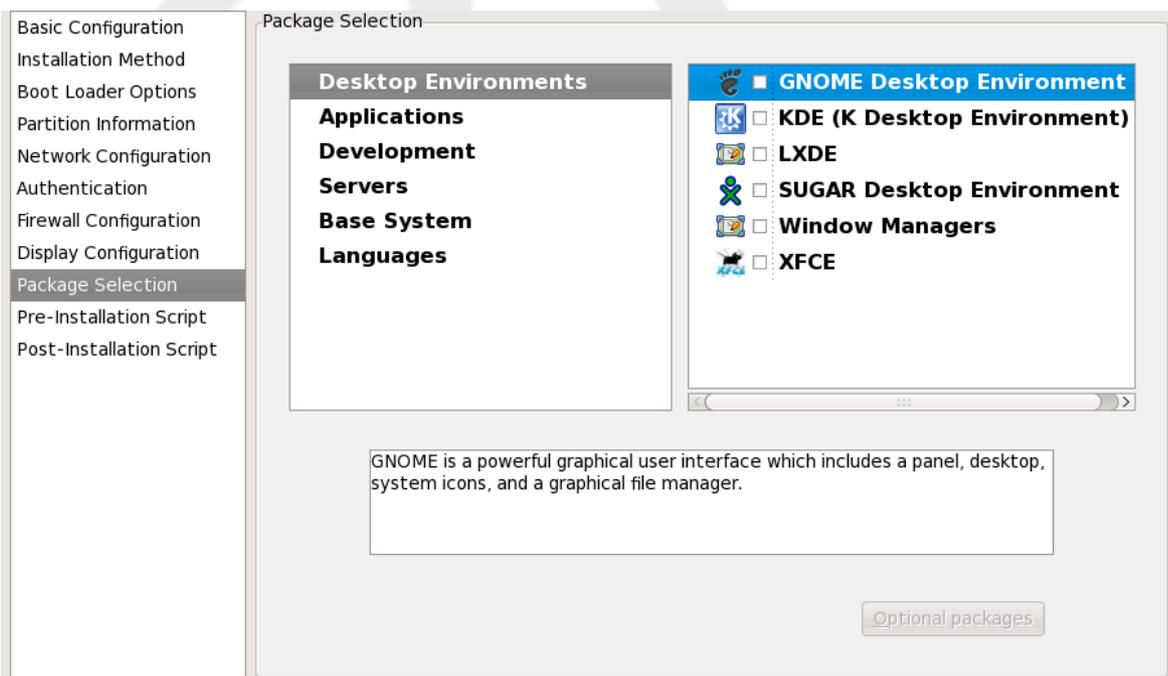
14.9. Package Selection

Figure 14.12. Package Selection

The **Package Selection** window allows you to choose which package groups to install.

Package resolution is carried out automatically.

Currently, **Kickstart Configurator** does not allow you to select individual packages. To install individual packages, modify the `%packages` section of the kickstart file after you save it. Refer to *Section 13.5, "Package Selection"* for details.
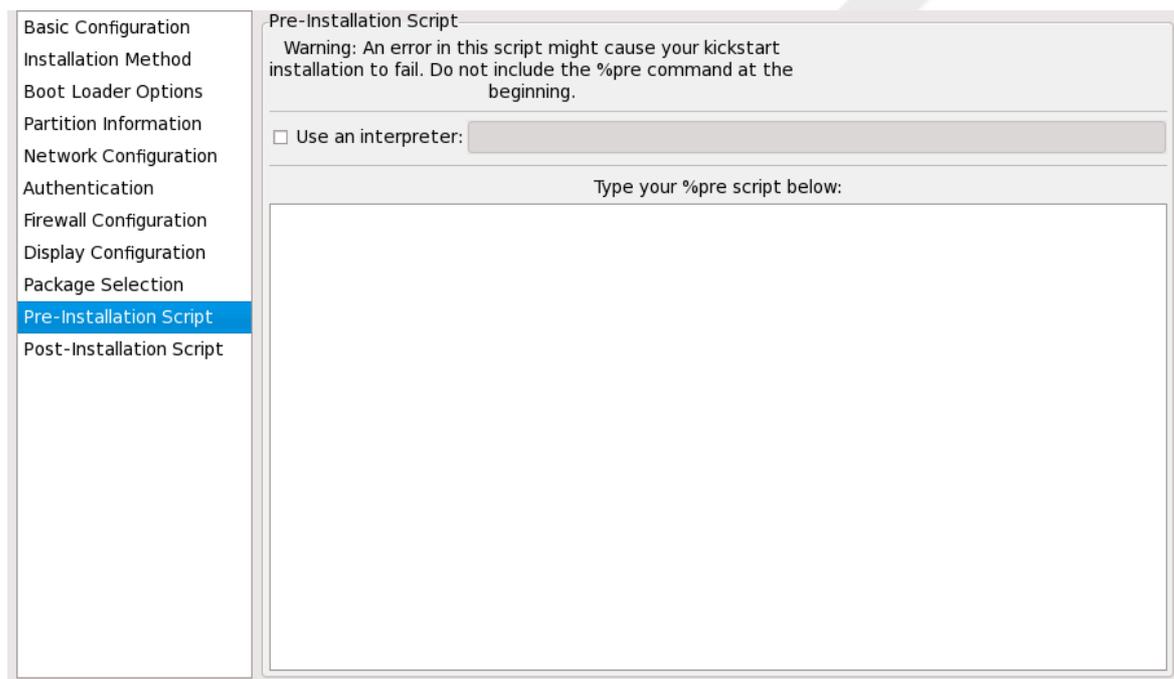
14.10. Pre-Installation Script

Figure 14.13. Pre-Installation Script

You can add commands to run on the system immediately after the kickstart file has been parsed and before the installation begins. If you have configured the network in the kickstart file, the network is enabled before this section is processed. To include a pre-installation script, type it in the text area.

To specify a scripting language to use to execute the script, select the **Use an interpreter** option and enter the interpreter in the text box beside it. For example, `/usr/bin/python2.4` can be specified for a Python script. This option corresponds to using `%pre --interpreter /usr/bin/python2.4` in your kickstart file.

Many of the commands that are available in the pre-installation environment are provided by a version of **busybox** called **busybox-anaconda**. **Busybox**-supplied commands do not provide all features, but supply only the most commonly used features. The following list of available commands include commands provided by **busybox**:

`addgroup`, `adduser`, `adjtimex`, `ar`, `arping`, `ash`, `awk`, `basename`, `bbconfig`, `bunzip2`, `busybox`, `bzcat`, `cal`, `cat`, `catv`, `chattr`, `chgrp`, `chmod`, `chown`, `chroot`, `chvt`, `cksum`, `clear`, `cmp`, `comm`, `cp`, `cpio`, `crond`, `crontab`, `cut`, `date`, `dc`, `dd`, `deallocvt`, `delgroup`, `deluser`, `devfsd`, `df`, `diff`, `dirname`, `dmesg`, `dnsd`, `dos2unix`, `dpkg`, `dpkg-deb`, `du`, `dumpkmap`, `dumpleases`, `e2fsck`, `e2label`, `echo`, `ed`, `egrep`, `eject`, `env`, `ether-wake`, `expr`,

fakeidentd, false, fbset, fdflush, fdformat, fdisk, fgrep, find, findfs, fold, free, freeramdisk, fsck, fsck.ext2, fsck.ext3, fsck.minix, ftpget, ftpput, fuser, getopt, getty, grep, gunzip, gzip, hdparm, head, hexdump, hostid, hostname, httpd, hush, hwclock, id, ifconfig, ifdown, ifup, inetd, insmod, install, ip, ipaddr, ipcalc, ipcrm, ipcs, iplink, iproute, iptunnel, kill, killall, lash, last, length, less, linux32, linux64, ln, load_policy, loadfont, loadkmap, login, logname, losetup, ls, lsattr, lsmod, lzmacat, makedevs, md5sum, mdev, mesg, mkdir, mke2fs, mkfifo, mkfs.ext2, mkfs.ext3, mkfs.minix, mknod, mkswap, mktemp, modprobe, more, mount, mountpoint, msh, mt, mv, nameif, nc, netstat, nice, nohup, nslookup, od, openvt, passwd, patch, pidof, ping, ping6, pipe_progress, pivot_root, printenv, printf, ps, pwd, rdate, readlink, readprofile, realpath, renice, reset, rm, rmdir, rmmod, route, rpm, rpm2cpio, run-parts, runlevel, rx, sed, seq, setarch, setconsole, setkeycodes, setlogcons, setsid, sh, sha1sum, sleep, sort, start-stop-daemon, stat, strings, stty, su, sulogin, sum, swapoff, swapon, switch_root, sync, sysctl, tail, tar, tee, telnet, telnetd, test, tftp, time, top, touch, tr, traceroute, true, tty, tune2fs, udhcpc, udhcpd, umount, uname, uncompress, uniq, unix2dos, unlzma, unzip, uptime, usleep, uudecode, uuencode, vconfig, vi, vlock, watch, watchdog, wc, wget, which, who, whoami, xargs, yes, zcat, zcip

For a description of any of these commands, run:

busybox *command* --help

In addition to the aforementioned commands, the following commands are provided in their full featured versions:

anaconda, bash, bzip2, jmacs, ftp, head, joe, kudzu-probe, list-harddrives, loadkeys, mtools, mbchk, mtools, mini-wm, mtools, jpico, pump, python, python2.4, raidstart, raidstop, rcp, rlogin, rsync, setxkbmap, sftp, shred, ssh, syslinux, syslogd, tac, termidx, vncconfig, vncpasswd, xkbcomp, Xorg, Xvnc, zcat

> ## Warning
> Do not include the **%pre** command. It is added for you.

> ## Note
> The pre-installation script is run after the source media is mounted and stage 2 of the bootloader has been loaded. For this reason it is not possible to change the source media in the pre-installation script.

14.11. Post-Installation Script

Figure 14.14. Post-Installation Script

You can also add commands to execute on the system after the installation is completed. If the network is properly configured in the kickstart file, the network is enabled, and the script can include commands to access resources on the network. To include a post-installation script, type it in the text area.

> ## ⚠ Warning
>
> Do not include the **%post** command. It is added for you.

For example, to change the message of the day for the newly installed system, add the following command to the **%post** section:

```
echo "Hackers will be punished!" > /etc/motd
```

> ## 📝 Note
>
> More examples can be found in *Section 13.7.1, "Examples"*.

14.11.1. Chroot Environment

To run the post-installation script outside of the chroot environment, click the checkbox next to this option on the top of the **Post-Installation** window. This is equivalent to using the **--nochroot** option in the **%post** section.

To make changes to the newly installed file system, within the post-installation section, but outside of the chroot environment, you must prepend the directory name with **/mnt/sysimage/**.

For example, if you select **Run outside of the chroot environment**, the previous example must be changed to the following:

```
echo "Hackers will be punished!" > /mnt/sysimage/etc/motd
```

14.11.2. Use an Interpreter

To specify a scripting language to use to execute the script, select the **Use an interpreter** option and enter the interpreter in the text box beside it. For example, **/usr/bin/python2.2** can be specified for a Python script. This option corresponds to using **%post --interpreter** */usr/bin/python2.2* in your kickstart file.

14.12. Saving the File

To review the contents of the kickstart file after you have finished choosing your kickstart options, select **File** => **Preview** from the pull-down menu.
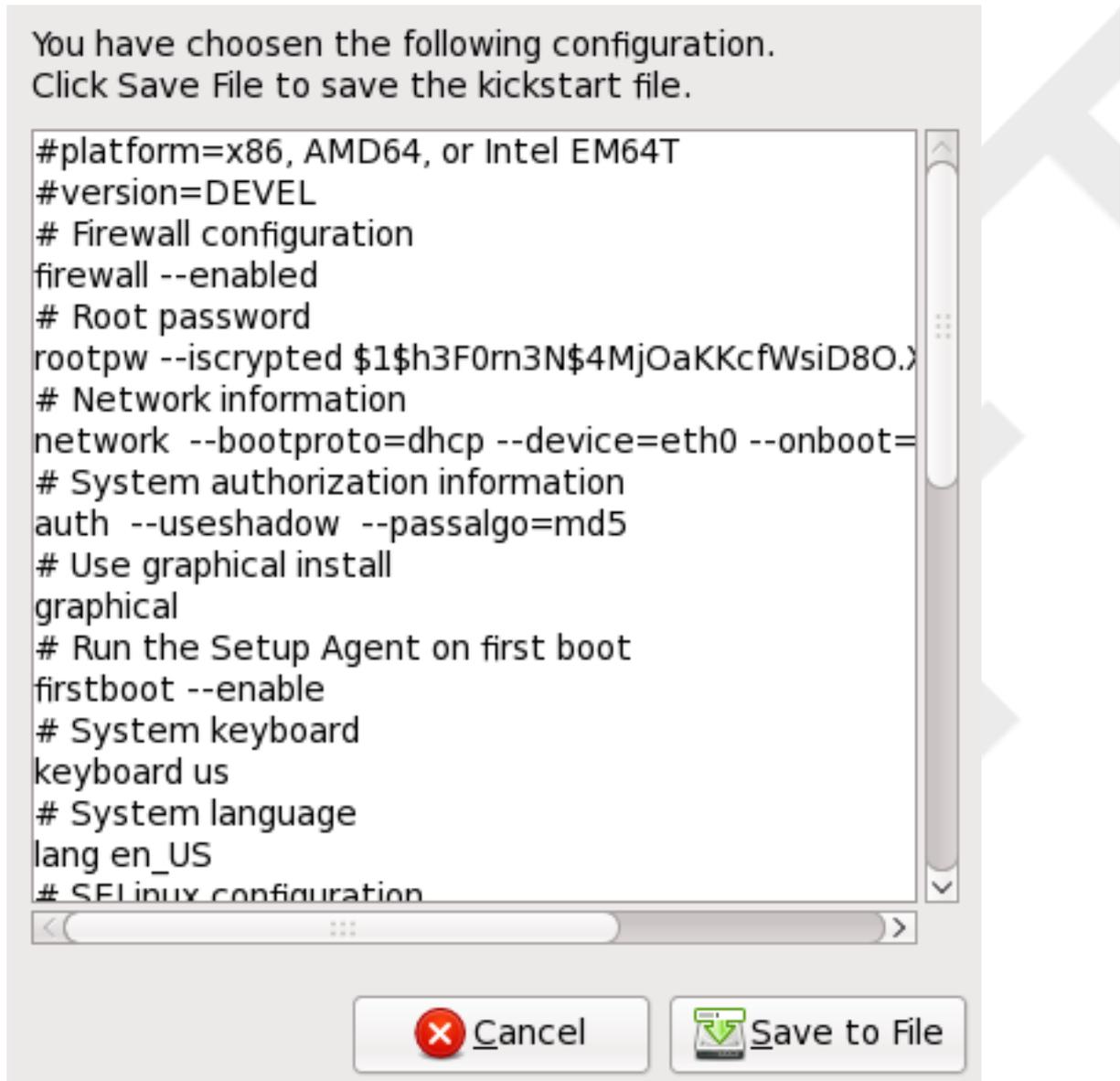
Figure 14.15. Preview

To save the kickstart file, click the **Save to File** button in the preview window. To save the file without previewing it, select **File** => **Save File** or press `Ctrl+S` . A dialog box appears. Select where to save the file.

After saving the file, refer to *Section 13.10, "Starting a Kickstart Installation"* for information on how to start the kickstart installation.

Part IV. After installation

This part of the *Fedora Installation Guide* covers finalizing the installation, as well as some installation-related tasks that you might perform at some time in the future. These include:

- using a Fedora installation disk to rescue a damaged system.

- upgrading to a new version of Fedora.

- removing Fedora from your computer.

Firstboot

Firstboot launches the first time that you start a new Fedora system. Use **Firstboot** to configure the system for use before you log in.

Figure 15.1. Firstboot welcome screen

Select **Forward** to start the **Firstboot**.

> ### Graphical Interface Required
> **Firstboot** requires a graphical interface. If you did not install one, or if Fedora has trouble starting it, you may see a slightly different setup screen.

15.1. License Agreement

This screen displays the overall licensing terms for Fedora. Each software package in Fedora is covered by its own license. All licensing guidelines for Fedora are located at *http://fedoraproject.org/wiki/Legal/Licenses*.
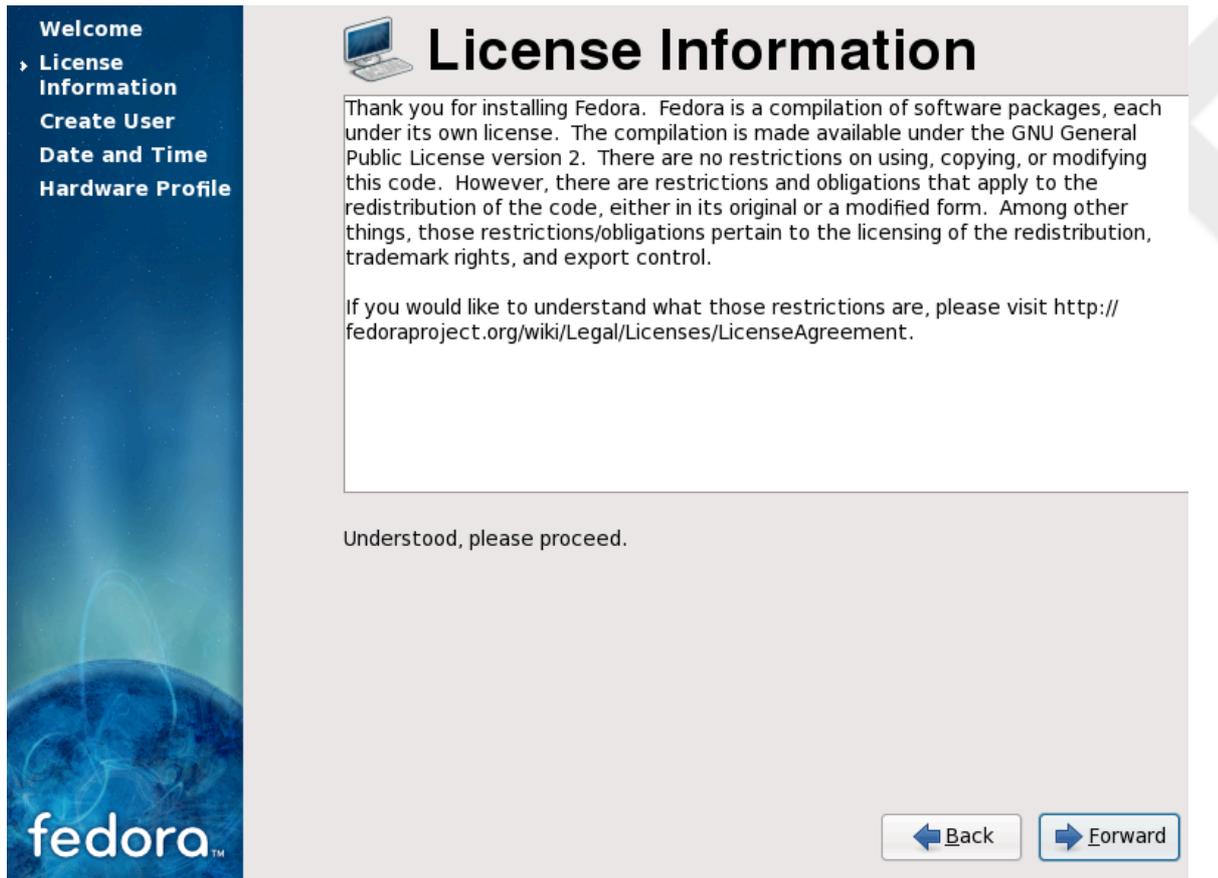
Figure 15.2. Firstboot license screen

If you agree to the terms of the licence, select **Forward**.

15.2. System User

Create a user account for yourself with this screen. Always use this account to log in to your Fedora system, rather than using the `root` account.
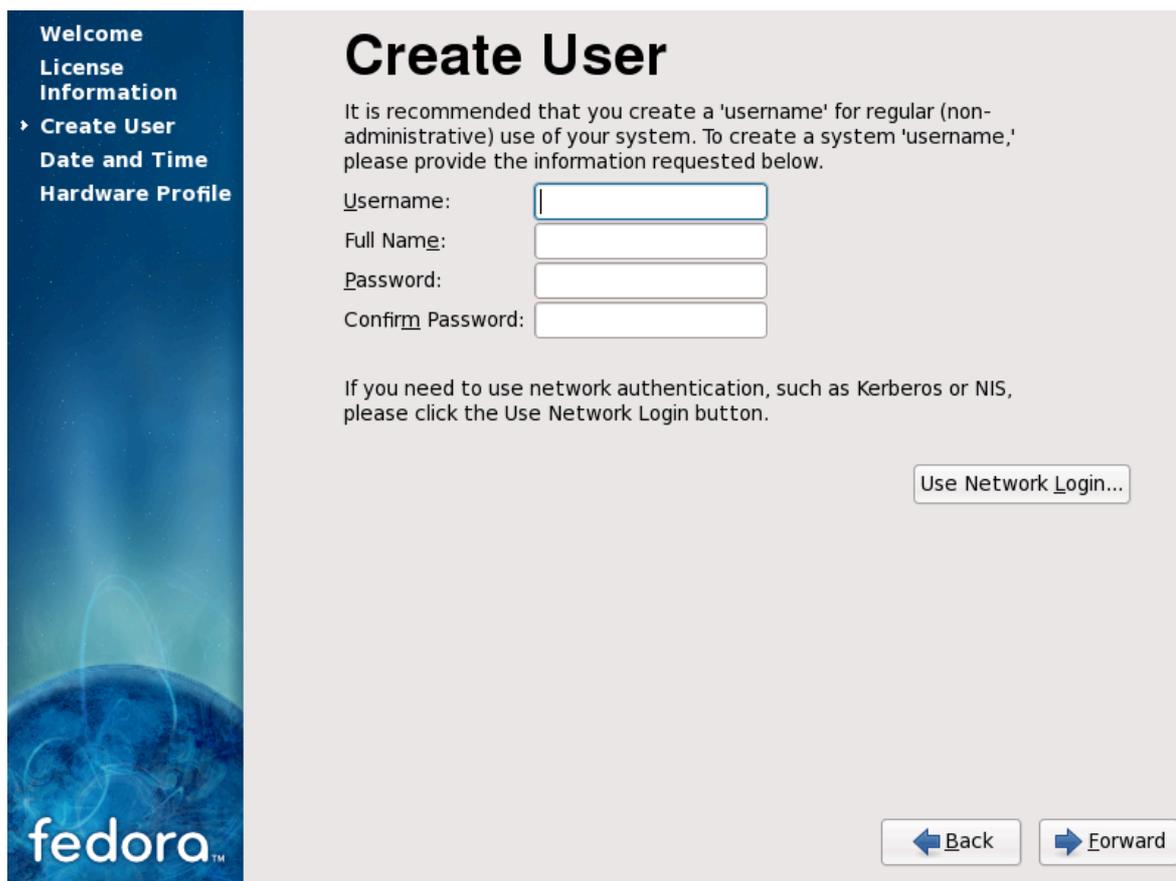
Figure 15.3. Firstboot create user screen

Enter a user name and your full name, and then enter your chosen password. Type your password once more in the **Confirm Password** box to ensure that it is correct. Refer to *Section 7.17, "Set the Root Password"* for guidelines on selecting a secure password.

> **Creating Extra User Accounts**
>
> To add additional user accounts to your system after the installation is complete, choose **System → Administration → Users & Groups**.

To configure Fedora to use network services for authentication or user information, select **Use Network Login...**.

15.3. Date and Time

If your system does not have Internet access or a network time server, manually set the date and time for your system on this screen. Otherwise, use
NTP (Network Time Protocol) servers to maintain the accuracy of the clock. NTP provides time synchronization service to computers on the same network. The Internet contains many computers that offer public NTP services.

The initial display enables you to set the date and time of your system manually.
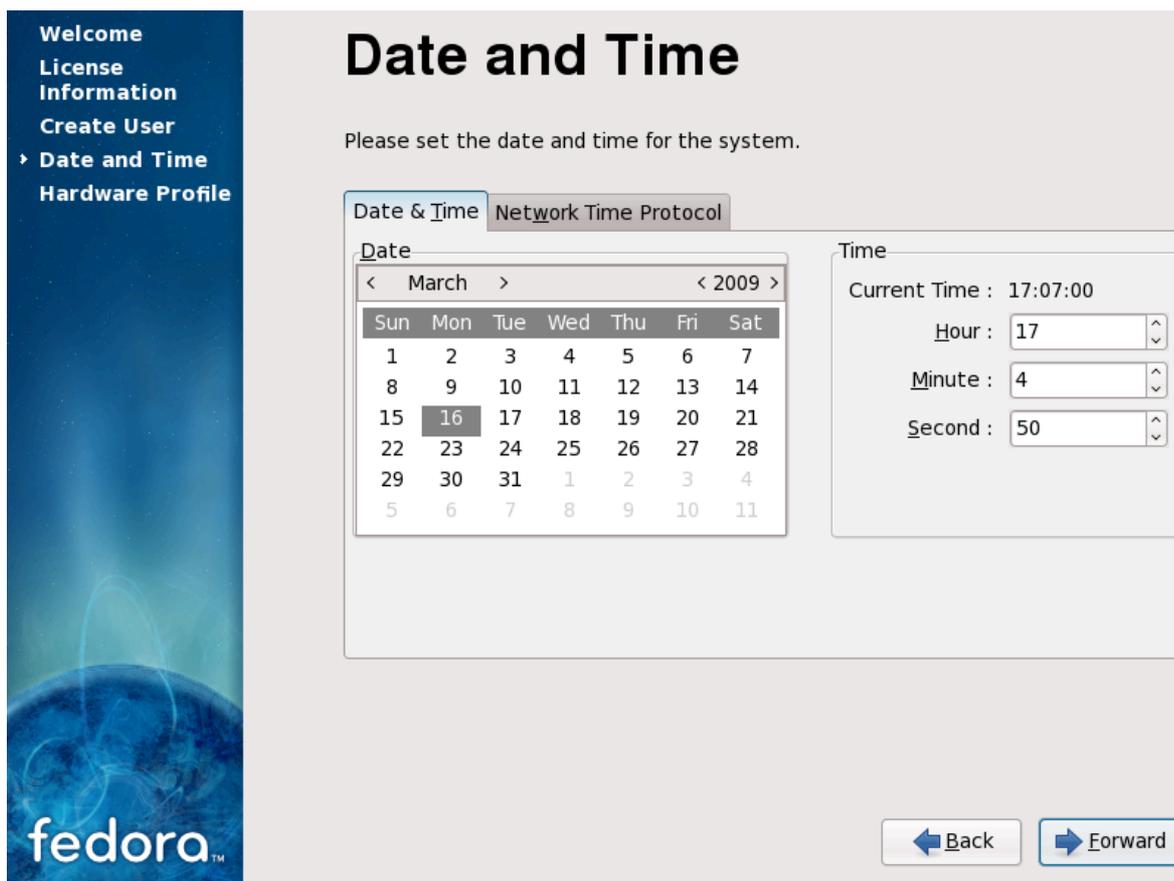
Figure 15.4. Firstboot date and time screen

Select the **Network Time Protocol** tab to configure your system to use NTP servers instead.

> **Setting the Clock**
>
> To change these settings later, choose **System** → **Administration** → **Date & Time**.

To configure your system to use network time servers, select the **Enable Network Time Protocol** option. This option disables the settings on the **Date and Time** tab and enables the other settings on this screen.
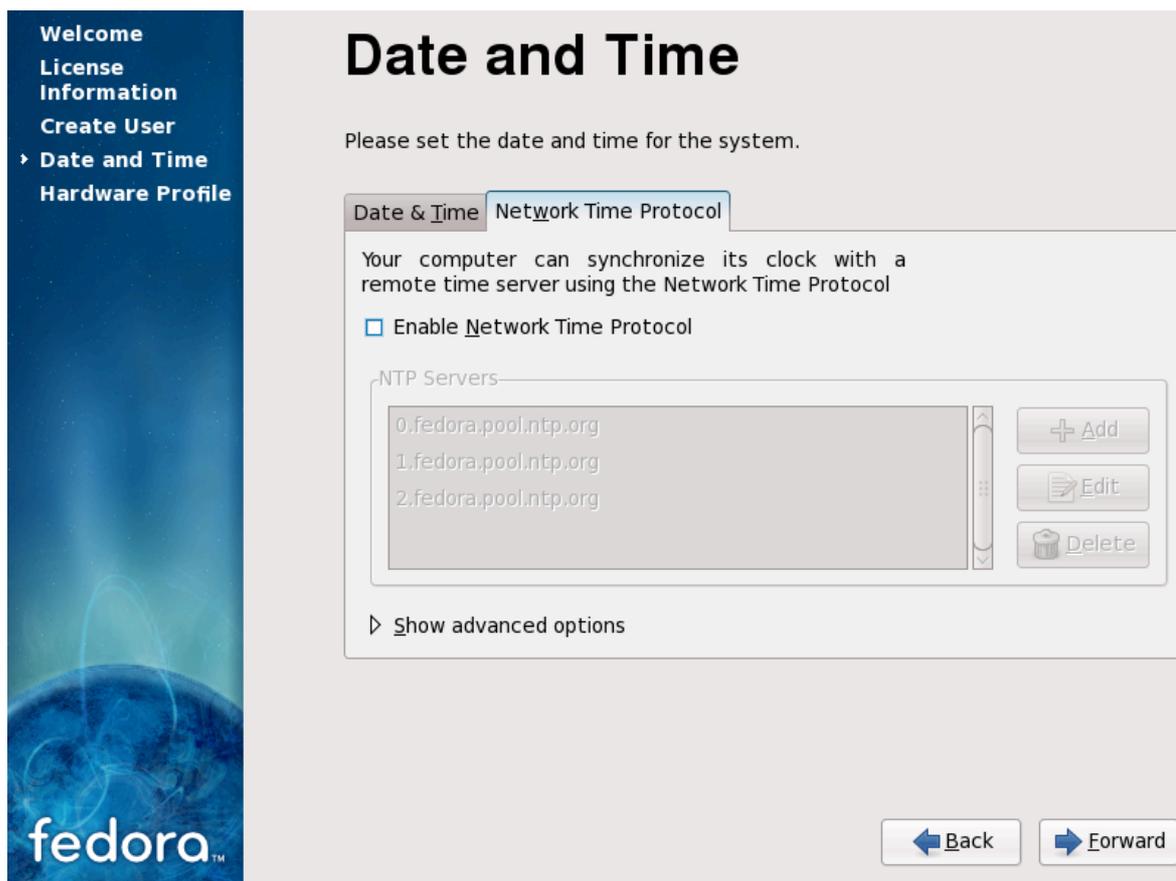
Figure 15.5. Firstboot Network Time Protocol screen

By default, Fedora is configured to use three separate groups, or *pools*, of time servers. Time server pools create redundancy, so if one time server is unavailable, your system synchronizes with another server.

To use an additional time server, select **Add**, and type the DNS name of the server into the box. To remove a server or server pool from the list, select the name and click **Delete**.

If your machine is always connected to the Internet through a wired connection, select the **Synchronize system clock before starting service** option. This option may cause a short delay during startup but ensures accurate time on your system even if the clock is significantly wrong at boot time.

> ### Laptops and NTP
> Do not use this option with laptop computers that sometimes use wireless networks.

If the hardware clock in your computer is highly inaccurate, you may turn off your local time source entirely. To turn off the local time source, select **Show advanced options** and then deselect the **Use Local Time Source** option. If you turn off your local time source, the NTP servers take priority over the internal clock.

If you enable the **Enable NTP Broadcast** advanced option, Fedora attempts to automatically locate time servers on the network.

15.4. Hardware Profile

Firstboot displays a screen that allows you to submit your hardware information anonymously to the Fedora Project. Developers use these hardware details to guide further support efforts. You can read more about this project and its development at *http://smolts.org/*.
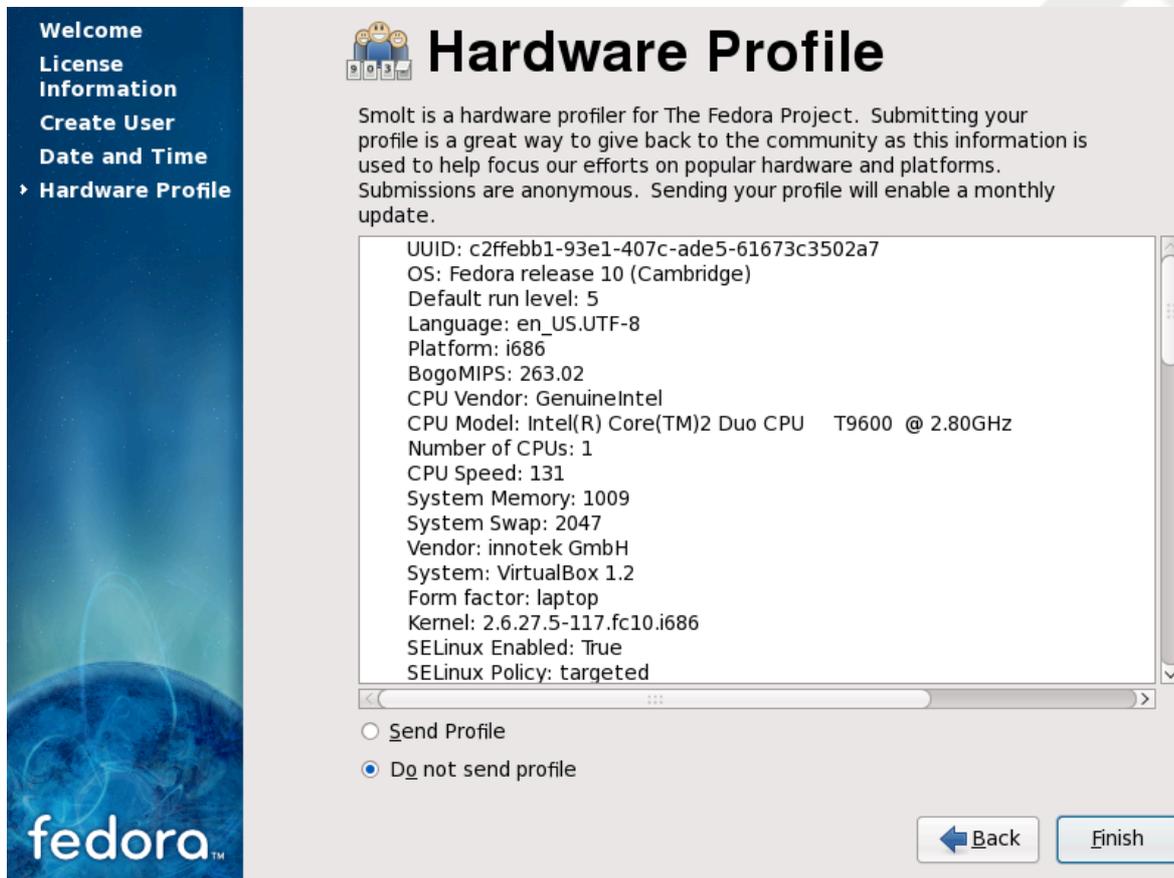

Figure 15.6. Firstboot hardware profile screen

To opt in to this important work, select **Send Profile**. If you choose not to submit any profile data, do not change the default. Select **Finish** to continue to the login screen.

Update Your System

To ensure the security of your system, run a package update after the installation completes. *Chapter 16, Your Next Steps* explains how to update your Fedora system.

Your Next Steps

Fedora provides you with a complete operating system with a vast range of capabilities, supported by a large community.

16.1. Updating Your System

The Fedora Project releases updated software packages for Fedora throughout the support period of each version. Updated packages add new features, improve reliability, resolve bugs, or remove security vulnerabilities. To ensure the security of your system, update regularly, and as soon as possible after a security announcement is issued. Refer to *Section 16.4, "Subscribing to Fedora Announcements and News"* for information on the Fedora announcements services.

An update applet reminds you of updates when they are available. This applet is installed by default in Fedora. It checks for software updates from all configured repositories, and runs as a background service. It generates a notification message on the desktop if updates are found, and you can click the message to update your system's software.

To update your system with the latest packages manually, use **Update System**:

The following updates are available:

! 23 security updates

🐞 240 bug fix updates

✳ 33 enhancement updates

Last update: A few days ago   Refresh

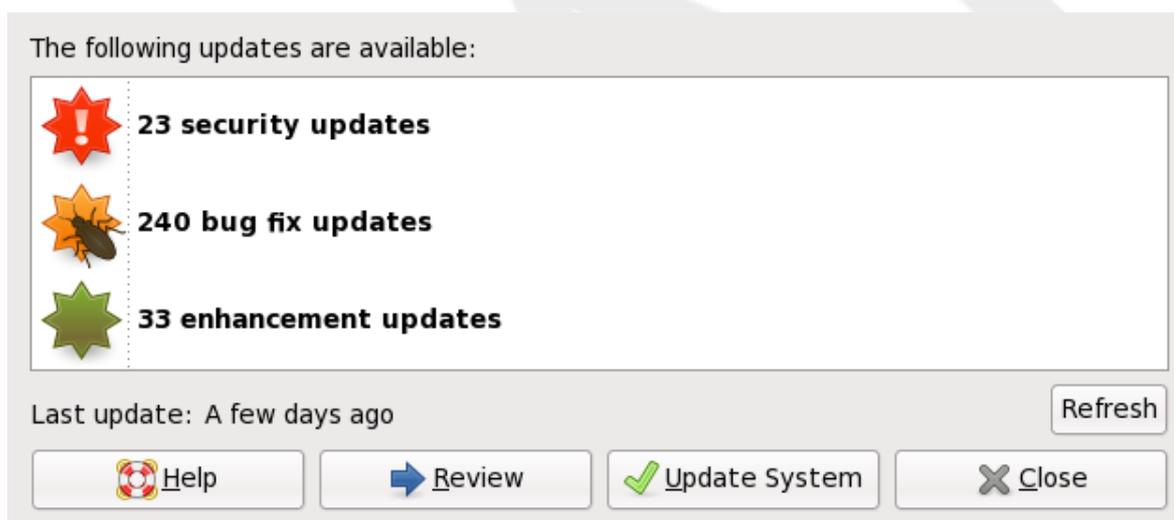Help   Review   Update System   Close

Figure 16.1. Updating your system

1. Choose **System** → **Administration** → **Update System**.

2. To review the list of updated packages, select **Review**.

3. Click **Update System** to begin the update process.

4. If one or more updates require a system reboot, the update process displays a dialog with the option to **Reboot Now**. Either select this option to reboot the system immediately, or **Cancel** it and reboot the system at a more convenient time.

5. If a reboot is not required the update will conclude with a dialog that indicates that the System Update Completed and all selected updates have been successfully installed as well as a button to **Close**
   Update System

To update packages from the command-line, use the
yum utility. Type this command to begin a full update of your system with **yum**:

```
su -c 'yum update'
```

Enter the `root` password when prompted.

Refer to *http://docs.fedoraproject.org/yum/* for more information on **yum**.

> **Network Connection Required**
>
> Ensure that your system has an active network connection before you run the **Update Software** tool, or the **yum** utility. The update process downloads information and packages from a network of servers.

If your Fedora system has a permanent network connection, you may choose to enable daily system updates. To enable automatic updates, follow the instructions on the webpage *http://docs.fedoraproject.org/yum/sn-updating-your-system.html*.

16.2. Finishing an Upgrade

> **System Updates Recommended**
>
> Once you have rebooted your system after performing an upgrade, you should also perform a manual system update. Consult *Section 16.1, "Updating Your System"* for more information.

If you chose to upgrade your system from a previous release rather than perform a fresh installation, you may want to examine the differences in the package set. *Section 7.14.2, "Upgrading Using the Installer"* advised you to create a package listing for your original system. You can now use that listing to determine how to bring your new system close to the original system state.

Most software repository configurations are stored in packages that end with the term **release**. Check the old package list for the repositories that were installed:

```
awk '{print $1}' ~/old-pkglist.txt | grep 'release$'
```

If necessary, retrieve and install these software repository packages from their original sources on the Internet. Follow the instructions at the originating site to install the repository configuration packages for use by **yum** and other software management tools on your Fedora system.

Then run the following commands to make a list of other missing software packages:

```
awk '{print $1}' ~/old-pkglist.txt | sort | uniq > ~/old-pkgnames.txt
rpm -qa --qf '%{NAME}\n' | sort | uniq > ~/new-pkgnames.txt
diff -u ~/old-pkgnames.txt ~/new-pkgnames.txt | grep '^-' | sed 's/^-//'
 > /tmp/pkgs-to-install.txt
```

Now use the file **/tmp/pkgs-to-install.txt** with the **yum** command to restore most or all of your old software:

```
su -c 'yum install `cat /tmp/pkgs-to-install.txt`'
```

> **Missing Software**
>
> Due to changes in package complements between Fedora releases, it is possible this method may not restore all the software on your system. You can use the routines above to again compare the software on your system, and remedy any problems you find.

16.3. Switching to a Graphical Login

If you installed using a text login and wish to switch to a graphical login, follow this procedure.

1. Switch users to the `root` account:

    ```
    su -
    ```

    Provide the administrator password when prompted.

2. If you have not already done so, install one of the graphical desktop environments. For instance, to install the GNOME desktop environment, use this command:

    ```
    yum groupinstall "GNOME Desktop Environment"
    ```

    This step may take some time as your Fedora system downloads and installs additional software. You may be asked to provide the installation media depending on your original installation source.

3. Run the following command to edit the **/etc/inittab** file:

    ```
    nano /etc/inittab
    ```

4. Find the line that includes the text **initdefault**. Change the numeral **3** to **5**.

5. Hit **Ctrl+O** to write the file to disk, and then hit **Ctrl+X** to exit the program.

6. Type **exit** to logout of the administrator account.

If desired, you can reboot the system using the **reboot** command. Your system will restart and present a graphical login.

If you encounter any problems with the graphical login, consult one of the help sources listed in *Section 1.2, "Getting Additional Help"*.

16.4. Subscribing to Fedora Announcements and News

To receive information about package updates, subscribe to either the announcements mailing list, or the RSS feeds.

Fedora Project announcements mailing list
https://www.redhat.com/mailman/listinfo/fedora-announce-list

Fedora Project RSS feeds
http://fedoraproject.org/infofeed/

The announcements mailing list also provides you with news on the Fedora Project, and the Fedora community.

> ⭐ **Security Announcements**
>
> Announcements with the keyword *[SECURITY]* in the title identify package updates that resolve security vulnerabilities.

16.5. Finding Documentation and Support

Members of the Fedora community provide support through mailing lists, Web forums and Linux User Groups (LUGs) across the world.

The Web site for the formally endorsed forums is *http://forums.fedoraforum.org/*.

The following resources provide information on many aspects of Fedora:

- The FAQ on the Fedora Project website
    http://fedoraproject.org/wiki/FAQ/

- The documents available from the Fedora Documentation Project Web site
    http://docs.fedoraproject.org/

- The Linux Documentation Project (LDP)
    http://www.tldp.org/

- The Red Hat Enterprise Linux documentation, much of which also applies to Fedora
    http://www.redhat.com/docs/manuals/enterprise/

Many other organizations and individuals also provide tutorials and HOWTOs for Fedora on their Web sites. You can locate information on any topic by using Google's Linux search site, located at *http://www.google.com/linux*.

16.6. Joining the Fedora Community

The Fedora Project is driven by the individuals that contribute to it. Community members provide support and documentation to other users, help to improve the software included in Fedora by testing, and develop new software alongside the programmers employed by Red Hat. The results of this work are available to all.

To make a difference, start here:

http://join.fedoraproject.org/

Basic System Recovery

When things go wrong, there are ways to fix problems. However, these methods require that you understand the system well. This chapter describes how to boot into rescue mode, single-user mode, and emergency mode, where you can use your own knowledge to repair the system.

17.1. Common Problems

You might need to boot into one of these recovery modes for any of the following reasons:

- You are unable to boot normally into Fedora (runlevel 3 or 5).

- You are having hardware or software problems, and you want to get a few important files off of your system's hard drive.

- You forgot the root password.

17.1.1. Unable to Boot into Fedora

This problem is often caused by the installation of another operating system after you have installed Fedora. Some other operating systems assume that you have no other operating system(s) on your computer. They overwrite the Master Boot Record (MBR) that originally contained the GRUB boot loader. If the boot loader is overwritten in this manner, you cannot boot Fedora unless you can get into rescue mode and reconfigure the boot loader.

Another common problem occurs when using a partitioning tool to resize a partition or create a new partition from free space after installation, and it changes the order of your partitions. If the partition number of your / partition changes, the boot loader might not be able to find it to mount the partition. To fix this problem, boot in rescue mode and modify the **/boot/grub/grub.conf** file.

For instructions on how to reinstall the GRUB boot loader from a rescue environment, refer to *Section 17.2.1, "Reinstalling the Boot Loader"*.

17.1.2. Hardware/Software Problems

This category includes a wide variety of different situations. Two examples include failing hard drives and specifying an invalid root device or kernel in the boot loader configuration file. If either of these occur, you might not be able to reboot into Fedora. However, if you boot into one of the system recovery modes, you might be able to resolve the problem or at least get copies of your most important files.

17.1.3. Root Password

What can you do if you forget your root password? To reset it to a different password, boot into rescue mode or single-user mode, and use the **passwd** command to reset the root password.

17.2. Booting into Rescue Mode

Rescue mode provides the ability to boot a small Fedora environment entirely from CD-ROM, or some other boot method, instead of the system's hard drive.

As the name implies, rescue mode is provided to rescue you from something. During normal operation, your Fedora system uses files located on your system's hard drive to do everything — run programs, store your files, and more.

However, there may be times when you are unable to get Fedora running completely enough to access files on your system's hard drive. Using rescue mode, you can access the files stored on your system's hard drive, even if you cannot actually run Fedora from that hard drive.

To boot into rescue mode, you must be able to boot the system using one of the following methods[1]:

- By booting the system from an installation boot CD-ROM or DVD.

- By booting the system from other installation boot media, such as USB flash devices.

- By booting the system from the Fedora CD-ROM #1 or DVD.

Once you have booted using one of the described methods, add the keyword **rescue** as a kernel parameter. For example, for an x86 system, type the following command at the installation boot prompt:

```
linux rescue
```

You are prompted to answer a few basic questions, including which language to use. It also prompts you to select where a valid rescue image is located. Select from **Local CD-ROM**, **Hard Drive**, **NFS image**, **FTP**, or **HTTP**. The location selected must contain a valid installation tree, and the installation tree must be for the same version of Fedora as the Fedora disk from which you booted. If you used a boot CD-ROM or other media to start rescue mode, the installation tree must be from the same tree from which the media was created. For more information about how to setup an installation tree on a hard drive, NFS server, FTP server, or HTTP server, refer to the earlier section of this guide.

If you select a rescue image that does not require a network connection, you are asked whether or not you want to establish a network connection. A network connection is useful if you need to backup files to a different computer or install some RPM packages from a shared network location, for example.

The following message is displayed:

```
The rescue environment will now attempt to find your Linux installation
and mount it under the directory /mnt/sysimage. You can then make any
changes required to your system. If you want to proceed with this step
choose 'Continue'. You can also choose to mount your file systems read-
only instead of read-write by choosing 'Read-only'. If for some reason this
process fails you can choose 'Skip' and this step will be skipped and you
will go directly to a command shell.
```

If you select **Continue**, it attempts to mount your file system under the directory **/mnt/sysimage/**. If it fails to mount a partition, it notifies you. If you select **Read-Only**, it attempts to mount your file system under the directory **/mnt/sysimage/**, but in read-only mode. If you select **Skip**, your file system is not mounted. Choose **Skip** if you think your file system is corrupted.

Once you have your system in rescue mode, a prompt appears on VC (virtual console) 1 and VC 2 (use the **Ctrl-Alt-F1** key combination to access VC 1 and **Ctrl-Alt-F2** to access VC 2):

---

[1] Refer to the earlier sections of this guide for more details.

```
sh-3.00b#
```

If you selected **Continue** to mount your partitions automatically and they were mounted successfully, you are in single-user mode.

Even if your file system is mounted, the default root partition while in rescue mode is a temporary root partition, not the root partition of the file system used during normal user mode (runlevel 3 or 5). If you selected to mount your file system and it mounted successfully, you can change the root partition of the rescue mode environment to the root partition of your file system by executing the following command:

```
chroot /mnt/sysimage
```

This is useful if you need to run commands such as **rpm** that require your root partition to be mounted as **/**. To exit the **chroot** environment, type **exit** to return to the prompt.

If you selected **Skip**, you can still try to mount a partition or LVM2 logical volume manually inside rescue mode by creating a directory such as *`/foo`*, and typing the following command:

```
mount -t ext3 /dev/mapper/VolGroup00-LogVol02 /foo
```

In the above command, *`/foo`* is a directory that you have created and *`/dev/mapper/VolGroup00-LogVol02`* is the LVM2 logical volume you want to mount. If the partition is of type **ext2**, replace **ext3** with **ext2**.

If you do not know the names of all physical partitions, use the following command to list them:

```
fdisk -l
```

If you do not know the names of all LVM2 physical volumes, volume groups, or logical volumes, use the following commands to list them:

```
pvdisplay
```

```
vgdisplay
```

```
lvdisplay
```

From the prompt, you can run many useful commands, such as:

- **ssh**, **scp**, and **ping** if the network is started

- **dump** and **restore** for users with tape drives

- **parted** and **fdisk** for managing partitions

- **rpm** for installing or upgrading software

- **joe** for editing configuration files

> **Note**
>
> If you try to start other popular editors such as **emacs**, **pico**, or **vi**, the **joe** editor is started.

17.2.1. Reinstalling the Boot Loader

In many cases, the GRUB boot loader can mistakenly be deleted, corrupted, or replaced by other operating systems.

The following steps detail the process on how GRUB is reinstalled on the master boot record:

- Boot the system from an installation boot medium.

- Type **linux rescue** at the installation boot prompt to enter the rescue environment.

- Type **chroot /mnt/sysimage** to mount the root partition.

- Type **/sbin/grub-install /dev/hda** to reinstall the GRUB boot loader, where **/dev/hda** is the boot partition.

- Review the **/boot/grub/grub.conf** file, as additional entries may be needed for GRUB to control additional operating systems.

- Reboot the system.

17.3. Booting into Single-User Mode

One of the advantages of single-user mode is that you do not need a boot CD-ROM; however, it does not give you the option to mount the file systems as read-only or not mount them at all.

If your system boots, but does not allow you to log in when it has completed booting, try single-user mode.

In single-user mode, your computer boots to runlevel 1. Your local file systems are mounted, but your network is not activated. You have a usable system maintenance shell. Unlike rescue mode, single-user mode automatically tries to mount your file system. *Do not use single-user mode if your file system cannot be mounted successfully.* You cannot use single-user mode if the runlevel 1 configuration on your system is corrupted.

On an x86 system using GRUB, use the following steps to boot into single-user mode:

1. At the GRUB splash screen at boot time, press any key to enter the GRUB interactive menu.

2. Select **Fedora** with the version of the kernel that you wish to boot and type **a** to append the line.

3. Go to the end of the line and type **single** as a separate word (press the **Spacebar** and then type **single**). Press **Enter** to exit edit mode.

17.4. Booting into Emergency Mode

In emergency mode, you are booted into the most minimal environment possible. The root file system is mounted read-only and almost nothing is set up. The main advantage of emergency mode over single-user mode is that the **init** files are not loaded. If **init** is corrupted or not working, you can still mount file systems to recover data that could be lost during a re-installation.

To boot into emergency mode, use the same method as described for single-user mode in *Section 17.3, "Booting into Single-User Mode"* with one exception, replace the keyword **single** with the keyword **emergency**.

Upgrading Your Current System

This chapter explains the various methods available for upgrading your Fedora system.

18.1. Determining Whether to Upgrade or Re-Install

While upgrading from Fedora 10 is supported, you are more likely to have a consistent experience by backing up your data and then installing this release of Fedora 11 over your previous Fedora installation.

To upgrade from Fedora 10 you should bring your system up to date before performing the upgrade.

> ⭐ **Important**
>
> It is not possible to upgrade directly from Fedora 9 to Fedora 11. To upgrade Fedora 9 to Fedora 11, upgrade to Fedora 10 first.

This recommended reinstallation method helps to ensure the best system stability possible.

If you currently use Fedora 10, you can perform a traditional, installation program-based upgrade.

However, before you chose to upgrade your system, there are a few things you should keep in mind:

- Individual package configuration files may or may not work after performing an upgrade due to changes in various configuration file formats or layouts.

- If you have one of Red Hat's layered products (such as the Cluster Suite) installed, it may need to be manually upgraded after the upgrade has been completed.

- Third party or ISV applications may not work correctly following the upgrade.

Upgrading your system installs updated versions of the packages which are currently installed on your system.

The upgrade process preserves existing configuration files by renaming them with an **.rpmsave** extension (for example, **sendmail.cf.rpmsave**). The upgrade process also creates a log of its actions in **/root/upgrade.log**.

> ⛔ **Warning**
>
> As software evolves, configuration file formats can change. It is very important to carefully compare your original configuration files to the new files before integrating your changes.

> 📝 **Note**
>
> It is always a good idea to back up any data that you have on your systems. For example, if you are upgrading or creating a dual-boot system, you should back up any data you wish to keep on your hard drive(s). Mistakes do happen and can result in the loss of all of your data.

Some upgraded packages may require the installation of other packages for proper operation. If you choose to customize your packages to upgrade, you may be required to resolve dependency problems. Otherwise, the upgrade procedure takes care of these dependencies, but it may need to install additional packages which are not on your system.

Depending on how you have partitioned your system, the upgrade program may prompt you to add an additional swap file. If the upgrade program does not detect a swap file that equals twice your RAM, it asks you if you would like to add a new swap file. If your system does not have a lot of RAM (less than 256 MB), it is recommended that you add this swap file.

18.2. Upgrading Your System

The **Upgrade Examine** screen appears if you have instructed the installation program to perform an upgrade.

> **Note**
>
> If the contents of your `/etc/fedora-release` file have been changed from the default, your Fedora installation may not be found when attempting an upgrade to Fedora 11.
>
> You can relax some of the checks against this file by booting with the following boot command:
>
> ```
> linux upgradeany
> ```
>
> Use the `linux upgradeany` command if your Fedora installation was not given as an option to upgrade.

To perform an upgrade, select **Perform an upgrade of an existing installation**. Click **Next** when you are ready to begin your upgrade.

To re-install your system, select **Perform a new Fedora installation** and refer to *Chapter 7, Installing on Intel and AMD Systems* for further instructions.

Removing Fedora

We respect your freedom to choose an operating system for your computer. This section explains how to uninstall Fedora.

> ⚠️ **These instructions may destroy data!**
>
> If you have data from Fedora that you want to keep, back it up before you proceed. Write your data to CD, DVD, external hard disk, or other storage device.
>
> As a precaution, also back up data from any other operating systems that are installed on the same computer. Mistakes do happen and can result in the loss of all your data.
>
> If you back up data from Fedora to be used later in another operating system, make sure that the storage medium or device is readable by that other operating system. For example, without extra third-party software, Microsoft Windows cannot read an external hard drive that you have formatted with Fedora to use the ext3 file system.

To uninstall Fedora from your x86-based system, you must remove the Fedora boot loader information from your master boot record (MBR) and any partitions that contain the operating system. The method for removing Fedora from your computer varies, depending on whether Fedora is the only operating system installed on the computer, or whether the computer is configured to dual-boot Fedora and another operating system.

These instructions cannot cover every possible computer configuration. If your computer is configured to boot three or more operating systems, or has a highly-customized partition scheme, use the following sections as a general guide to partition removal with the various tools described. In these situations, you will also need to learn to configure your chosen bootloader. See *Appendix E, The GRUB Boot Loader* for a general introduction to the subject, but detailed instructions are beyond the scope of this document.

> ⭐ **Legacy versions of Microsoft operating systems**
>
> **Fdisk**, the disk partitioning tool provided with MS-DOS and Microsoft Windows, is unable to remove the file systems used by Fedora. MS-DOS and versions of Windows prior to Windows XP (except for Windows 2000) have no other means of removing or modifying partitions. Refer to *Section 19.3, "Replacing Fedora with MS-DOS or legacy versions of Microsoft Windows"* for alternative removal methods for use with MS-DOS and these versions of Windows.

19.1. Fedora is the only operating system on the computer

If Fedora is the only operating system on your computer, use the installation media for the replacement operating system to remove Fedora. Examples of installation media include the Windows XP installation CD, Windows Vista installation DVD, Mac OS X installation CDs or DVD, or the installation CD, CDs, or DVD of another Linux distribution.

Note that some manufacturers of factory-built computers pre-installed with Microsoft Windows do not supply the Windows installation CD or DVD with the computer. The manufacturer may instead have supplied their own "system restore disk", or have included software with the computer that allowed

you to create your own "system restore disk" when you first started the computer. In some cases, the system restore software is stored on a separate partition on the system's hard drive. If you cannot identify the installation media for an operating system that was pre-installed on your computer, consult the documentation supplied with the machine, or contact the manufacturer.

When you have located the installation media for your chosen operating system:

1. Back up any data that you want to keep.

2. Shut down the computer.

3. Boot your computer with the installation disk for the replacement operating system.

4. Follow the prompts presented during the installation process. Windows, OS X, and most Linux installation disks allow you to manually partition your hard drive during the installation process, or will offer you the option to remove all partitions and start with a fresh partition scheme. At this point, remove any existing partitions that the installation software detects or allow the installer to remove the partitions automatically. "System restore" media for computers pre-installed with Microsoft Windows might create a default partition layout automatically without input from you.

> ⚠ **Warning**
>
> If your computer has system restore software stored on a partition on a hard drive, take care when removing partitions while installing an operating system from other media. Under these circumstances, you could destroy the partition holding the system restore software.

19.2. Your computer dual-boots Fedora and another operating system

If your computer is configured to dual-boot Fedora and another operating system, removing Fedora without removing the partitions containing the other operating system and its data is more complicated. Specific instructions for a number of operating systems are set out below. To keep neither the Fedora nor the other operating system, follow the steps described for a computer with only Fedora installed: *Section 19.1, "Fedora is the only operating system on the computer"*

19.2.1. Your computer dual-boots Fedora and a Microsoft Windows operating system

19.2.1.1. Windows 2000, Windows Server 2000, Windows XP, and Windows Server 2003

> ⚠ **Warning**
>
> Once you commence this process, your computer may be left in an unbootable state until you complete the entire set of instructions. Carefully read the steps below before beginning the removal process. Consider opening these instructions on another computer or printing them so that you have access to them at all times during the process.

> This procedure relies on the **Windows Recovery Console** that loads from the Windows installation disk, so you will not be able to complete the procedure without access to this disk. If you start this procedure and do not complete it, you could leave your computer in a condition where you cannot boot it. The "system restore disk" supplied with some factory-built computers that are sold with Windows pre-installed on them might not include the **Windows Recovery Console**.
>
> During the process outlined in these instructions, the **Windows Recovery Console** will prompt you for the Administrator password for your Windows system. Do not follow these instructions unless you know the Administrator password for your system or are certain that an Administrator password has never been created, even by the computer manufacturer.

1. Remove the Fedora partitions

   a. Boot your computer into your Microsoft Windows environment.

   b. Click **Start**>**Run...**, type `diskmgmt.msc` and press `Enter`. The **Disk Management** tool opens.

   The tool displays a graphical representation of your disk, with bars representing each partition. The first partition is usually labeled `NTFS` and corresponds to your `C:` drive. At least two Fedora partitions will be visible. Windows will not display a file system type for these partitions, but may allocate drive letters to some of them.

   c. Right-click on one of the Fedora partitions, then click **Delete Partition** and click **Yes** to confirm the deletion. Repeat this process for the other Fedora partitions on your system. As you delete partitions, Windows labels the space on the hard drive previously occupied by those partitions as `unallocated`.

2. Enable Windows to use the space on your hard drive vacated by Fedora (optional)

   > **Note**
   >
   > This step is not required to remove Fedora from your computer. However, if you skip this step, you will leave part of your hard drive's storage capacity unusable by Windows. Depending on your configuration, this might be a a significant portion of the storage capacity of the drive.

   Decide whether to extend an existing Windows partition to use the extra space, or create a new Windows partition in that space. If you create new a Windows partition, Windows will allocate a new drive letter to it and will interact with it as if it is a separate hard drive.

   Extending an existing Windows partition

   > **Note**
   >
   > The **diskpart** tool used in this step is installed as part of the Windows XP and Windows 2003 operating systems. If you are performing this step on a computer

> running Windows 2000 or Windows Server 2000, you can download a version of **diskpart** for your operating system from the Microsoft website.

a. Click **Start>Run...**, type `diskpart` and press `Enter`. A command window appears.

b. Type `list volume` and press `Enter`. **Diskpart** displays a list of the partitions on your system with a volume number, its drive letter, volume label, filesystem type, and size. Identify the Windows partition that you would like to use to occupy the space vacated on your hard drive by Fedora and take note of its volume number (for example, your Windows `C:` drive might be "Volume 0").

c. Type `select volume N` (where *N* is the volume number for the Windows partition that you want to extend) and press `Enter`. Now type `extend` and press `Enter`. **Diskpart** now extends your chosen partition to fill the remaining space on your hard drive. It will notify you when the operation is complete.

Adding a new Windows partition

a. In the the `Disk Management` window, right-click on disk space that Windows labels as `unallocated` and select `New Partition` from the menu. The **New Partition Wizard** starts.

b. Follow the prompts presented by the **New Partition Wizard**. If you accept the default options, the tool will create a new partition that fills all available space on the hard drive, assigns it the next available drive letter, and formats it with the NTFS file system.

3. Restore the Windows bootloader

a. Insert the Windows installation disk and restart your computer. As your computer starts, the following message will appear on the screen for a few seconds:

```
Press any key to boot from CD
```

Press any key while the message is still showing and the Windows installation software will load.

b. When the `Welcome to Setup` screen appears, you can start the **Windows Recovery Console**. The procedure is slightly different on different versions of Windows:

- On Windows 2000 and Windows Server 2000, press the `R` key, then the `C` key.

- On Windows XP and Windows Server 2003, press the `R` key.

c. The **Windows Recovery Console** scans your hard drives for Windows installations, and assigns a number to each one. It displays a list of Windows installations and prompts you to select one. Type the number corresponding to the Windows installation that you want to restore.

d. The **Windows Recovery Console** prompts you for the Administrator password for your Windows installation. Type the Administrator password and press the `Enter` key. If there is no administrator password for this system, press only the `Enter` key.

e. At the prompt, type the command `fixmbr` and press the `Enter`. The **fixmbr** tool now restores the Master Boot Record for the system.

f. When the prompt reappears, type **exit** and press the **Enter** key.

g. Your computer will restart and boot your Windows operating system.

19.2.1.2. Windows Vista and Windows Server 2008

> ⚠ **Warning**
>
> Once you commence this process, your computer may be left in an unbootable state until you complete the entire set of instructions. Carefully read the steps below before beginning the removal process. Consider opening these instructions on another computer or printing them so that you have access to them at all times during the process.
>
> This procedure relies on the **Windows Recovery Environment** that loads from the Windows installation disk and you will not be able to complete the procedure without access to this disk. If you start this procedure and do not complete it, you could leave your computer in a condition where you cannot boot it. The "system restore disk" supplied with some factory-built computers that are sold with Windows pre-installed on them might not include the **Windows Recovery Environment**.

1. Remove the Fedora partitions

    a. Boot your computer into your Microsoft Windows environment.

    b. Click **Start** then type **diskmgmt.msc** into the **Start Search** box and press **Enter**. The **Disk Management** tool opens.

        The tool displays a graphical representation of your disk, with bars representing each partition. The first partition is usually labeled **NTFS** and corresponds to your **C:** drive. At least two Fedora partitions will be visible. Windows will not display a file system type for these partitions, but may allocate drive letters to some of them.

    c. Right-click on one of the Fedora partitions, then click **Delete Partition** and click **Yes** to confirm the deletion. Repeat this process for the other Fedora partitions on your system. As you delete partitions, Windows labels the space on the hard drive previously occupied by those partitions as **unallocated**.

2. Enable Windows to use the space on your hard drive vacated by Fedora (optional)

> 📝 **Note**
>
> This step is not required to remove Fedora from your computer. However, if you skip this step, you will leave part of your hard drive's storage capacity unusable by Windows. Depending on your configuration, this might be a a significant portion of the storage capacity of the drive.

Decide whether to extend an existing Windows partition to use the extra space, or create a new Windows partition in that space. If you create new a Windows partition, Windows will allocate a new drive letter to it and will interact with it as if it is a separate hard drive.

Extending an existing Windows partition

a. In the **Disk Management** window, right-click on the Windows partition that you want to extend and select **Extend Volume** from the menu. The **Extend Volume Wizard** opens.

b. Follow the prompts presented by the **Extend Volume Wizard**. If you accept the defaults that it offers you, the tool will extend the selected volume to fill all available space on the hard drive.

Adding a new Windows partition

a. In the `Disk Management` window, right-click on disk space that Windows labels as `unallocated` and select `New Simple Volume` from the menu. The **New Simple Volume Wizard** starts.

b. Follow the prompts presented by the **New Simple Volume Wizard**. If you accept the default options, the tool will create a new partition that fills all available space on the hard drive, assigns it the next available drive letter, and formats it with the NTFS file system.

3. Restore the Windows bootloader

    a. Insert the Windows installation disk and restart your computer. As your computer starts, the following message will appear on the screen for a few seconds:

    ```
    Press any key to boot from CD or DVD
    ```

    Press any key while the message is still showing and the Windows installation software will load.

    b. In the **Install Windows** dialog, select a language, time and currency format, and keyboard type. Click **Next**

    c. Click **Repair your computer**.

    d. The **Windows Recovery Environment** (WRE) shows you the Windows installations that it can detect on your system. Select the installation that you want to restore, then click **Next**.

    e. Click **Command prompt**. A command window will open.

    f. Type **bootrec /fixmbr** and press **Enter**.

    g. When the prompt reappears, close the command window, then click **Restart**.

    h. Your computer will restart and boot your Windows operating system.

19.2.2. Your computer dual-boots Fedora and Mac OS X

The procedure to remove Fedora from a system that dual-boots Fedora and Mac OS X varies depending on whether you have installed **Boot Camp** on your computer:

You are not using **Boot Camp** on your computer

1. Open the **Disk Utility** in `/Applications/Utilities`.

2. Select the entry on the left for the disk volume containing Fedora.

3. Click the **Partition** tab on the right side of the dialog.

4. Select the Fedora partitions and click the minus button below the partition layout diagram.

5. Resize your OS X partition to include the newly freed space.

You are using **Boot Camp** on your computer

1. Open the **Boot Camp Assistant** in `/Applications/Utilities`.

2. Select **Create or remove a Windows partition** and click **Next**.

3. If your computer has a single internal disk, click **Restore**.

4. If your computer has multiple internal disks, select the Linux disk, and then select **Restore to a single Mac OS partition**. Click **Continue**.

19.2.3. Your computer dual-boots Fedora and a different Linux distruibution

> **Note**
>
> Because of the differences between the many different Linux distributions, these instructions are a general guide only. Specific details will vary according to your chosen distribution and the configuration of your particular computer. This example uses **GParted** as a partition editor and **gedit** as a text editor, but many other tools are available to perform these tasks. To follow these instructions exactly as written, install **GParted** and **gedit**.

1. Remove Fedora partitions

   a. Boot the Linux version that you want to keep on your computer.

   b. Open **GParted**, either from a desktop menu or by typing `gparted` at the command line and pressing `Enter`.

   c. **GParted** displays the partitions that it detects on your computer, both as a graph and as a table.

   Right-click the Fedora partitions, then select **Delete**.

2. Remove Fedora entries from your bootloader

> **Example only**
>
> These instructions assume that your system uses the **GRUB** bootloader. If you use a different bootloader (such as **LILO**) consult the documentation for that software to identify and remove Fedora entries from its list of boot targets and to ensure that your default operating system is correctly specified.

   a. At the command line, type `su -` and press `Enter`. When the system prompts you for the root password, type the password and press `Enter`.

b. Type **gedit /boot/grub/grub.conf** and press **Enter**. This opens the **grub.conf** file in the **gedit** text editor.

c. A typical Fedora entry in the **grub.conf** file consists of four lines:

title Fedora (2.6.27.19-170.2.35.fc10.i686)

root (hd0,1)

kernel /vmlinuz-2.6.27.19-170.2.35.fc10.i686 ro root=UUID=04a07c13-e6bf-6d5a-b207-002689545705 rhgb quiet

initrd /initrd-2.6.27.19-170.2.35.fc10.i686.img

Example 19.1. Example Fedora entry in **grub.conf**

Depending on the configuration of your system, there may be multiple Fedora entries in **grub.conf**, each corresponding to a different version of the Linux kernel. Delete each of the Fedora entries from the file.

d. **Grub.conf** contains a line that specifies the default operating system to boot, in the format **default=N** where *N* is a number equal to or greater than 0. If *N* is set to 0, **GRUB** will boot the first operating system in the list. If *N* is set to 1, it will boot the second operating system, and so forth.

Identify the entry for the operating system that you want **GRUB** to boot by default and note its place in the order within the list.

Make sure that the **default=** line contains the number *one below* the number of your chosen default operating system in the list.

Save the updated **grub.conf** file and close **gedit**

3. Make space available to your operating system

> **Note**
>
> This step is not required to remove Fedora from your computer. However, if you skip this step, you will leave part of your hard drive's storage capacity unusable by your other Linux operating system. Depending on your configuration, this might be a a significant portion of the storage capacity of the drive.

> **Note**
>
> To carry out this step, you require live media for a Linux distribution, for example, the Fedora Live CD or the Knoppix DVD.

The method to make the space freed by removing the Fedora partitions available to your other Linux operating system differs, depending on whether your chosen operating system is installed on disk partitions configured to use Logical Volume Management (LVM) or not.

- If you do not use LVM

  a. Boot your computer from Linux live media, and install **gparted** if it is not already present.

  b. Open **GParted**, either from a desktop menu or by typing `gparted` at the command line and pressing `Enter`.

  c. **GParted** displays the partitions on your system both as a graph and as a table. Click on the partition that you want to extend to use the space freed by removing Fedora, and click the **Resize/Move** button.

  d. A new dialog opens, allowing you to specify a new size for the partition by entering it as numbers, or by dragging the sides of the graphical representation of the partition so that it fills the available space. Click the **Resize/Move** button in this dialog to confirm your choice.

  e. Back in the main **GParted** window, click **Apply**. Take note of the name of the partition that you just resized, for example, `/dev/sda3`.

  f. When **GParted** finishes resizing the partition, type `e2fsck partition` at a command line and press `Enter`, where `partition` is the partition that you just resized. For example, if you just resized `/dev/sda3`, you would type `e2fsck /dev/sda3`.

     Linux now checks the file system of the newly-resized partition.

  g. When the file system check finishes, type `resize2fs partition` at a command line and press `Enter`, where `partition` is the partition that you just resized. For example, if you just resized `/dev/sda3`, you would type `resize2fs /dev/sda3`.

     Linux now resizes your file system to fill the newly-resized partition.

  h. Restart your computer. The extra space is now available to your Linux installation.

- If you use LVM

  a. Boot your computer from Linux live media and install **gparted** and **lvm2** if they are not already present.

  b. Create a new partition in the free space on the disk

     i. Open **GParted**, either from a desktop menu or by typing `gparted` at the command line and pressing `Enter`.

     ii. **GParted** displays the partitions on your system both as a graph and as a table. The space freed by removing Fedora is labeled `unallocated`. Right-click on the unallocated space and select **New**. Accept the defaults and **GParted** will create a new partition that fills the space available on the drive.

     iii. Click **Apply**. **GParted** writes the changes to your hard drive. Take note of the name of the partition that you just created, and the name of the device that holds the partition. For example, you may have created `/dev/sda3` on device `/dev/sda`.

  c. Change the partition type identifier

     i. **Fdisk** is a partitioning tool capable of preparing partitions for LVM. At a command line, type `fdisk device` and press `Enter`, where `device` is the name of the device on which you just created a partition. For example, `fdisk /dev/sda`.

ii.   At the prompt `Command (m for help):`, press **T** and **Enter** to use **fdisk** to change a partition type.

iii. At the prompt `Partition number (1-4):`, type the number of the partition that you just created. For example, if you just created partition **/dev/sda3**, type the number **3** and press **Enter**. This identifies the partition whose type **fdisk** will change.

iv.   At the prompt `Hex code (type L to list codes):`, type the code **8e** and press **Enter**. This is the code for a Linux LVM partition.

v.   At the prompt `Command (m for help):`, press **W** and **Enter**. **Fdisk** writes the new type code to the partition and exits.

d.   Expand the volume group

i.   At the command prompt, type **lvm** and press **Enter** to start the **lvm2** tool.

ii.   At the `lvm>` prompt, type **pvcreate *partition*** and press **Enter**, where *partition* is the partition that you recently created. For example, **pvcreate /dev/sda3**. This creates **/dev/sda3** as a physical volume in LVM.

iii. At the `lvm>` prompt, type **vgextend *VolumeGroup partition*** and press **Enter**, where *VolumeGroup* is the LVM volume group on which Linux is installed and *partition* is the partition that you recently created. For example, if Linux is installed on **/dev/VolumeGroup00**, you would type **vgextend /dev/VolumeGroup00 /dev/sda3** to extend that volume group to include the physical volume at **/dev/sda3**.

iv.   At the `lvm>` prompt, type **lvextend -l +100%FREE *LogVol*** and press **Enter**, where *LogVol* is the logical volume that contains your Linux filesystem. For example, to extend **LogVol00** to fill the newly-available space in its volume group, **VolGroup00**, type **lvextend -l +100%FREE /dev/VolGroup00/LogVol00**.

v.   At the `lvm>` prompt, type **exit** and press **Enter** to exit **lvm2**

e.   Type **e2fsck *LogVol*** at the command line and press **Enter**, where *LogVol* is the logical volume that you just resized. For example, if you just resized **/dev/VolumeGroup00/LogVol00**, you would type **e2fsck /dev/VolumeGroup00/LogVol00**.

Linux now checks the file system of the newly-resized logical volume.

f.   When the file system check finishes, type **resize2fs *LogVol*** at a command line and press **Enter**, where *LogVol* is the partition that you just resized. For example, if you just resized **/dev/VolumeGroup00/LogVol00**, you would type **resize2fs /dev/VolumeGroup00/LogVol00**.

Linux now resizes your file system to fill the newly-resized logical volume.

g.   Restart your computer. The extra space is now available to your Linux installation.

19.3. Replacing Fedora with MS-DOS or legacy versions of Microsoft Windows

In DOS and Windows, use the Windows **fdisk** utility to create a new MBR with the *undocumented* flag **/mbr**. This *ONLY* rewrites the MBR to boot the primary DOS partition. The command should look like the following:

```
fdisk /mbr
```

If you need to remove Linux from a hard drive and have attempted to do this with the default DOS (Windows) **fdisk**, you will experience the *Partitions exist but they do not exist* problem. The best way to remove non-DOS partitions is with a tool that understands partitions other than DOS.

To begin, insert the Fedora CD #1 and boot your system. Once you have booted off the CD, a boot prompt appears. At the boot prompt, type: **linux rescue**. This starts the rescue mode program.

You are prompted for your keyboard and language requirements. Enter these values as you would during the installation of Fedora.

Next, a screen appears telling you that the program attempts to find a Fedora install to rescue. Select **Skip** on this screen.

After selecting **Skip**, you are given a command prompt where you can access the partitions you would like to remove.

First, type the command **list-harddrives**. This command lists all hard drives on your system that are recognizable by the installation program, as well as their sizes in megabytes.

> ⚠ **Warning**
>
> Be careful to remove only the necessary Fedora partitions. Removing other partitions could result in data loss or a corrupted system environment.

To remove partitions, use the partitioning utility **parted**. Start **parted**, where */dev/hda* is the device on which to remove the partition:

```
parted /dev/hda
```

Using the **print** command, view the current partition table to determine the minor number of the partition to remove:

```
print
```

The **print** command also displays the partition's type (such as linux-swap, ext2, ext3, and so on). Knowing the type of the partition helps you in determining whether to remove the partition.

Remove the partition with the command **rm**. For example, to remove the partition with minor number 3:

```
rm 3
```

> **Important**
>
> The changes start taking place as soon as you press [Enter], so review the command before committing to it.

After removing the partition, use the `print` command to confirm that it is removed from the partition table.

Once you have removed the Linux partitions and made all of the changes you need to make, type `quit` to quit `parted`.

After quitting `parted`, type `exit` at the boot prompt to exit rescue mode and reboot your system, instead of continuing with the installation. The system should reboot automatically. If it does not, you can reboot your computer using `Control+Alt+Delete` .

Part V. Technical appendixes

The appendixes in this section do not contain instructions that tell you how to install Fedora. Instead, they provide technical background that you might find helpful to understand the options that Fedora offers you at various points in the installation process.

Appendix A. An Introduction to Disk Partitions

> **Note**
>
> This appendix is not necessarily applicable to non-x86-based architectures. However, the general concepts mentioned here may apply.

This appendix is not necessarily applicable to non-x86-based architectures. However, the general concepts mentioned here may apply.

If you are reasonably comfortable with disk partitions, you could skip ahead to *Section A.1.4, "Making Room For Fedora"*, for more information on the process of freeing up disk space to prepare for a Fedora installation. This section also discusses the partition naming scheme used by Linux systems, sharing disk space with other operating systems, and related topics.

A.1. Hard Disk Basic Concepts

Hard disks perform a very simple function — they store data and reliably retrieve it on command.

When discussing issues such as disk partitioning, it is important to know a bit about the underlying hardware. Unfortunately, it is easy to become bogged down in details. Therefore, this appendix uses a simplified diagram of a disk drive to help explain what is really happening when a disk drive is partitioned. *Figure A.1, "An Unused Disk Drive"*, shows a brand-new, unused disk drive.

Figure A.1. An Unused Disk Drive

Not much to look at, is it? But if we are talking about disk drives on a basic level, it is adequate. Say that we would like to store some data on this drive. As things stand now, it will not work. There is something we need to do first.

A.1.1. It is Not What You Write, it is How You Write It

Experienced computer users probably got this one on the first try. We need to *format* the drive. Formatting (usually known as "making a *file system*") writes information to the drive, creating order out of the empty space in an unformatted drive.
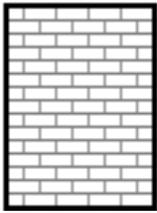
Figure A.2. Disk Drive with a File System

As *Figure A.2, "Disk Drive with a File System"*, implies, the order imposed by a file system involves some trade-offs:

- A small percentage of the drive's available space is used to store file system-related data and can be considered as overhead.

- A file system splits the remaining space into small, consistently-sized segments. For Linux, these segments are known as *blocks.* [1]

Given that file systems make things like directories and files possible, these trade-offs are usually seen as a small price to pay.

It is also worth noting that there is no single, universal file system. As *Figure A.3, "Disk Drive with a Different File System"*, shows, a disk drive may have one of many different file systems written on it. As you might guess, different file systems tend to be incompatible; that is, an operating system that supports one file system (or a handful of related file system types) may not support another. This last statement is not a hard-and-fast rule, however. For example, Fedora supports a wide variety of file systems (including many commonly used by other operating systems), making data interchange between different file systems easy.
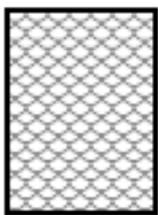
Figure A.3. Disk Drive with a Different File System

Of course, writing a file system to disk is only the beginning. The goal of this process is to actually *store* and *retrieve* data. Let us take a look at our drive after some files have been written to it.
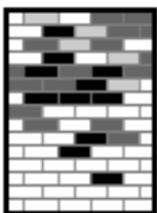
Figure A.4. Disk Drive with Data Written to It

As *Figure A.4, "Disk Drive with Data Written to It"*, shows, some of the previously-empty blocks are now holding data. However, by just looking at this picture, we cannot determine exactly how many files reside on this drive. There may only be one file or many, as all files use at least one block and some files use multiple blocks. Another important point to note is that the used blocks do not have to form

a contiguous region; used and unused blocks may be interspersed. This is known as *fragmentation*. Fragmentation can play a part when attempting to resize an existing partition.

As with most computer-related technologies, disk drives changed over time after their introduction. In particular, they got bigger. Not larger in physical size, but bigger in their capacity to store information. And, this additional capacity drove a fundamental change in the way disk drives were used.

A.1.2. Partitions: Turning One Drive Into Many

As disk drive capacities soared, some people began to wonder if having all of that formatted space in one big chunk was such a great idea. This line of thinking was driven by several issues, some philosophical, some technical. On the philosophical side, above a certain size, it seemed that the additional space provided by a larger drive created more clutter. On the technical side, some file systems were never designed to support anything above a certain capacity. Or the file systems *could* support larger drives with a greater capacity, but the overhead imposed by the file system to track files became excessive.

The solution to this problem was to divide disks into *partitions*. Each partition can be accessed as if it was a separate disk. This is done through the addition of a *partition table*.

> **Note**
>
> While the diagrams in this chapter show the partition table as being separate from the actual disk drive, this is not entirely accurate. In reality, the partition table is stored at the very start of the disk, before any file system or user data. But for clarity, they are separate in our diagrams.
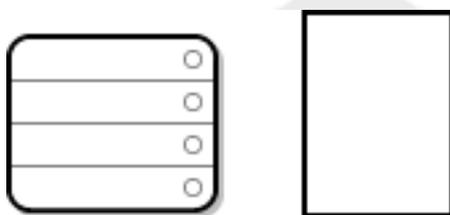
Figure A.5. Disk Drive with Partition Table

As *Figure A.5, "Disk Drive with Partition Table"* shows, the partition table is divided into four sections or four *primary* partitions. A primary partition is a partition on a hard drive that can contain only one logical drive (or section). Each section can hold the information necessary to define a single partition, meaning that the partition table can define no more than four partitions.

Each partition table entry contains several important characteristics of the partition:

- The points on the disk where the partition starts and ends

- Whether the partition is "active"

- The partition's type

Let us take a closer look at each of these characteristics. The starting and ending points actually define the partition's size and location on the disk. The "active" flag is used by some operating systems' boot loaders. In other words, the operating system in the partition that is marked "active" is booted.

The partition's type can be a bit confusing. The type is a number that identifies the partition's anticipated usage. If that statement sounds a bit vague, that is because the meaning of the partition type is a bit vague. Some operating systems use the partition type to denote a specific file system type, to flag the partition as being associated with a particular operating system, to indicate that the partition contains a bootable operating system, or some combination of the three.

By this point, you might be wondering how all this additional complexity is normally used. Refer to *Figure A.6, "Disk Drive With Single Partition"*, for an example.
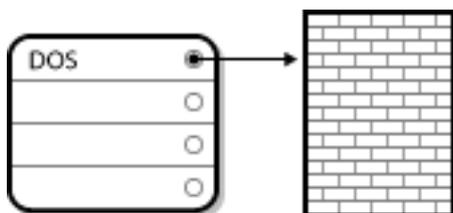
Figure A.6. Disk Drive With Single Partition

In many cases, there is only a single partition spanning the entire disk, essentially duplicating the method used before partitions. The partition table has only one entry used, and it points to the start of the partition.

We have labeled this partition as being of the "DOS" type. Although it is only one of several possible partition types listed in *Table A.1, "Partition Types"*, it is adequate for the purposes of this discussion.

Table A.1, "Partition Types", contains a listing of some popular (and obscure) partition types, along with their hexadecimal numeric values.

Partition Type	Value	Partition Type	Value
Empty	00	Novell Netware 386	65
DOS 12-bit FAT	01	PIC/IX	75
XENIX root	02	Old MINIX	80
XENIX usr	03	Linux/MINUX	81
DOS 16-bit <=32M	04	Linux swap	82
Extended	05	Linux native	83
DOS 16-bit >=32	06	Linux extended	85
OS/2 HPFS	07	Amoeba	93
AIX	08	Amoeba BBT	94
AIX bootable	09	BSD/386	a5
OS/2 Boot Manager	0a	OpenBSD	a6
Win95 FAT32	0b	NEXTSTEP	a7
Win95 FAT32 (LBA)	0c	BSDI fs	b7
Win95 FAT16 (LBA)	0e	BSDI swap	b8
Win95 Extended (LBA)	0f	Syrinx	c7
Venix 80286	40	CP/M	db
Novell	51	DOS access	e1
PPC PReP Boot	41	DOS R/O	e3

Partition Type	Value	Partition Type	Value
GNU HURD	63	DOS secondary	f2
Novell Netware 286	64	BBT	ff

Table A.1. Partition Types

A.1.3. Partitions within Partitions — An Overview of Extended Partitions

Of course, over time it became obvious that four partitions would not be enough. As disk drives continued to grow, it became more and more likely that a person could configure four reasonably-sized partitions and still have disk space left over. There needed to be some way of creating more partitions.

Enter the extended partition. As you may have noticed in *Table A.1, "Partition Types"*, there is an "Extended" partition type. It is this partition type that is at the heart of extended partitions.

When a partition is created and its type is set to "Extended," an extended partition table is created. In essence, the extended partition is like a disk drive in its own right — it has a partition table that points to one or more partitions (now called *logical partitions*, as opposed to the four *primary partitions*) contained entirely within the extended partition itself. *Figure A.7, "Disk Drive With Extended Partition"*, shows a disk drive with one primary partition and one extended partition containing two logical partitions (along with some unpartitioned free space).
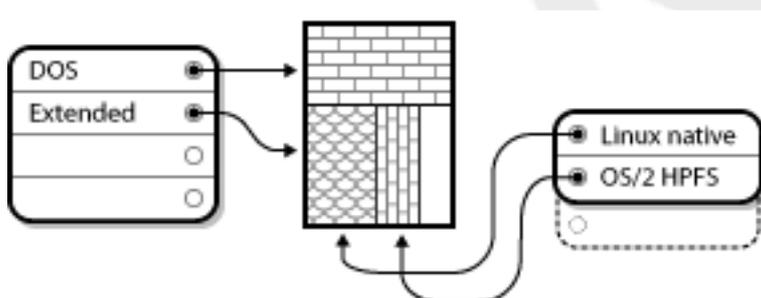
Figure A.7. Disk Drive With Extended Partition

As this figure implies, there is a difference between primary and logical partitions — there can only be four primary partitions, but there is no fixed limit to the number of logical partitions that can exist. However, due to the way in which partitions are accessed in Linux, you should avoid defining more than 12 logical partitions on a single disk drive.

Now that we have discussed partitions in general, let us review how to use this knowledge to install Fedora.

A.1.4. Making Room For Fedora

The following list presents some possible scenarios you may face when attempting to repartition your hard disk:

• Unpartitioned free space is available

• An unused partition is available

• Free space in an actively used partition is available

Let us look at each scenario in order.

> **Note**
>
> Keep in mind that the following illustrations are simplified in the interest of clarity and do not reflect the exact partition layout that you encounter when actually installing Fedora.

A.1.4.1. Using Unpartitioned Free Space

In this situation, the partitions already defined do not span the entire hard disk, leaving unallocated space that is not part of any defined partition. *Figure A.8, "Disk Drive with Unpartitioned Free Space"*, shows what this might look like.
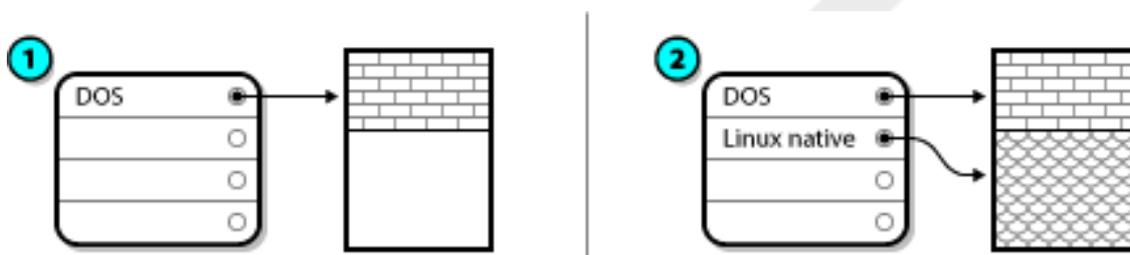
Figure A.8. Disk Drive with Unpartitioned Free Space

In *Figure A.8, "Disk Drive with Unpartitioned Free Space"*, *1* represents an undefined partition with unallocated space and *2* represents a defined partition with allocated space.

If you think about it, an unused hard disk also falls into this category. The only difference is that *all* the space is not part of any defined partition.

In any case, you can create the necessary partitions from the unused space. Unfortunately, this scenario, although very simple, is not very likely (unless you have just purchased a new disk just for Fedora). Most pre-installed operating systems are configured to take up all available space on a disk drive (refer to *Section A.1.4.3, "Using Free Space from an Active Partition"*).

Next, we will discuss a slightly more common situation.

A.1.4.2. Using Space from an Unused Partition

In this case, maybe you have one or more partitions that you do not use any longer. Perhaps you have dabbled with another operating system in the past, and the partition(s) you dedicated to it never seem to be used anymore. *Figure A.9, "Disk Drive With an Unused Partition"*, illustrates such a situation.
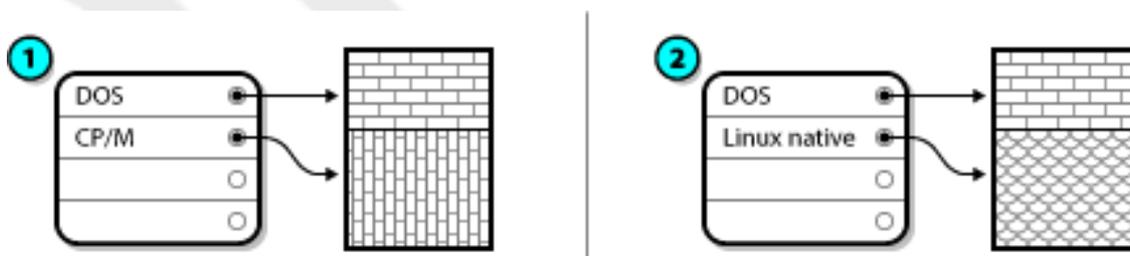
Figure A.9. Disk Drive With an Unused Partition

In *Figure A.9, "Disk Drive With an Unused Partition"*, *1* represents an unused partition and *2* represents reallocating an unused partition for Linux.

If you find yourself in this situation, you can use the space allocated to the unused partition. You first must delete the partition and then create the appropriate Linux partition(s) in its place. You can delete the unused partition and manually create new partitions during the installation process.

A.1.4.3. Using Free Space from an Active Partition

This is the most common situation. It is also, unfortunately, the hardest to handle. The main problem is that, even if you have enough free space, it is presently allocated to a partition that is already in use. If you purchased a computer with pre-installed software, the hard disk most likely has one massive partition holding the operating system and data.

Aside from adding a new hard drive to your system, you have two choices:

Destructive Repartitioning

> Basically, you delete the single large partition and create several smaller ones. As you might imagine, any data you had in the original partition is destroyed. This means that making a complete backup is necessary. For your own sake, make two backups, use verification (if available in your backup software), and try to read data from your backup *before* you delete the partition.

> ⚠️ **Warning**
>
> If there was an operating system of some type installed on that partition, it needs to be reinstalled as well. Be aware that some computers sold with pre-installed operating systems may not include the CD-ROM media to reinstall the original operating system. The best time to notice if this applies to your system is *before* you destroy your original partition and its operating system installation.

> After creating a smaller partition for your existing operating system, you can reinstall any software, restore your data, and start your Fedora installation. *Figure A.10, "Disk Drive Being Destructively Repartitioned"* shows this being done.
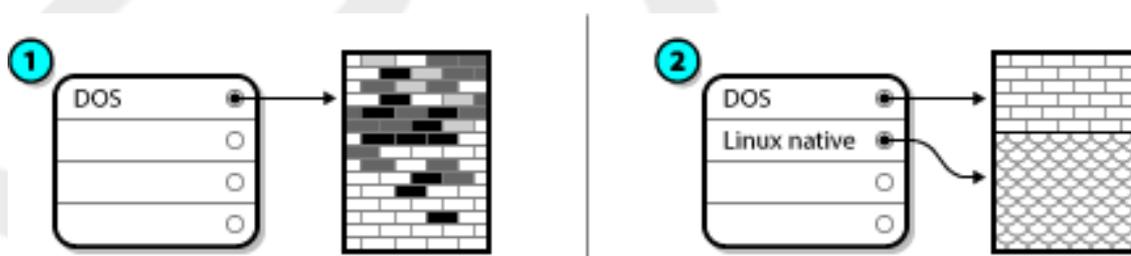
Figure A.10. Disk Drive Being Destructively Repartitioned

> In *Figure A.10, "Disk Drive Being Destructively Repartitioned"*, *1* represents before and *2* represents after.

> ⚠️ **Warning**
>
> As *Figure A.10, "Disk Drive Being Destructively Repartitioned"*, shows, any data present in the original partition is lost without proper backup!

Non-Destructive Repartitioning

Here, you run a program that does the seemingly impossible: it makes a big partition smaller without losing any of the files stored in that partition. Many people have found this method to be reliable and trouble-free. What software should you use to perform this feat? There are several disk management software products on the market. Do some research to find the one that is best for your situation.

While the process of non-destructive repartitioning is rather straightforward, there are a number of steps involved:

- Compress and backup existing data

- Resize the existing partition

- Create new partition(s)

Next we will look at each step in a bit more detail.

A.1.4.3.1. Compress existing data

As *Figure A.11, "Disk Drive Being Compressed"*, shows, the first step is to compress the data in your existing partition. The reason for doing this is to rearrange the data such that it maximizes the available free space at the "end" of the partition.
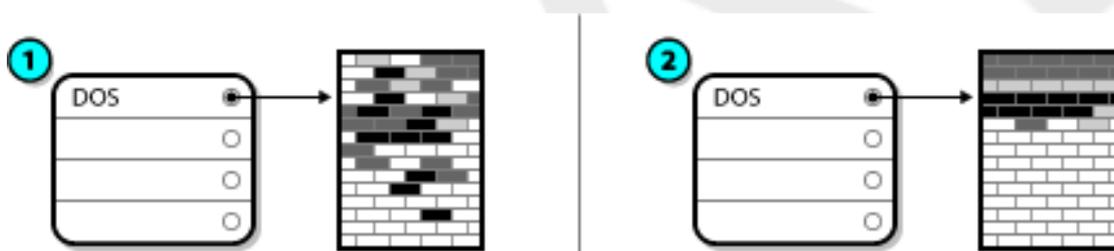
Figure A.11. Disk Drive Being Compressed

In *Figure A.11, "Disk Drive Being Compressed"*, *1* represents before and *2* represents after.

This step is crucial. Without it, the location of your data could prevent the partition from being resized to the extent desired. Note also that, for one reason or another, some data cannot be moved. If this is the case (and it severely restricts the size of your new partition(s)), you may be forced to destructively repartition your disk.

A.1.4.3.2. Resize the existing partition

Figure A.12, "Disk Drive with Partition Resized", shows the actual resizing process. While the actual result of the resizing operation varies depending on the software used, in most cases the newly freed space is used to create an unformatted partition of the same type as the original partition.
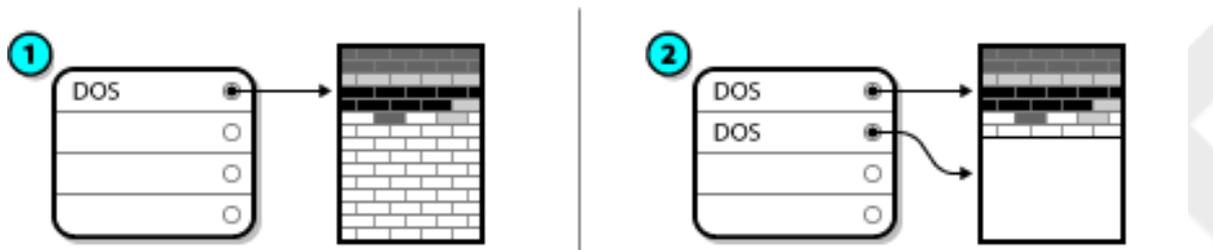
Figure A.12. Disk Drive with Partition Resized

In *Figure A.12, "Disk Drive with Partition Resized"*, *1* represents before and *2* represents after.

It is important to understand what the resizing software you use does with the newly freed space, so that you can take the appropriate steps. In the case we have illustrated, it would be best to delete the new DOS partition and create the appropriate Linux partition(s).

A.1.4.3.3. Create new partition(s)

As the previous step implied, it may or may not be necessary to create new partitions. However, unless your resizing software is Linux-aware, it is likely that you must delete the partition that was created during the resizing process. *Figure A.13, "Disk Drive with Final Partition Configuration"*, shows this being done.
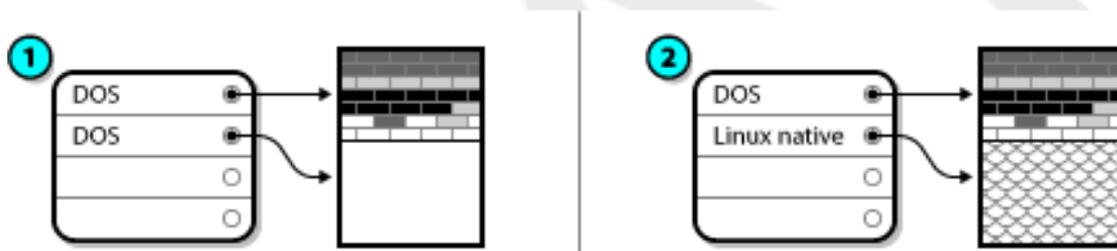
Figure A.13. Disk Drive with Final Partition Configuration

In *Figure A.13, "Disk Drive with Final Partition Configuration"*, *1* represents before and *2* represents after.

> **Note**
>
> The following information is specific to x86-based computers only.

As a convenience to our customers, we provide the `parted` utility. This is a freely available program that can resize partitions.

If you decide to repartition your hard drive with `parted`, it is important that you be familiar with disk storage and that you perform a backup of your computer data. You should make two copies of all the important data on your computer. These copies should be to removable media (such as tape, CD-ROM, or diskettes), and you should make sure they are readable before proceeding.

Should you decide to use `parted`, be aware that after `parted` runs you are left with *two* partitions: the one you resized, and the one `parted` created out of the newly freed space. If your goal is to use that space to install Fedora, you should delete the newly created partition, either by using the partitioning utility under your current operating system or while setting up partitions during installation.

A.1.5. Partition Naming Scheme

Linux refers to disk partitions using a combination of letters and numbers which may be confusing, particularly if you are used to the "C drive" way of referring to hard disks and their partitions. In the DOS/Windows world, partitions are named using the following method:

- Each partition's type is checked to determine if it can be read by DOS/Windows.

- If the partition's type is compatible, it is assigned a "drive letter." The drive letters start with a "C" and move on to the following letters, depending on the number of partitions to be labeled.

- The drive letter can then be used to refer to that partition as well as the file system contained on that partition.

Fedora uses a naming scheme that is more flexible and conveys more information than the approach used by other operating systems. The naming scheme is file-based, with file names in the form of `/dev/xxyN`.

Here is how to decipher the partition naming scheme:

`/dev/`

This is the name of the directory in which all device files reside. Since partitions reside on hard disks, and hard disks are devices, the files representing all possible partitions reside in `/dev/`.

`xx`

The first two letters of the partition name indicate the type of device on which the partition resides, usually either `hd` (for IDE disks) or `sd` (for SCSI disks).

`y`

This letter indicates which device the partition is on. For example, `/dev/hda` (the first IDE hard disk) or `/dev/sdb` (the second SCSI disk).

`N`

The final number denotes the partition. The first four (primary or extended) partitions are numbered `1` through `4`. Logical partitions start at `5`. So, for example, `/dev/hda3` is the third primary or extended partition on the first IDE hard disk, and `/dev/sdb6` is the second logical partition on the second SCSI hard disk.

> **Note**
>
> There is no part of this naming convention that is based on partition type; unlike DOS/ Windows, *all* partitions can be identified under Fedora. Of course, this does not mean that Fedora can access data on every type of partition, but in many cases it is possible to access data on a partition dedicated to another operating system.

Keep this information in mind; it makes things easier to understand when you are setting up the partitions Fedora requires.

A.1.6. Disk Partitions and Other Operating Systems

If your Fedora partitions are sharing a hard disk with partitions used by other operating systems, most of the time you will have no problems. However, there are certain combinations of Linux and other operating systems that require extra care.

A.1.7. Disk Partitions and Mount Points

One area that many people new to Linux find confusing is the matter of how partitions are used and accessed by the Linux operating system. In DOS/Windows, it is relatively simple: Each partition gets a "drive letter." You then use the correct drive letter to refer to files and directories on its corresponding partition.

This is entirely different from how Linux deals with partitions and, for that matter, with disk storage in general. The main difference is that each partition is used to form part of the storage necessary to support a single set of files and directories. This is done by associating a partition with a directory through a process known as *mounting*. Mounting a partition makes its storage available starting at the specified directory (known as a *mount point*).

For example, if partition `/dev/hda5` is mounted on `/usr/`, that would mean that all files and directories under `/usr/` physically reside on `/dev/hda5`. So the file `/usr/share/doc/FAQ/txt/Linux-FAQ` would be stored on `/dev/hda5`, while the file `/etc/gdm/custom.conf` would not.

Continuing our example, it is also possible that one or more directories below `/usr/` would be mount points for other partitions. For instance, a partition (say, `/dev/hda7`) could be mounted on `/usr/local/`, meaning that `/usr/local/man/whatis` would then reside on `/dev/hda7` rather than `/dev/hda5`.

A.1.8. How Many Partitions?

At this point in the process of preparing to install Fedora, you must give some consideration to the number and size of the partitions to be used by your new operating system. The question of "how many partitions" continues to spark debate within the Linux community and, without any end to the debate in sight, it is safe to say that there are probably as many partition layouts as there are people debating the issue.

Keeping this in mind, we recommend that, unless you have a reason for doing otherwise, you should at least create the following partitions: `swap`, `/boot/` (or a `/boot/efi/` partition for Itanium systems), a `/var/` partition for Itanium systems, and `/` (root).

For more information, refer to *Section 7.21.4, "Recommended Partitioning Scheme"*.

Appendix B. ISCSI disks

Internet Small Computer System Interface (iSCSI) is a protocol that allows computers to communicate with storage devices by SCSI requests and responses carried over TCP/IP. Because iSCSI is based on the standard SCSI protocols, it uses some terminology from SCSI. The device on the SCSI bus to which requests get sent (and which answers these requests) is known as the *target* and the device issuing requests is known as the *initiator*. In other words, an iSCSI disk is a target and the iSCSI software equivalent of a SCSI controller or SCSI Host Bus Adapter (HBA) is called an initiator. This appendix only covers Linux as an iSCSI initiator: how Linux uses iSCSI disks, but not how Linux hosts iSCSI disks.

Linux has a software iSCSI initiator in the kernel that takes the place and form of a SCSI HBA driver and therefore allows Linux to use iSCSI disks. However, as iSCSI is a fully network-based protocol, iSCSI initiator support needs more than just the ability to send SCSI packets over the network. Before Linux can use an iSCSI target, Linux must find the target on the network and make a connection to it. In some cases, Linux must send authentication information to gain access to the target. Linux must also detect any failure of the network connection and must establish a new connection, including logging in again if necessary.

The discovery, connection, and logging in is handled in userspace by the **iscsiadm** utility, and the error handling is also handled in userspace by **iscsid**.

Both **iscsiadm** and **iscsid** are part of the **iscsi-initiator-utils** package under Fedora.

B.1. iSCSI disks in anaconda

Anaconda can discover (and then log in to) iSCSI disks in two ways:

1. When anaconda starts, it checks if the BIOS or add-on boot ROMs of the system support *iSCSI Boot Firmware Table* (iBFT), a BIOS extension for systems which can boot from iSCSI. If the BIOS supports iBFT, **anaconda** will read the iSCSI target information for the configured boot disk from the BIOS and log in to this target, making it available as an installation target.

2. The initial partitioning screen presents you with an **Advanced storage configuration** button that allows you to add iSCSI target information like the discovery IP-address. **Anaconda** will probe the given IP-address and log in to any targets that it finds. See *Section 7.19, " Advanced Storage Options "* for the details that you can specify for iSCSI targets.

While **anaconda** uses **iscsiadm** to find and log into iSCSI targets, **iscsiadm** automatically stores any information about these targets iscsiadm iSCSI database. **Anaconda** then copies this database to the installed system and marks any iSCSI targets not used for / so that the system will automatically log in to them when it starts. If / is placed on an iSCSI target, **initrd** will log into this target and **anaconda** does not include this target in start up scripts to avoid multiple attempts to log into the same target.

If / is placed on an iSCSI target, **anaconda** sets **NetworkManager** to ignore any network interfaces that were active during the installation process. These interfaces will also be configured by **initrd** when the system starts. If **NetworkManager** were to reconfigure these interfaces, the system would lose its connection to /.

B.2. iSCSI disks during start up

ISCSI-related events might occur at a number of points while the system starts:

1. The init script in the **initrd** will log in to iSCSI targets used for `/` (if any). This is done using the **iscsistart** utility (which can do this without requiring **iscsid** to run).

2. When the root filesystem has been mounted and the various service initscripts get run, the **iscsid** initscript will get called. This script will then start **iscsid** if any iSCSI targets are used for `/`, or if any targets in the iSCSI database are marked to be logged in to automatically.

3. After the classic network service script has been run (or would have been run if enabled) the iscsi initscript will run. If the network is accessible, this will log in to any targets in the iSCSI database which are marked to be logged in to automatically. If the network is not accessible, this script will exit quietly.

4. When using **NetworkManager** to access the network (instead of the classic network service script), **NetworkManager** will call the iscsi initscript. See: `/etc/NetworkManager/dispatcher.d/04-iscsi`

> ### Important
> Because **NetworkManager** is installed in `/usr`, you cannot use it to configure network access if `/usr` is on network-attached storage such as an iSCSI target.

If **iscsid** is not needed as the system starts, it will not start automatically. If you start **iscsiadm**, **iscsiadm** will start **iscsid** in turn.

Appendix C. Disk Encryption Guide

C.1. What is block device encryption?

Block device encryption protects the data on a block device by encrypting it. To access the device's decrypted contents, a user must provide a passphrase or key as authentication. This provides additional security beyond existing OS security mechanisms in that it protects the device's contents even if it has been physically removed from the system.

C.2. Encrypting block devices using dm-crypt/LUKS

LUKS[1] (Linux Unified Key Setup) is a specification for block device encryption. It establishes an on-disk format for the data, as well as a passphrase/key management policy.

LUKS uses the kernel device mapper subsystem via the `dm-crypt` module. This arrangement provides a low-level mapping that handles encryption and decryption of the device's data. User-level operations, such as creating and accessing encrypted devices, are accomplished through the use of the `cryptsetup` utility.

C.2.1. Overview of LUKS

- What LUKS does:
  - LUKS encrypts entire block devices
    - LUKS is thereby well-suited for protecting the contents of mobile devices such as:
      - Removable storage media

      - Laptop disk drives

  - The underlying contents of the encrypted block device are arbitrary.
    - This makes it useful for encrypting `swap` devices.

    - This can also be useful with certain databases that use specially formatted block devices for data storage.

  - LUKS uses the existing device mapper kernel subsystem.
    - This is the same subsystem used by LVM, so it is well tested.

  - LUKS provides passphrase strengthening.
    - This protects against dictionary attacks.

  - LUKS devices contain multiple key slots.
    - This allows users to add backup keys/passphrases.

- What LUKS does *not* do:
  - LUKS is not well-suited for applications requiring many (more than eight) users to have distinct access keys to the same device.

  - LUKS is not well-suited for applications requiring file-level encryption.

---

[1] http://luks.endorphin.org

C.2.2. How will I access the encrypted devices after installation? (System Startup)

During system startup you will be presented with a passphrase prompt. After the correct passphrase has been provided the system will continue to boot normally. If you used different passphrases for multiple encrypted devices you may need to enter more than one passphrase during the startup.

> **Tip**
>
> Consider using the same passphrase for all encrypted block devices in a given system. This will simplify system startup and you will have fewer passphrases to remember. Just make sure you choose a good passphrase!

C.2.3. Choosing a Good Passphrase

While dm-crypt/LUKS supports both keys and passphrases, the anaconda installer only supports the use of passphrases for creating and accessing encrypted block devices during installation.

LUKS does provide passphrase strengthening but it is still a good idea to choose a good (meaning "difficult to guess") passphrase. Note the use of the term "passphrase", as opposed to the term "password". This is intentional. Providing a phrase containing multiple words to increase the security of your data is important.

C.3. Creating Encrypted Block Devices in Anaconda

You can create encrypted devices during system installation. This allows you to easily configure a system with encrypted partitions.

To enable block device encryption, check the "Encrypt System" checkbox when selecting automatic partitioning or the "Encrypt" checkbox when creating an individual partition, software RAID array, or logical volume. After you finish partitioning, you will be prompted for an encryption passphrase. This passphrase will be required to access the encrypted devices. If you have pre-existing LUKS devices and provided correct passphrases for them earlier in the install process the passphrase entry dialog will also contain a checkbox. Checking this checkbox indicates that you would like the new passphrase to be added to an available slot in each of the pre-existing encrypted block devices.

> **Tip**
>
> Checking the "Encrypt System" checkbox on the "Automatic Partitioning" screen and then choosing "Create custom layout" does not cause any block devices to be encrypted automatically.

> **Tip**
>
> You can use `kickstart` to set a separate passphrase for each new encrypted block device.

C.3.1. What Kinds of Block Devices Can Be Encrypted?

Most types of block devices can be encrypted using LUKS. From anaconda you can encrypt partitions, LVM physical volumes, LVM logical volumes, and software RAID arrays.

C.3.2. Limitations of Anaconda's Block Device Encryption Support

This section is about Anaconda's Block Device Encryption Support

C.4. Creating Encrypted Block Devices on the Installed System After Installation

Encrypted block devices can be created and configured after installation.

C.4.1. Create the block devices

Create the block devices you want to encrypt by using **parted**, **pvcreate**, **lvcreate** and **mdadm**.

C.4.2. Optional: Fill the device with random data

Filling <device> (eg: **/dev/sda3**) with random data before encrypting it greatly increases the strength of the encryption. The downside is that it can take a very long time.

> ⚠ **Warning**
>
> The commands below will destroy any existing data on the device.

- The best way, which provides high quality random data but takes a long time (several minutes per gigabyte on most systems):

```
dd if=/dev/urandom of=<device>
```

- Fastest way, which provides lower quality random data:

```
badblocks -c 10240 -s -w -t random -v <device>
```

C.4.3. Format the device as a dm-crypt/LUKS encrypted device

> ⚠ **Warning**
>
> The command below will destroy any existing data on the device.

```
cryptsetup luksFormat <device>
```

> 📌 **Tip**
>
> For more information, read the **cryptsetup(8)** man page.

After supplying the passphrase twice the device will be formatted for use. To verify, use the following command:

```
cryptsetup isLuks <device> && echo Success
```

To see a summary of the encryption information for the device, use the following command:

```
cryptsetup luksDump <device>
```

C.4.4. Create a mapping to allow access to the device's decrypted contents

To access the device's decrypted contents, a mapping must be established using the kernel **device-mapper**.

It is useful to choose a meaningful name for this mapping. LUKS provides a UUID (Universally Unique Identifier) for each device. This, unlike the device name (eg: **/dev/sda3**), is guaranteed to remain constant as long as the LUKS header remains intact. To find a LUKS device's UUID, run the following command:

```
cryptsetup luksUUID <device>
```

An example of a reliable, informative and unique mapping name would be **luks-<uuid>**, where <uuid> is replaced with the device's LUKS UUID (eg: **luks-50ec957a-5b5a-47ee-85e6-f8085bbc97a8**). This naming convention might seem unwieldy but is it not necessary to type it often.

```
cryptsetup luksOpen <device> <name>
```

There should now be a device node, **/dev/mapper/<name>**, which represents the decrypted device. This block device can be read from and written to like any other unencrypted block device.

To see some information about the mapped device, use the following command:

```
dmsetup info <name>
```

> 📌 **Tip**
>
> For more information, read the **dmsetup(8)** man page.

C.4.5. Create filesystems on the mapped device, or continue to build complex storage structures using the mapped device

Use the mapped device node (**/dev/mapper/<name>**) as any other block device. To create an **ext2** filesystem on the mapped device, use the following command:

```
mke2fs /dev/mapper/<name>
```

To mount this filesystem on **/mnt/test**, use the following command:

> **Important**
>
> The directory **/mnt/test** must exist before executing this command.

```
mount /dev/mapper/<name> /mnt/test
```

C.4.6. Add the mapping information to **/etc/crypttab**

In order for the system to set up a mapping for the device, an entry must be present in the **/etc/crypttab** file. If the file doesn't exist, create it and change the owner and group to root (**root:root**) and change the mode to **0744**. Add a line to the file with the following format:

```
<name>   <device>   none
```

The <device> field should be given in the form "UUID=<luks_uuid>", where <luks_uuid> is the LUKS uuid as given by the command **cryptsetup luksUUID <device>**. This ensures the correct device will be identified and used even if the device node (eg: **/dev/sda5**) changes.

> **Tip**
>
> For details on the format of the **/etc/crypttab** file, read the **crypttab(5)** man page.

C.4.7. Add an entry to **/etc/fstab**

Add an entry to /etc/fstab. This is only necessary if you want to establish a persistent association between the device and a mountpoint. Use the decrypted device, **/dev/mapper/<name>** in the **/etc/fstab** file.

In many cases it is desirable to list devices in **/etc/fstab** by UUID or by a filesystem label. The main purpose of this is to provide a constant identifier in the event that the device name (eg: **/dev/sda4**) changes. LUKS device names in the form of **/dev/mapper/luks-<luks_uuid>** are based only on the device's LUKS UUID, and are therefore guaranteed to remain constant. This fact makes them suitable for use in **/etc/fstab**.

> **Title**
>
> For details on the format of the `/etc/fstab` file, read the `fstab(5)` man page.

C.5. Common Post-Installation Tasks

The following sections are about common post-installation tasks.

C.5.1. Set a randomly generated key as an additional way to access an encrypted block device

These sections are about generating keys and adding keys.

C.5.1.1. Generate a key

This will generate a 256-bit key in the file `$HOME/keyfile`.

```
dd if=/dev/urandom of=$HOME/keyfile bs=32 count=1
chmod 600 $HOME/keyfile
```

C.5.1.2. Add the key to an available keyslot on the encrypted device

```
cryptsetup luksAddKey <device> ~/keyfile
```

C.5.2. Add a new passphrase to an existing device

```
cryptsetup luksAddKey <device>
```

After being prompted for any one of the existing passprases for authentication, you will be prompted to enter the new passphrase.

C.5.3. Remove a passphrase or key from a device

```
cryptsetup luksRemoveKey <device>
```

You will be prompted for the passphrase you wish to remove and then for any one of the remaining passphrases for authentication.

Appendix D. Understanding LVM

LVM (Logical Volume Management) partitions provide a number of advantages over standard partitions. LVM partitions are formatted as
physical volumes. One or more physical volumes are combined to form a
volume group. Each volume group's total storage is then divided into one or more
logical volumes. The logical volumes function much like standard partitions. They have a file system type, such as `ext3`, and a mount point.

> ### The `/boot` Partition and LVM
> The boot loader cannot read LVM volumes. You must make a standard, non-LVM disk partition for your `/boot` partition.

To understand LVM better, imagine the physical volume as a pile of *blocks.* A block is simply a storage unit used to store data. Several piles of blocks can be combined to make a much larger pile, just as physical volumes are combined to make a volume group. The resulting pile can be subdivided into several smaller piles of arbitrary size, just as a volume group is allocated to several logical volumes.

An administrator may grow or shrink logical volumes without destroying data, unlike standard disk partitions. If the physical volumes in a volume group are on separate drives or RAID arrays then administrators may also spread a logical volume across the storage devices.

You may lose data if you shrink a logical volume to a smaller capacity than the data on the volume requires. To ensure maximum flexibility, create logical volumes to meet your current needs, and leave excess storage capacity unallocated. You may safely grow logical volumes to use unallocated space, as your needs dictate.

> ### LVM and the Default Partition Layout
> By default, the installation process creates `/` and swap partitions within LVM volumes, with a separate `/boot` partition.

Appendix E. The GRUB Boot Loader

When a computer running Linux is turned on, the operating system is loaded into memory by a special program called a *boot loader*. A boot loader usually exists on the system's primary hard drive (or other media device) and has the sole responsibility of loading the Linux kernel with its required files or (in some cases) other operating systems into memory.

This appendix discusses commands and configuration options for the GRUB boot loader included with Fedora for the x86 architecture.

E.1. GRUB

The *GNU GRand Unified Boot loader* (GRUB) is a program which enables the selection of the installed operating system or kernel to be loaded at system boot time. It also allows the user to pass arguments to the kernel.

E.1.1. GRUB and the x86 Boot Process

This section discusses the specific role GRUB plays when booting an x86 system. For a look at the overall boot process, refer to *Section F.2, "A Detailed Look at the Boot Process"*.

GRUB loads itself into memory in the following stages:

1. *The Stage 1 or primary boot loader is read into memory by the BIOS from the MBR* [1]*.* The primary boot loader exists on less than 512 bytes of disk space within the MBR and is capable of loading either the Stage 1.5 or Stage 2 boot loader.

2. *The Stage 1.5 boot loader is read into memory by the Stage 1 boot loader, if necessary.* Some hardware requires an intermediate step to get to the Stage 2 boot loader. This is sometimes true when the `/boot/` partition is above the 1024 cylinder head of the hard drive or when using LBA mode. The Stage 1.5 boot loader is found either on the `/boot/` partition or on a small part of the MBR and the `/boot/` partition.

3. *The Stage 2 or secondary boot loader is read into memory.* The secondary boot loader displays the GRUB menu and command environment. This interface allows the user to select which kernel or operating system to boot, pass arguments to the kernel, or look at system parameters.

4. *The secondary boot loader reads the operating system or kernel as well as the contents of `/boot/sysroot/` into memory.* Once GRUB determines which operating system or kernel to start, it loads it into memory and transfers control of the machine to that operating system.

The method used to boot Linux is called *direct loading* because the boot loader loads the operating system directly. There is no intermediary between the boot loader and the kernel.

The boot process used by other operating systems may differ. For example, the Microsoft® Windows® operating system, as well as other operating systems, are loaded using *chain loading*. Under this method, the MBR points to the first sector of the partition holding the operating system, where it finds the files necessary to actually boot that operating system.

GRUB supports both direct and chain loading boot methods, allowing it to boot almost any operating system.

> ⚠️ **Warning**
>
> During installation, Microsoft's DOS and Windows installation programs completely overwrite the MBR, destroying any existing boot loaders. If creating a dual-boot system, it is best to install the Microsoft operating system first.

E.1.2. Features of GRUB

GRUB contains several features that make it preferable to other boot loaders available for the x86 architecture. Below is a partial list of some of the more important features:

- *GRUB provides a true command-based, pre-OS environment on x86 machines.* This feature affords the user maximum flexibility in loading operating systems with specified options or gathering information about the system. For years, many non-x86 architectures have employed pre-OS environments that allow system booting from a command line.

- *GRUB supports Logical Block Addressing (LBA) mode.* LBA places the addressing conversion used to find files in the hard drive's firmware, and is used on many IDE and all SCSI hard devices. Before LBA, boot loaders could encounter the 1024-cylinder BIOS limitation, where the BIOS could not find a file after the 1024 cylinder head of the disk. LBA support allows GRUB to boot operating systems from partitions beyond the 1024-cylinder limit, so long as the system BIOS supports LBA mode. Most modern BIOS revisions support LBA mode.

- *GRUB can read ext2 partitions.* This functionality allows GRUB to access its configuration file, `/boot/grub/grub.conf`, every time the system boots, eliminating the need for the user to write a new version of the first stage boot loader to the MBR when configuration changes are made. The only time a user needs to reinstall GRUB on the MBR is if the physical location of the `/boot/` partition is moved on the disk. For details on installing GRUB to the MBR, refer to *Section E.2, "Installing GRUB"*.

E.2. Installing GRUB

If GRUB was not installed during the installation process, it can be installed afterward. Once installed, it automatically becomes the default boot loader.

Before installing GRUB, make sure to use the latest GRUB package available or use the GRUB package from the installation CD-ROMs.

Once the GRUB package is installed, open a root shell prompt and run the command `/sbin/grub-install <location>`, where `<location>` is the location that the GRUB Stage 1 boot loader should be installed. For example, the following command installs GRUB to the MBR of the master IDE device on the primary IDE bus:

```
/sbin/grub-install /dev/hda
```

The next time the system boots, the GRUB graphical boot loader menu appears before the kernel loads into memory.

> **Important**
>
> If GRUB is installed on a RAID 1 array, the system may become unbootable in the event of disk failure.

E.3. GRUB Terminology

One of the most important things to understand before using GRUB is how the program refers to devices, such as hard drives and partitions. This information is particularly important when configuring GRUB to boot multiple operating systems.

E.3.1. Device Names

When referring to a specific device with GRUB, do so using the following format (note that the parentheses and comma are very important syntactically):

(*<type-of-device><bios-device-number>,<partition-number>*)

The *<type-of-device>* specifies the type of device from which GRUB boots. The two most common options are **hd** for a hard disk or **fd** for a 3.5 diskette. A lesser used device type is also available called **nd** for a network disk. Instructions on configuring GRUB to boot over the network are available online at *http://www.gnu.org/software/grub/manual/*.

The *<bios-device-number>* is the BIOS device number. The primary IDE hard drive is numbered **0** and a secondary IDE hard drive is numbered **1**. This syntax is roughly equivalent to that used for devices by the kernel. For example, the **a** in **hda** for the kernel is analogous to the **0** in **hd0** for GRUB, the **b** in **hdb** is analogous to the **1** in **hd1**, and so on.

The *<partition-number>* specifies the number of a partition on a device. Like the *<bios-device-number>*, most types of partitions are numbered starting at **0**. However, BSD partitions are specified using letters, with **a** corresponding to **0**, **b** corresponding to **1**, and so on.

> **Note**
>
> The numbering system for devices under GRUB always begins with **0**, not **1**. Failing to make this distinction is one of the most common mistakes made by new users.

To give an example, if a system has more than one hard drive, GRUB refers to the first hard drive as **(hd0)** and the second as **(hd1)**. Likewise, GRUB refers to the first partition on the first drive as **(hd0,0)** and the third partition on the second hard drive as **(hd1,2)**.

In general the following rules apply when naming devices and partitions under GRUB:

- It does not matter if system hard drives are IDE or SCSI, all hard drives begin with the letters **hd**. The letters **fd** are used to specify 3.5 diskettes.

- To specify an entire device without respect to partitions, leave off the comma and the partition number. This is important when telling GRUB to configure the MBR for a particular disk. For example, **(hd0)** specifies the MBR on the first device and **(hd3)** specifies the MBR on the fourth device.

- If a system has multiple drive devices, it is very important to know how the drive boot order is set in the BIOS. This is a simple task if a system has only IDE or SCSI drives, but if there is a mix of devices, it becomes critical that the type of drive with the boot partition be accessed first.

E.3.2. File Names and Blocklists

When typing commands to GRUB that reference a file, such as a menu list, it is necessary to specify an absolute file path immediately after the device and partition numbers.

The following illustrates the structure of such a command:

`(<device-type><device-number>,<partition-number>)</path/to/file>`

In this example, replace `<device-type>` with **hd**, **fd**, or **nd**. Replace `<device-number>` with the integer for the device. Replace `</path/to/file>` with an absolute path relative to the top-level of the device.

It is also possible to specify files to GRUB that do not actually appear in the file system, such as a chain loader that appears in the first few blocks of a partition. To load such files, provide a *blocklist* that specifies block by block where the file is located in the partition. Since a file is often comprised of several different sets of blocks, blocklists use a special syntax. Each block containing the file is specified by an offset number of blocks, followed by the number of blocks from that offset point. Block offsets are listed sequentially in a comma-delimited list.

The following is a sample blocklist:

```
0+50,100+25,200+1
```

This sample blocklist specifies a file that starts at the first block on the partition and uses blocks 0 through 49, 100 through 124, and 200.

Knowing how to write blocklists is useful when using GRUB to load operating systems which require chain loading. It is possible to leave off the offset number of blocks if starting at block 0. As an example, the chain loading file in the first partition of the first hard drive would have the following name:

```
(hd0,0)+1
```

The following shows the **chainloader** command with a similar blocklist designation at the GRUB command line after setting the correct device and partition as root:

```
chainloader +1
```

E.3.3. The Root File System and GRUB

The use of the term *root file system* has a different meaning in regard to GRUB. It is important to remember that GRUB's root file system has nothing to do with the Linux root file system.

The GRUB root file system is the top level of the specified device. For example, the image file `(hd0,0)/grub/splash.xpm.gz` is located within the `/grub/` directory at the top-level (or root) of the `(hd0,0)` partition (which is actually the `/boot/` partition for the system).

Next, the `kernel` command is executed with the location of the kernel file as an option. Once the Linux kernel boots, it sets up the root file system that Linux users are familiar with. The original GRUB root file system and its mounts are forgotten; they only existed to boot the kernel file.

Refer to the `root` and `kernel` commands in *Section E.5, "GRUB Commands"* for more information.

E.4. GRUB Interfaces

GRUB features three interfaces which provide different levels of functionality. Each of these interfaces allows users to boot the Linux kernel or another operating system.

The interfaces are as follows:

> **Note**
>
> The following GRUB interfaces can only be accessed by pressing any key within the three seconds of the GRUB menu bypass screen.

Menu Interface

This is the default interface shown when GRUB is configured by the installation program. A menu of operating systems or preconfigured kernels are displayed as a list, ordered by name. Use the arrow keys to select an operating system or kernel version and press the **Enter** key to boot it. If you do nothing on this screen, then after the time out period expires GRUB will load the default option.

Press the **e** key to enter the entry editor interface or the **c** key to load a command line interface.

Refer to *Section E.6, "GRUB Menu Configuration File"* for more information on configuring this interface.

Menu Entry Editor Interface

To access the menu entry editor, press the **e** key from the boot loader menu. The GRUB commands for that entry are displayed here, and users may alter these command lines before booting the operating system by adding a command line (**o** inserts a new line after the current line and **O** inserts a new line before it), editing one (**e**), or deleting one (**d**).

After all changes are made, the **b** key executes the commands and boots the operating system. The **Esc** key discards any changes and reloads the standard menu interface. The **c** key loads the command line interface.

> **Note**
>
> For information about changing runlevels using the GRUB menu entry editor, refer to *Section E.7, "Changing Runlevels at Boot Time"*.

Command Line Interface

The command line interface is the most basic GRUB interface, but it is also the one that grants the most control. The command line makes it possible to type any relevant GRUB commands

followed by the **Enter** key to execute them. This interface features some advanced shell-like features, including **Tab** key completion based on context, and **Ctrl** key combinations when typing commands, such as **Ctrl+a** to move to the beginning of a line and **Ctrl+e** to move to the end of a line. In addition, the arrow, **Home**, **End**, and **Delete** keys work as they do in the **bash** shell.

Refer to *Section E.5, "GRUB Commands"* for a list of common commands.

E.4.1. Interfaces Load Order

When GRUB loads its second stage boot loader, it first searches for its configuration file. Once found, the menu interface bypass screen is displayed. If a key is pressed within three seconds, GRUB builds a menu list and displays the menu interface. If no key is pressed, the default kernel entry in the GRUB menu is used.

If the configuration file cannot be found, or if the configuration file is unreadable, GRUB loads the command line interface, allowing the user to type commands to complete the boot process.

If the configuration file is not valid, GRUB prints out the error and asks for input. This helps the user see precisely where the problem occurred. Pressing any key reloads the menu interface, where it is then possible to edit the menu option and correct the problem based on the error reported by GRUB. If the correction fails, GRUB reports an error and reloads the menu interface.

E.5. GRUB Commands

GRUB allows a number of useful commands in its command line interface. Some of the commands accept options after their name; these options should be separated from the command and other options on that line by space characters.

The following is a list of useful commands:

- **boot** — Boots the operating system or chain loader that was last loaded.

- **chainloader </path/to/file>** — Loads the specified file as a chain loader. If the file is located on the first sector of the specified partition, use the blocklist notation, **+1**, instead of the file name.

  The following is an example **chainloader** command:

  ```
  chainloader +1
  ```

- **displaymem** — Displays the current use of memory, based on information from the BIOS. This is useful to determine how much RAM a system has prior to booting it.

- **initrd </path/to/initrd>** — Enables users to specify an initial RAM disk to use when booting. An **initrd** is necessary when the kernel needs certain modules in order to boot properly, such as when the root partition is formatted with the ext3 file system.

  The following is an example **initrd** command:

  ```
  initrd /initrd-2.6.8-1.523.img
  ```

- **install** *<stage-1>* *<install-disk>* *<stage-2>* **p** *config-file* — Installs GRUB to the system MBR.

  - *<stage-1>* — Signifies a device, partition, and file where the first boot loader image can be found, such as **(hd0,0)/grub/stage1**.

  - *<install-disk>* — Specifies the disk where the stage 1 boot loader should be installed, such as **(hd0)**.

  - *<stage-2>* — Passes the stage 2 boot loader location to the stage 1 boot loader, such as **(hd0,0)/grub/stage2**.

  - p *<config-file>* — This option tells the **install** command to look for the menu configuration file specified by *<config-file>*, such as **(hd0,0)/grub/grub.conf**.

> ⚠ **Warning**
> The **install** command overwrites any information already located on the MBR.

- **kernel** *</path/to/kernel>* *<option-1>* *<option-N>* ... — Specifies the kernel file to load when booting the operating system. Replace *</path/to/kernel>* with an absolute path from the partition specified by the root command. Replace *<option-1>* with options for the Linux kernel, such as **root=/dev/VolGroup00/LogVol00** to specify the device on which the root partition for the system is located. Multiple options can be passed to the kernel in a space separated list.

  The following is an example **kernel** command:

  ```
  kernel /vmlinuz-2.6.8-1.523 ro root=/dev/VolGroup00/LogVol00
  ```

  The option in the previous example specifies that the root file system for Linux is located on the **hda5** partition.

- **root (***<device-type><device-number>,<partition>***)** — Configures the root partition for GRUB, such as **(hd0,0)**, and mounts the partition.

  The following is an example **root** command:

  ```
  root (hd0,0)
  ```

- **rootnoverify (***<device-type><device-number>,<partition>***)** — Configures the root partition for GRUB, just like the **root** command, but does not mount the partition.

Other commands are also available; type **help --all** for a full list of commands. For a description of all GRUB commands, refer to the documentation available online at *http://www.gnu.org/software/grub/manual/*.

E.6. GRUB Menu Configuration File

The configuration file (`/boot/grub/grub.conf`), which is used to create the list of operating systems to boot in GRUB's menu interface, essentially allows the user to select a pre-set group of commands to execute. The commands given in *Section E.5, "GRUB Commands"* can be used, as well as some special commands that are only available in the configuration file.

E.6.1. Configuration File Structure

The GRUB menu interface configuration file is `/boot/grub/grub.conf`. The commands to set the global preferences for the menu interface are placed at the top of the file, followed by stanzas for each operating kernel or operating system listed in the menu.

The following is a very basic GRUB menu configuration file designed to boot either Fedora or Microsoft Windows Vista:

```
default=0
timeout=10
splashimage=(hd0,0)/grub/splash.xpm.gz
hiddenmenu
title Fedora (2.6.27.19-170.2.35.fc10.i686)
root (hd0,1)
kernel /vmlinuz-2.6.27.19-170.2.35.fc10.i686 ro root=UUID=04a07c13-
e6bf-6d5a-b207-002689545705 rhgb quiet
initrd /initrd-2.6.27.19-170.2.35.fc10.i686.img


# section to load Windows
title Windows
rootnoverify (hd0,0)
chainloader +1
```

This file configures GRUB to build a menu with Fedora as the default operating system and sets it to autoboot after 10 seconds. Two sections are given, one for each operating system entry, with commands specific to the system disk partition table.

> **Note**
>
> Note that the default is specified as an integer. This refers to the first `title` line in the GRUB configuration file. For the `Windows` section to be set as the default in the previous example, change the `default=0` to `default=1`.

Configuring a GRUB menu configuration file to boot multiple operating systems is beyond the scope of this chapter. Consult *Section E.8, "Additional Resources"* for a list of additional resources.

E.6.2. Configuration File Directives

The following are directives commonly used in the GRUB menu configuration file:

- `chainloader </path/to/file>` — Loads the specified file as a chain loader. Replace *</path/to/file>* with the absolute path to the chain loader. If the file is located on the first sector of the specified partition, use the blocklist notation, `+1`.

- **color** *<normal-color> <selected-color>* — Allows specific colors to be used in the menu, where two colors are configured as the foreground and background. Use simple color names such as **red/black**. For example:

```
color red/black green/blue
```

- **default=***<integer>* — Replace *<integer>* with the default entry title number to be loaded if the menu interface times out.

- **fallback=***<integer>* — Replace *<integer>* with the entry title number to try if the first attempt fails.

- **hiddenmenu** — Prevents the GRUB menu interface from being displayed, loading the **default** entry when the **timeout** period expires. The user can see the standard GRUB menu by pressing the **Esc** key.

- **initrd** *</path/to/initrd>* — Enables users to specify an initial RAM disk to use when booting. Replace *</path/to/initrd>* with the absolute path to the initial RAM disk.

- **kernel** *</path/to/kernel> <option-1> <option-N>* — Specifies the kernel file to load when booting the operating system. Replace *</path/to/kernel>* with an absolute path from the partition specified by the root directive. Multiple options can be passed to the kernel when it is loaded.

- **password=***<password>* — Prevents a user who does not know the password from editing the entries for this menu option.

  Optionally, it is possible to specify an alternate menu configuration file after the **password=***<password>* directive. In this case, GRUB restarts the second stage boot loader and uses the specified alternate configuration file to build the menu. If an alternate menu configuration file is left out of the command, a user who knows the password is allowed to edit the current configuration file.

  For more information about securing GRUB, refer to the chapter titled *Workstation Security* in the *Red Hat Enterprise Linux Deployment Guide*.

- **root (***<device-type><device-number>,<partition>***)** — Configures the root partition for GRUB, such as **(hd0,0)**, and mounts the partition.

- **rootnoverify (***<device-type><device-number>,<partition>***)** — Configures the root partition for GRUB, just like the **root** command, but does not mount the partition.

- **timeout=***<integer>* — Specifies the interval, in seconds, that GRUB waits before loading the entry designated in the **default** command.

- **splashimage=***<path-to-image>* — Specifies the location of the splash screen image to be used when GRUB boots.

- **title** *group-title* — Specifies a title to be used with a particular group of commands used to load a kernel or operating system.

To add human-readable comments to the menu configuration file, begin the line with the hash mark character (**#**).

E.7. Changing Runlevels at Boot Time

Under Fedora, it is possible to change the default runlevel at boot time.

To change the runlevel of a single boot session, use the following instructions:

- When the GRUB menu bypass screen appears at boot time, press any key to enter the GRUB menu (within the first three seconds).

- Press the `a` key to append to the `kernel` command.

- Add `<space><runlevel>` at the end of the boot options line to boot to the desired runlevel. For example, the following entry would initiate a boot process into runlevel 3:

```
grub append> ro root=/dev/VolGroup00/LogVol00 rhgb quiet 3
```

E.8. Additional Resources

This chapter is only intended as an introduction to GRUB. Consult the following resources to discover more about how GRUB works.

E.8.1. Installed Documentation

- `/usr/share/doc/grub-<version-number>/` — This directory contains good information about using and configuring GRUB, where `<version-number>` corresponds to the version of the GRUB package installed.

- `info grub` — The GRUB info page contains a tutorial, a user reference manual, a programmer reference manual, and a FAQ document about GRUB and its usage.

E.8.2. Useful Websites

- *http://www.gnu.org/software/grub/*[2] — The home page of the GNU GRUB project. This site contains information concerning the state of GRUB development and an FAQ.

- *http://kbase.redhat.com/faq/docs/DOC-6864* — Details booting operating systems other than Linux.

E.8.3. Related Books

- *Red Hat Enterprise Linux Deployment Guide*; Red Hat, Inc. — The *Workstation Security* chapter explains, in a concise manner, how to secure the GRUB boot loader.

Appendix F. Boot Process, Init, and Shutdown

An important and powerful aspect of Fedora is the open, user-configurable method it uses for starting the operating system. Users are free to configure many aspects of the boot process, including specifying the programs launched at boot-time. Similarly, system shutdown gracefully terminates processes in an organized and configurable way, although customization of this process is rarely required.

Understanding how the boot and shutdown processes work not only allows customization, but also makes it easier to troubleshoot problems related to starting or shutting down the system.

F.1. The Boot Process

Below are the basic stages of the boot process for an x86 system:

1. The system BIOS checks the system and launches the first stage boot loader on the MBR of the primary hard disk.

2. The first stage boot loader loads itself into memory and launches the second stage boot loader from the **/boot/** partition.

3. The second stage boot loader loads the kernel into memory, which in turn loads any necessary modules and mounts the root partition read-only.

4. The kernel transfers control of the boot process to the **/sbin/init** program.

5. The **/sbin/init** program loads all services and user-space tools, and mounts all partitions listed in **/etc/fstab**.

6. The user is presented with a login screen for the freshly booted Linux system.

Because configuration of the boot process is more common than the customization of the shutdown process, the remainder of this chapter discusses in detail how the boot process works and how it can be customized to suite specific needs.

F.2. A Detailed Look at the Boot Process

The beginning of the boot process varies depending on the hardware platform being used. However, once the kernel is found and loaded by the boot loader, the default boot process is identical across all architectures. This chapter focuses primarily on the x86 architecture.

F.2.1. The BIOS

When an x86 computer is booted, the processor looks at the end of system memory for the *Basic Input/Output System* or *BIOS* program and runs it. The BIOS controls not only the first step of the boot process, but also provides the lowest level interface to peripheral devices. For this reason it is written into read-only, permanent memory and is always available for use.

Other platforms use different programs to perform low-level tasks roughly equivalent to those of the BIOS on an x86 system. For instance, Itanium-based computers use the *Extensible Firmware Interface* (*EFI*) *Shell*.

Once loaded, the BIOS tests the system, looks for and checks peripherals, and then locates a valid device with which to boot the system. Usually, it checks any diskette drives and CD-ROM drives present for bootable media, then, failing that, looks to the system's hard drives. In most cases, the order of the drives searched while booting is controlled with a setting in the BIOS, and it looks on the master IDE device on the primary IDE bus. The BIOS then loads into memory whatever program is residing in the first sector of this device, called the *Master Boot Record* or *MBR*. The MBR is only 512 bytes in size and contains machine code instructions for booting the machine, called a boot loader, along with the partition table. Once the BIOS finds and loads the boot loader program into memory, it yields control of the boot process to it.

F.2.2. The Boot Loader

This section looks at the default boot loader for the x86 platform, GRUB. For more information about configuring and using GRUB, see *Appendix E, The GRUB Boot Loader*.

A boot loader for the x86 platform is broken into at least two stages. The first stage is a small machine code binary on the MBR. Its sole job is to locate the second stage boot loader and load the first part of it into memory.

GRUB has the advantage of being able to read ext2 and ext3 [1] partitions and load its configuration file — `/boot/grub/grub.conf` — at boot time. Refer to *Section E.6, "GRUB Menu Configuration File"* for information on how to edit this file.

> **Note**
>
> If upgrading the kernel using the **Red Hat Update Agent**, the boot loader configuration file is updated automatically. More information on Red Hat Network can be found online at the following URL: *https://rhn.redhat.com/*.

Once the second stage boot loader is in memory, it presents the user with a graphical screen showing the different operating systems or kernels it has been configured to boot. On this screen a user can use the arrow keys to choose which operating system or kernel they wish to boot and press **Enter**. If no key is pressed, the boot loader loads the default selection after a configurable period of time has passed.

Once the second stage boot loader has determined which kernel to boot, it locates the corresponding kernel binary in the `/boot/` directory. The kernel binary is named using the following format — `/boot/vmlinuz-<kernel-version>` file (where `<kernel-version>` corresponds to the kernel version specified in the boot loader's settings).

For instructions on using the boot loader to supply command line arguments to the kernel, refer to *Appendix E, The GRUB Boot Loader*. For information on changing the runlevel at the boot loader prompt, refer *Section E.7, "Changing Runlevels at Boot Time"*.

The boot loader then places one or more appropriate *initramfs* images into memory. Next, the kernel decompresses these images from memory to `/sysroot/`, a RAM-based virtual file system, via `cpio`. The `initramfs` is used by the kernel to load drivers and modules necessary to boot the system. This is particularly important if SCSI hard drives are present or if the systems use the ext3 file system.

---

[1] GRUB reads ext3 file systems as ext2, disregarding the journal file. Refer to the chapter titled *The ext3 File System* in the *Red Hat Enterprise Linux Deployment Guide* for more information on the ext3 file system.

Once the kernel and the **initramfs** image(s) are loaded into memory, the boot loader hands control of the boot process to the kernel.

For a more detailed overview of the GRUB boot loader, refer to *Appendix E, The GRUB Boot Loader*.

F.2.3. The Kernel

When the kernel is loaded, it immediately initializes and configures the computer's memory and configures the various hardware attached to the system, including all processors, I/O subsystems, and storage devices. It then looks for the compressed **initramfs** image(s) in a predetermined location in memory, decompresses it directly to **/sysroot/**, and loads all necessary drivers. Next, it initializes virtual devices related to the file system, such as LVM or software RAID, before completing the **initramfs** processes and freeing up all the memory the disk image once occupied.

The kernel then creates a root device, mounts the root partition read-only, and frees any unused memory.

At this point, the kernel is loaded into memory and operational. However, since there are no user applications that allow meaningful input to the system, not much can be done with the system.

To set up the user environment, the kernel executes the **/sbin/init** program.

F.2.4. The **/sbin/init** Program

The **/sbin/init** program (also called **init**) coordinates the rest of the boot process and configures the environment for the user.

When the **init** command starts, it becomes the parent or grandparent of all of the processes that start up automatically on the system. First, it runs the **/etc/rc.d/rc.sysinit** script, which sets the environment path, starts swap, checks the file systems, and executes all other steps required for system initialization. For example, most systems use a clock, so **rc.sysinit** reads the **/etc/sysconfig/clock** configuration file to initialize the hardware clock. Another example is if there are special serial port processes which must be initialized, **rc.sysinit** executes the **/etc/rc.serial** file.

The **init** command then runs the **/etc/inittab** script, which describes how the system should be set up in each *SysV init runlevel*. Runlevels are a state, or *mode*, defined by the services listed in the SysV **/etc/rc.d/rc<x>.d/** directory, where *<x>* is the number of the runlevel. For more information on SysV init runlevels, refer to *Section F.4, "SysV Init Runlevels"*.

Next, the **init** command sets the source function library, **/etc/rc.d/init.d/functions**, for the system, which configures how to start, kill, and determine the PID of a program.

The **init** program starts all of the background processes by looking in the appropriate **rc** directory for the runlevel specified as the default in **/etc/inittab**. The **rc** directories are numbered to correspond to the runlevel they represent. For instance, **/etc/rc.d/rc5.d/** is the directory for runlevel 5.

When booting to runlevel 5, the **init** program looks in the **/etc/rc.d/rc5.d/** directory to determine which processes to start and stop.

Below is an example listing of the **/etc/rc.d/rc5.d/** directory:

```
K05innd -> ../init.d/innd
```

```
K05saslauthd -> ../init.d/saslauthd
K10dc_server -> ../init.d/dc_server
K10psacct -> ../init.d/psacct
K10radiusd -> ../init.d/radiusd
K12dc_client -> ../init.d/dc_client
K12FreeWnn -> ../init.d/FreeWnn
K12mailman -> ../init.d/mailman
K12mysqld -> ../init.d/mysqld
K15httpd -> ../init.d/httpd
K20netdump-server -> ../init.d/netdump-server
K20rstatd -> ../init.d/rstatd
K20rusersd -> ../init.d/rusersd
K20rwhod -> ../init.d/rwhod
K24irda -> ../init.d/irda
K25squid -> ../init.d/squid
K28amd -> ../init.d/amd
K30spamassassin -> ../init.d/spamassassin
K34dhcrelay -> ../init.d/dhcrelay
K34yppasswdd -> ../init.d/yppasswdd
K35dhcpd -> ../init.d/dhcpd
K35smb -> ../init.d/smb
K35vncserver -> ../init.d/vncserver
K36lisa -> ../init.d/lisa
K45arpwatch -> ../init.d/arpwatch
K45named -> ../init.d/named
K46radvd -> ../init.d/radvd
K50netdump -> ../init.d/netdump
K50snmpd -> ../init.d/snmpd
K50snmptrapd -> ../init.d/snmptrapd
K50tux -> ../init.d/tux
K50vsftpd -> ../init.d/vsftpd
K54dovecot -> ../init.d/dovecot
K61ldap -> ../init.d/ldap
K65kadmin -> ../init.d/kadmin
K65kprop -> ../init.d/kprop
K65krb524 -> ../init.d/krb524
K65krb5kdc -> ../init.d/krb5kdc
K70aep1000 -> ../init.d/aep1000
K70bcm5820 -> ../init.d/bcm5820
K74ypserv -> ../init.d/ypserv
K74ypxfrd -> ../init.d/ypxfrd
K85mdmpd -> ../init.d/mdmpd
K89netplugd -> ../init.d/netplugd
K99microcode_ctl -> ../init.d/microcode_ctl
S04readahead_early -> ../init.d/readahead_early
S05kudzu -> ../init.d/kudzu
S06cpuspeed -> ../init.d/cpuspeed
S08ip6tables -> ../init.d/ip6tables
S08iptables -> ../init.d/iptables
S09isdn -> ../init.d/isdn
S10network -> ../init.d/network
```

```
S12syslog -> ../init.d/syslog
S13irqbalance -> ../init.d/irqbalance
S13portmap -> ../init.d/portmap
S15mdmonitor -> ../init.d/mdmonitor
S15zebra -> ../init.d/zebra
S16bgpd -> ../init.d/bgpd
S16ospf6d -> ../init.d/ospf6d
S16ospfd -> ../init.d/ospfd
S16ripd -> ../init.d/ripd
S16ripngd -> ../init.d/ripngd
S20random -> ../init.d/random
S24pcmcia -> ../init.d/pcmcia
S25netfs -> ../init.d/netfs
S26apmd -> ../init.d/apmd
S27ypbind -> ../init.d/ypbind
S28autofs -> ../init.d/autofs
S40smartd -> ../init.d/smartd
S44acpid -> ../init.d/acpid
S54hpoj -> ../init.d/hpoj
S55cups -> ../init.d/cups
S55sshd -> ../init.d/sshd
S56rawdevices -> ../init.d/rawdevices
S56xinetd -> ../init.d/xinetd
S58ntpd -> ../init.d/ntpd
S75postgresql -> ../init.d/postgresql
S80sendmail -> ../init.d/sendmail
S85gpm -> ../init.d/gpm
S87iiim -> ../init.d/iiim
S90canna -> ../init.d/canna
S90crond -> ../init.d/crond
S90xfs -> ../init.d/xfs
S95atd -> ../init.d/atd
S96readahead -> ../init.d/readahead
S97messagebus -> ../init.d/messagebus
S97rhnsd -> ../init.d/rhnsd
S99local -> ../rc.local
```

As illustrated in this listing, none of the scripts that actually start and stop the services are located in the **/etc/rc.d/rc5.d/** directory. Rather, all of the files in **/etc/rc.d/rc5.d/** are *symbolic links* pointing to scripts located in the **/etc/rc.d/init.d/** directory. Symbolic links are used in each of the **rc** directories so that the runlevels can be reconfigured by creating, modifying, and deleting the symbolic links without affecting the actual scripts they reference.

The name of each symbolic link begins with either a K or an S. The K links are processes that are killed on that runlevel, while those beginning with an S are started.

The **init** command first stops all of the K symbolic links in the directory by issuing the **/etc/rc.d/init.d/<command> stop** command, where *<command>* is the process to be killed. It then starts all of the S symbolic links by issuing **/etc/rc.d/init.d/<command> start**.

> **Note**
>
> After the system is finished booting, it is possible to log in as root and execute these same scripts to start and stop services. For instance, the command `/etc/rc.d/init.d/httpd stop` stops the Apache HTTP Server.

Each of the symbolic links are numbered to dictate start order. The order in which the services are started or stopped can be altered by changing this number. The lower the number, the earlier it is started. Symbolic links with the same number are started alphabetically.

> **Note**
>
> One of the last things the `init` program executes is the `/etc/rc.d/rc.local` file. This file is useful for system customization. Refer to *Section F.3, "Running Additional Programs at Boot Time"* for more information about using the `rc.local` file.

After the `init` command has progressed through the appropriate `rc` directory for the runlevel, the `/etc/inittab` script forks an `/sbin/mingetty` process for each virtual console (login prompt) allocated to the runlevel. Runlevels 2 through 5 have all six virtual consoles, while runlevel 1 (single user mode) has one, and runlevels 0 and 6 have none. The `/sbin/mingetty` process opens communication pathways to *tty* devices[2], sets their modes, prints the login prompt, accepts the user's username and password, and initiates the login process.

In runlevel 5, the `/etc/inittab` runs a script called `/etc/X11/prefdm`. The `prefdm` script executes the preferred X display manager[3] — `gdm`, `kdm`, or `xdm`, depending on the contents of the `/etc/sysconfig/desktop` file.

Once finished, the system operates on runlevel 5 and displays a login screen.

F.3. Running Additional Programs at Boot Time

The `/etc/rc.d/rc.local` script is executed by the `init` command at boot time or when changing runlevels. Adding commands to the bottom of this script is an easy way to perform necessary tasks like starting special services or initialize devices without writing complex initialization scripts in the `/etc/rc.d/init.d/` directory and creating symbolic links.

The `/etc/rc.serial` script is used if serial ports must be setup at boot time. This script runs `setserial` commands to configure the system's serial ports. Refer to the `setserial` man page for more information.

F.4. SysV Init Runlevels

The SysV init runlevel system provides a standard process for controlling which programs `init` launches or halts when initializing a runlevel. SysV init was chosen because it is easier to use and more flexible than the traditional BSD-style init process.

The configuration files for SysV init are located in the `/etc/rc.d/` directory. Within this directory, are the `rc`, `rc.local`, `rc.sysinit`, and, optionally, the `rc.serial` scripts as well as the following directories:

---

[2] Refer to the Red Hat Enterprise Linux Deployment Guide for more information about `tty` devices.
[3] Refer to the Red Hat Enterprise Linux Deployment Guide for more information about display managers.

```
init.d/ rc0.d/ rc1.d/ rc2.d/ rc3.d/ rc4.d/ rc5.d/ rc6.d/
```

The **init.d/** directory contains the scripts used by the **/sbin/init** command when controlling services. Each of the numbered directories represent the six runlevels configured by default under Fedora.

F.4.1. Runlevels

The idea behind SysV init runlevels revolves around the idea that different systems can be used in different ways. For example, a server runs more efficiently without the drag on system resources created by the X Window System. Or there may be times when a system administrator may need to operate the system at a lower runlevel to perform diagnostic tasks, like fixing disk corruption in runlevel 1.

The characteristics of a given runlevel determine which services are halted and started by **init**. For instance, runlevel 1 (single user mode) halts any network services, while runlevel 3 starts these services. By assigning specific services to be halted or started on a given runlevel, **init** can quickly change the mode of the machine without the user manually stopping and starting services.

The following runlevels are defined by default under Fedora:

- **0** — Halt

- **1** — Single-user text mode

- **2** — Not used (user-definable)

- **3** — Full multi-user text mode

- **4** — Not used (user-definable)

- **5** — Full multi-user graphical mode (with an X-based login screen)

- **6** — Reboot

In general, users operate Fedora at runlevel 3 or runlevel 5 — both full multi-user modes. Users sometimes customize runlevels 2 and 4 to meet specific needs, since they are not used.

The default runlevel for the system is listed in **/etc/inittab**. To find out the default runlevel for a system, look for the line similar to the following near the top of **/etc/inittab**:

```
id:5:initdefault:
```

The default runlevel listed in this example is five, as the number after the first colon indicates. To change it, edit **/etc/inittab** as root.

> ⚠ **Warning**
>
> Be very careful when editing **/etc/inittab**. Simple typos can cause the system to become unbootable. If this happens, either use a boot diskette, enter single-user mode, or enter rescue mode to boot the computer and repair the file.

> For more information on single-user and rescue mode, refer to the chapter titled *Basic System Recovery* in the *Red Hat Enterprise Linux Deployment Guide*.

It is possible to change the default runlevel at boot time by modifying the arguments passed by the boot loader to the kernel. For information on changing the runlevel at boot time, refer to *Section E.7, "Changing Runlevels at Boot Time"*.

F.4.2. Runlevel Utilities

One of the best ways to configure runlevels is to use an *initscript utility*. These tools are designed to simplify the task of maintaining files in the SysV init directory hierarchy and relieves system administrators from having to directly manipulate the numerous symbolic links in the subdirectories of `/etc/rc.d/`.

Fedora provides three such utilities:

- `/sbin/chkconfig` — The `/sbin/chkconfig` utility is a simple command line tool for maintaining the `/etc/rc.d/init.d/` directory hierarchy.

- **/usr/sbin/ntsysv** — The ncurses-based **/sbin/ntsysv** utility provides an interactive text-based interface, which some find easier to use than `chkconfig`.

- **Services Configuration Tool** — The graphical **Services Configuration Tool** (`system-config-services`) program is a flexible utility for configuring runlevels.

Refer to the chapter titled *Controlling Access to Services* in the *Red Hat Enterprise Linux Deployment Guide* for more information regarding these tools.

F.5. Shutting Down

To shut down Fedora, the root user may issue the `/sbin/shutdown` command. The `shutdown` man page has a complete list of options, but the two most common uses are:

```
/sbin/shutdown -h now /sbin/shutdown -r now
```

After shutting everything down, the `-h` option halts the machine, and the `-r` option reboots.

PAM console users can use the `reboot` and `halt` commands to shut down the system while in runlevels 1 through 5. For more information about PAM console users, refer to the Red Hat Enterprise Linux Deployment Guide.

If the computer does not power itself down, be careful not to turn off the computer until a message appears indicating that the system is halted.

Failure to wait for this message can mean that not all the hard drive partitions are unmounted, which can lead to file system corruption.

Appendix G. Other Technical Documentation

This document provides a reference for using the Fedora installation software, known as **anaconda**. To learn more about
anaconda, visit the project Web page: *http://www.fedoraproject.org/wiki/Anaconda*.

Both **anaconda** and Fedora systems use a common set of software components. For detailed information on key technologies, refer to the Web sites listed below:

Boot Loader

> Fedora uses the
> **GRUB** boot loader. Refer to *http://www.gnu.org/software/grub/* for more information.

Disk Partitioning

> Fedora uses **parted** to partition disks. Refer to *http://www.gnu.org/software/parted/* for more information.

Storage Management

> Logical Volume Management (LVM) provides administrators with a range of facilities to manage storage. By default, the Fedora installation process formats drives as LVM volumes. Refer to *http://www.tldp.org/HOWTO/LVM-HOWTO/* for more information.

Audio Support

> The Linux kernel used by Fedora incorporates
> PulseAudio audio server. For more information about PulseAudio, refer to the project documentation: *http://www.pulseaudio.org/wiki/Documentation*.

Graphics System

> Both the installation system and Fedora use the
> **Xorg** suite to provide graphical capabilities. Components of **Xorg** manage the display, keyboard and mouse for the desktop environments that users interact with. Refer to *http://www.x.org/* for more information.

Remote Displays

> Fedora and **anaconda** include
> VNC (Virtual Network Computing) software to enable remote access to graphical displays.
> For more information about VNC, refer to the documentation on the RealVNC Web site: *http://www.realvnc.com/documentation.html*.

Command-line Interface

> By default, Fedora uses the GNU **bash** shell to provide a command-line interface. The GNU Core Utilities complete the command-line environment. Refer to *http://www.gnu.org/software/bash/bash.html* for more information on **bash**. To learn more about the GNU Core Utilities, refer to *http://www.gnu.org/software/coreutils/*.

Remote System Access

> Fedora incorporates the
> OpenSSH suite to provide remote access to the system. The SSH

service enables a number of functions, which include access to the command-line from other systems, remote command execution, and network file transfers. During the installation process **anaconda** may use the **scp** feature of OpenSSH to transfer crash reports to remote systems. Refer to the OpenSSH Web site for more information: *http://www.openssh.com/*.

Access Control

SELinux provides Mandatory Access Control (MAC) capabilities that supplement the standard Linux security features. Refer to the SELinux Project Pages for more information: *http://docs.fedoraproject.org/selinux-guide*.

Firewall

The Linux kernel used by Fedora incorporates the **netfilter** framework to provide firewall features. The Netfilter project website provides documentation for both **netfilter**, and the **iptables** administration facilities: *http://netfilter.org/documentation/index.html*.

Software Installation

Fedora uses **yum** to manage the RPM packages that make up the system. Refer to *http://docs.fedoraproject.org/yum/* for more information.

Virtualization

Virtualization provides the capability to simultaneously run multiple operating systems on the same computer. Fedora also includes tools to install and manage the secondary systems on a Fedora host. You may select virtualization support during the installation process, or at any time thereafter. Refer to *http://www.fedoraproject.org/wiki/Tools/Virtualization* for more information.

Appendix H. Contributors and production methods

H.1. Contributors

- *Fabian Affolter*[1] (translator - German)

- *Amanpreet Singh Alam*[2] (translator - Punjabi)

- *Domingo Becker*[3] (translator - Spanish)

- Micha# Bentkowski (translator - Polish)

- *Teta Bilianou*[4] (translator - Greek)

- Lucas Brausch (translator - German)

- Guido Caruso (translator - Italian)

- Guillaume Chardin (translator - French)

- *Nikos Charonitakis*[5] (translator - Greek)

- Fabien Decroux (translator - French)

- *Hans De Goede*[6] (writer)

- *Piotr Dr#g*[7] (translator - Polish)

- *Damien Durand*[8] (translator - French)

- *Stuart Ellis*[9] (writer, editor)

- *Paul W. Frields*[10] (writer, editor)

- *Dimitris Glezos*[11] (translator - Greek)

- *Guillermo Gómez*[12] (translator - Spanish)

- *Rui Gouveia*[13] (translator - Portuguese)

- Severin Heiniger (translator - German)

- Ryuichi Hyugabaru (translator - Japanese)

- Chris Johnson (writer)

- *Miloš Komar#evi#*[14] (translator - Serbian)

- *Rüdiger Landmann*[15] (writer, editor)

- *Magnus Larsson*[16] (translator - Swedish)

- *Christopherus Laurentius*[17] (translator - Indonesian)

- Erick Lemon (writer)

- Wei Liu (translator - Chinese)

- Gan Lu (translator - Chinese)

- *Igor Mileti#*[18] (translator - Serbian)

- Jeremy W. Mooney (writer)

- Enik# Nagy (translator - Hungarian)

- Igor Nestorovi# (translator - Serbian)

- *David Nalley*[19] (writer, editor, document lead)

- John Nguyen (writer)

- Janis Ozolins (translator - Latvian)

- *Davidson Paulo*[20] (translator - Brazilian Portuguese)

- *Silvio Pierro*[21] (translator - Italian)

- Micha Pietsch (translator - German)

- José Nuno Pires (translator - Portuguese)

- Piotr Podgórski (translator - Polish)

- *Tommy Reynolds* [22] (writer)

- Tim Richert (translator - German)

- *Dominik Sandjaja*[23] (translator - German)

- *Sharuzzaman Ahmat Raslan*[24] (translator - Malay)

- Keld Simonsen (translator - Danish)

- *Jared K. Smith*[25] (writer, editor)

- Espen Stefansen (translator - Norwegian Bokmål)

- *Francesco Tombolini*[26] (translator - Italian)

- *Dimitris Typaldos*[27] (translator - Greek)

- Göran Uddeborg (translator - Swedish)

- *Michaël Ughetto*[28] (translator - French)

- *Karsten Wade*[29] (writer, editor, publisher)

- *Geert Warrink*[30] (translator - Dutch)

- Elizabeth Ann West (editor)

- Xiaofan Yang (translator - Chinese)

- *Yuan Yijun*[31] (translator - Chinese)

- *Diego Búrigo Zacarão*[32] (translator - Brazilian Portuguese)

http://docs.fedoraproject.org/install-guide/

H.2. Production methods

Writers produce the Install Guide directly in DocBook XML in a revision control repository. They collaborate with other subject matter experts during the beta release phase of Fedora to explain the installation process. The editorial team ensures consistency and quality of the finished guide. At this point, the team of translators produces other language versions of the release notes, and then they become available to the general public as part of Fedora. The publication team also makes the guide, and subsequent errata, available via the Web.

Appendix I. Revision History

Revision History

Revision 10.0.1	Mon Feb 16 2009	RuedigerLandmann*r.landmann@redhat.co*
Convert to build in Publican		
Revision 10.0.0	Mon Nov 24 2008	KarstenWade*kwade@redhat.com*
Build and publish Fedora 10 version		
Revision 9.9.2	Sat Oct 18 2008	JaredK.Smith*jaredsmith@jaredsmith.ne*
Prepare for release of Fedora 10		
Revision 9.0.2	Fri Jul 25 2008	PaulW.Frields*stickster@gmail.com*
Fix incorrect livecd-tools instructions		
Revision 9.0.1	Sat Jun 28 2008	PaulW.Frields*stickster@gmail.com*
Lots of bug fixes		
Revision 9.0.0	Tue May 13 2008	PaulW.Frields*stickster@gmail.com*
Add information on upgrading a distribution		

Index

Symbols

/boot/ partition
    recommended partitioning, 63
/root/install.log
    install log file location, 80
/var/ partition
    recommended partitioning, 63

A

adding partitions, 66
    file system type, 68
Anaconda, 243
architecture, 4
    determining, 4
array (see RAID)
ATAPI CD-ROM
    unrecognized, problems with, 35
automatic partitioning, 51, 51, 54

B

Basic Input/Output System (see BIOS)
BIOS
    definition of, 235
        (see also boot process)
BIOS (Basic Input/Output System), 7
BitTorrent, 4
    seeding, 4
boot CD-ROM, 12
    creating, 12
boot loader, 45, 69
    (see also GRUB)
    alternatives to, 73
        commercial products, 74
        LOADLIN, 73
        SYSLINUX, 74
    configuration, 69
    GRUB, 69
    installing on boot partition, 72
    MBR, 72
    password, 71
    upgrading, 45
boot loader password, 71
boot loaders, 225
    (see also GRUB)
    definition of, 225
boot methods
    boot CD-ROM, 12
    USB pen drive, 12

boot options, 32
    boot.iso, 33
    linux mediacheck, 15
    mediacheck, 32
    serial mode, 33
        UTF-8, 33
    text mode, 32
boot process, 235, 235
    (see also boot loaders)
    chain loading, 225
    direct loading, 225
    for x86, 235
    stages of, 235, 235
        /sbin/init command, 237
        BIOS, 235
        boot loader, 236
        EFI shell, 235
        kernel, 237
boot.iso, 33
booting
    emergency mode, 185
    installation program
        x86, AMD64 and Intel 64, 31
    rescue mode, 182
    single-user mode, 184

C

canceling the installation, 35
CD-ROM
    ATAPI, 34
        unrecognized, problems with, 35
    boot CD-ROM, creating, 12
    IDE, 34
        unrecognized, problems with, 35
    installation from, 34
    SCSI, 35
CD/DVD media
    booting, 7
    downloading, 3
        (see also ISO images)
chkconfig, 242
    (see also services)
clock, 48
configuration
    clock, 48, 48
    GRUB, 69
    hardware, 17
    time, 48
    time zone, 48
consoles, virtual, 28

D

DHCP
    diskless environment, 112
    PXE installations, 112
DHCP (Dynamic Host Configuration Protocol), 45
Disk Druid
    adding partitions
        file system type, 68
    buttons, 58
    deleting partitions, 68
    editing partitions, 68
    partitions, 56
Disk Partitioner
    adding partitions, 66
disk partitioning, 51
disk space, 11
diskette
    media, 19
diskless environment
    DHCP configuration, 112
dmraid
    installation, 53
domain name, 46
driver diskette, 31
driver disks, 98
driver media, 19
    creating diskette from image, 20
    produced by others, 20
    produced by Red Hat, 19
    using a driver image, 20

E

EFI shell
    definition of, 235
        (see also boot process)
emergency mode, 185
ext2 (see file systems)
ext3 (see file systems)
extended partitions, 207
Extensible Firmware Interface shell (see EFI shell)

F

feedback
    contact information for this manual, xiv
file system
    formats, overview of, 203
file system types, 68
file systems
    ext2, 35
    ext3, 35

vfat, 35
firewall
    documentation, 244
Firstboot, 171
FTP
    installation, 13

G

GRUB, 69, 236
    (see also boot loaders)
    additional resources, 234
        installed documentation, 234
        related books, 234
        useful websites, 234
    alternatives to, 73
        commercial products, 74
        LOADLIN, 73
        SYSLINUX, 74
    boot process, 225
    Changing Runlevels at Boot Time, 234
    changing runlevels with, 229
    commands, 230
    configuration, 69
    configuration file
        /boot/grub/grub.conf, 232
        structure, 232
    definition of, 225
    documentation, 243
    features, 226
    installing, 226
    interfaces, 229
        command line, 229
        menu, 229
        menu entry editor, 229
        order of, 230
    menu configuration file, 232
        directives, 232
    role in boot process, 236
    SMP motherboards, 74
    terminology, 227
        devices, 227
        files, 228
        root file system, 228
grub.conf, 232
    (see also GRUB)

H

halt, 242
    (see also shutdown)
Hard disk
    initializing, 42