Hentzenwerke Whitepaper Series

Updating Fedora Core

By Whil Hentzen

Unless you're installing a brand-new version of a piece of software, there are bound to be updates available for it, and Fedora Core is no exception. Updating your system with the latest patches, bug fixes, and security updates will keep your system running as smoothly and problem-free as possible. Because Fedora Core is Linux, there's not just one way to update your system. In this whitepaper, I'll explain how the Fedora Core basic update process works and how to use the various mechanisms available to you.

Hentzenwerke Publishing, Inc. • books@hentzenwerke.com • www.hentzenwerke.com

1. Preface

1.1 Copyright

1.2 Revisions

1.2.1 History

Version	Date	Synopsis	Author
1.0.0	2004/3/3	Original	WH

1.2.2 New version

The newest version of this document will be found at www.hentzenwerke.com.

1.2.3 Feedback and corrections

If you have questions, comments, or corrections about this document, please feel free to email me at 'books@hentzenwerke.com'. I also welcome suggestions for passages you find unclear.

1.3 References and acknowledgments

N/A.

1.4 Disclaimer

No warranty! This material is provided as is, with no warranty of fitness for any particular purpose. Use the concepts, examples and other content at your own risk. There may be errors and inaccuracies that in some configurations may be damaging to your system. The author(s) disavows all liability for the contents of this document.

Before making any changes to your system, ensure that you have backups and other resources to restore the system to its state before making those changes.

All copyrights are held by their respective owners, unless specifically noted otherwise. Use of a term in this document should not be regarded as affecting the validity of any trademark or service mark. Naming of particular products or brands should not be seen as endorsements.

1.5 Prerequisites

This document was written using Fedora Core 1.0 and assumes a beginner's familiarity with use of Linux via the GUI and the Command Window.

2. Introduction

Updating a piece of software has become a nearly trivial process. In the olden days, many software manufacturers didn't automatically issue patches and updates. Instead, they rolled fixes into an update or upgrade that may or may not have been free. You had to wait for patch disks to show up in the mail, or, worse, had to request the disks after a chance discovery of their availability.

Once bulletin boards became more widespread, and communication software and hardware advanced sufficiently, some manufacturers posted patch files on their boards, or on third-party boards like CompuServe, and posted announcements about their availability on various electronic forums.

After Web sites made their debut in the early '90s, companies began migrating their patches and updates to a company support site, and started using e-mail to notify users of availability as well.

Still, it was up to the user to be proactive enough to get and install the updates. This wasn't always easy, and a combination of lack of sophistication on the part of some users as well as less-than-coherent directions from the software company made this all too often a risky procedure.

Nowadays, it's become common for software to "phone home" to the manufacturer in order to check on whether or not updates are available, and, if so, alert the user. Some software can be configured to look for and install updates automatically, without any user intervention. Whether or not you take advantage of this capability is up to you. Many users of Windows systems configure their anti-virus products to go through automatic updates, but don't let the operating system get patched automatically, due to the irregular quality of service packs and other patches.

Fedora Core does the first part—checking to see if updates are available and notifying the user—via the Red Hat Network Alert Notification Tool, which appears as an icon in the panel, as shown in **Figure 1**.

***Figure 1**. Upon startup, the Red Hat Network Alert Notification Tool will display a blue check mark that indicates it is not aware of any new updates.*

When the machine can connect to the Internet, it will attempt to query a Web site for information about available updates. When the icon is green, it means that FC is currently querying for updates, as shown in **Figure 2**.

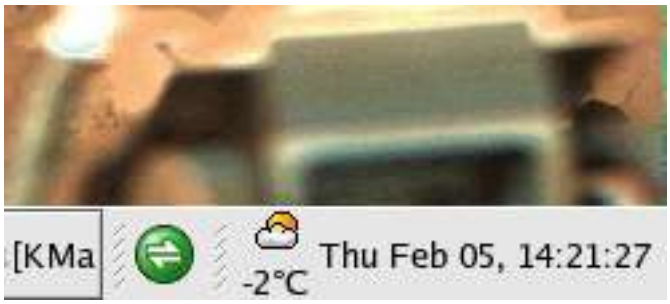

***Figure 2**. The Red Hat Network Alert Notification Tool icon's green color means updates are currently being queried.*

If there are updates available that have not been applied, the icon turns red, as shown in **Figure 3**.

***Figure 3**. The round red icon in the lower right corner of the panel indicates that there are updates available.*

After all available updates have been applied, the icon turns blue again, meaning that the system is up to date, as shown in **Figure 4**.

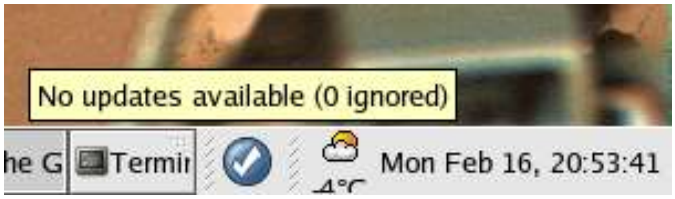

***Figure 4**. The blue check mark indicates that no updates are available.*

If you move your mouse over the icon, a tooltip will display how many updates are available and how many have been ignored during the last update session. If the tooltip in Figure 3 alarms you (nearly 100 updates available), remember that there could be thousands of packages installed on an FC system. Furthermore, the more time that has passed since the time when the ISOs you used to install were made available and the last update session, the more updates will be available. If you install FC (or any distro, for that matter) with old files, there will likely be a lot of updates to install the first time you do an update, but significantly fewer each successive update.

3. The big picture

Fedora Core provides three mechanisms to update your system. Before I get into the specific mechanisms, a word about three items: dependencies, RPMs, and the Red Hat Network.

3.1 Dependencies

No man is an island, nor are too many pieces of software. Many applications aren't monolithic files, but instead rely on other files or applications. For example, in order to run application A, application B must be installed as well. This requirement is called a "dependency" (in that A is dependent on B).

The core Linux operating system, as with all operating systems, consists of hundreds and hundreds of files—and if you include the applications that come with most distros, you're looking at thousands of files. Many of them have dependencies—including multiple dependencies. For example, A depends on B and C. B depends on D, which depends on E, and C depends on F. In order to install A, then, you have to install B, C, D, E, and F. Given that other programs also have intertwined dependencies, you can see how the whole situation can get confusing quickly.

In the olden days, you had to resolve these dependencies ##not clear what "that" is – install dependencies? ##by hand. Not appealing. What people would do, then, is write scripts (think "batch files" or "macros") that would automate these processes. Much better. But still a nuisance.

So Red Hat created the Red Hat Package Manager, intended to provide a clean front end to the scripts that automated the dependency problem.

3.2 RPMs

RPM is short for "Red Hat Package Manager." It's a tool that automates much of the process of installing and uninstalling parts of an FC system. The RPM tool has several pieces.

The first is a database on your own system that contains information about packages and the files required by each package. When you install or uninstall packages, RPM tracks this information in the database. Suppose you install a package that uses File A. Later, you install a second package that also uses File A. Because File A was installed with the first package, it's not installed again. If you later uninstall the first package, File A is not removed, because it's still needed by the second package. All of this is handled by RPM.

The second piece of the RPM tool is software that provides a variety of functions that work with this database, such as installation, uninstallation, and information about in the contents of a package, what is currently installed, and so on.

The third piece is the RPM package itself. Every RPM has two files, a header file and the package file. The header contains a list of all files, descriptions of the packages, requirement and conflict lists, and some other descriptive information. Different update mechanisms use the RPM database on your system and the RPM headers in different ways.

Fedora Core is not the only Linux distribution using the RPM system. Obviously, Red Hat's corporate offerings use it, but so do SuSE and others.

Installing a RPM package is as simple as issuing this command:

```
rpm -i abc.i386.rpm
```

If the installation succeeds, no output will be returned. You can use the -ivh parameters to provide verbose feedback on the installation.

If the installation fails, say, because the package has unresolved dependencies, those requirements will be provided in the output returned from the command. The rpm command is actually a rather complex beast; indeed, entire books have been written about RPM, and just the help included with the command to list the various options is well over 100 lines long! It is possible to munge your system beyond recognition with the inappropriate use of rpm, so I suggest you stick with the up2date mechanism described in this chapter until you're considerably more comfortable.

3.3 Red Hat Network

Red Hat Corporation used to provide a version of Linux called Red Hat Linux that you could purchase in a box set (either directly from Red Hat, from an online distributor, or from a physical store) or download for free from the Red Hat Web site.

Part of the Red Hat Linux package was a service called the Red Hat Network, which was available as a demo, free for a couple of months. You could then extend your free subscription by filling out a survey every few months. An alternative was to buy a subscription to the Red Hat Network. Such subscriptions started at $60 per year.

Red Hat Linux could be configured to connect to the Red Hat Network, download available updates, and install them—all automatically—without any work on the part of the user. The Red Hat Network used a software tool called up2date that I'll discuss shortly.

With the introduction of Fedora Core, access to the Red Hat Network is restricted to people who purchase Red Hat's enterprise products. Fedora Core can be configured to check for available updates, but the automated update mechanisms of Red Hat Network (whereby the updates are automatically downloaded and installed) are not available to Fedora Core users. More on this in a minute.

With this background in mind, let's look at the specific mechanisms available for Fedora Core users.

3.4 up2date

The first update mechanism is the Red Hat Update Agent, referred to as "up2date" in shorthand. up2date is also used by the Red Hat Network for its enterprise products. The difference between Red Hat Network's version of up2date and Fedora Core's implementation is the work that the user has to perform.

Fedora Core still has the Red Hat Network Alert Notification Tool (as shown earlier in Figure 1) that appears red when updates are available. However, in order to use up2date, you'll run the up2date tool to fetch the headers for the available updates. These headers will be displayed in a list, from which you can then pick and choose, depending on which packages you want to install.

After selecting the packages of interest, you'll fetch the actual updates from the Red Hat Web site and download them to your computer. Once downloaded, up2date will verify that the downloaded files are good, and then attempt to install them. You can also manually install packages using the rpm command as mentioned earlier.

The default configuration of up2date suffers from some performance and reliability issues. It attempts to connect to just one server by default, which means that just about every FC user on the planet is trying to get updates from the same server. That server can get overloaded rather easily with all of this traffic, and as a result connections can be very slow, and they often time out, even if you've got a fast connection on your end. The error messages and resulting failures are confusing and frustrating. Dealing with these problems is one of the most popular subjects on the FC mailing lists.

One thing I'll explain is how to change up2date's configuration so that it looks at a different server to fetch updates, resulting in improved performance and fewer confusing error messages.

3.5 yum

Similar to up2date, yum (Yellow Dog Updater, Modified) is an automatic updating tool that runs on the system being updated. Like up2date, it determines dependencies and what needs to be installed or uninstalled. From the point of view of an end user, it is a piece of software installed on one's system. (There is a corresponding piece of software on the server that you only need to be concerned with if you are creating a repository for yum updates.)

yum can be configured to work with any old FTP or HTTP server, which makes it ideal for sites that create custom configurations or that require multiple repositories of RPMs. For example, if you're supporting multiple groups of users, each of whom wants their own custom configuration of packages, you can create separate repositories for each group. Then they update from their own repository, and only those updates applicable to their configuration are available.

Unlike other tools, yum copies the header from the RPMs on the server. The yum client (on the system being updated) then uses those headers to determine what needs to be installed or uninstalled, instead of using a custom library or index of updating information. After determining what will be done, yum relies on RPM to do the actual work.

yum is highly configurable, and is ideal for the scenario mentioned earlier, in which an administrator is supporting multiple groups of users who each are responsible for their own network's administration. I can't even begin to do it justice here, so if this is something that sounds interesting, I'll refer you to the definitive HOWTO here (note that the repeated directory at the end is not a typo):

```
http://www.phy.duke.edu/~rgb/General/yum_HOWTO/yum_HOWTO/
```

3.6 APT

APT (Advanced Packaging Tool) is a third mechanism for updating Fedora Core. It, too, is an automatic updating tool that determines dependencies and handles installation and uninstallation, and was originally created for the Debian distribution. I mention it here because as you become comfortable with updating, you will undoubtedly run into someone who mentions APT. While it is command-based ("apt-get <package name>"), APT users have the option of using a GUI front end called Synaptic.

Like yum, a complete description of how to use APT is beyond the scope of this chapter. If you're interested, you can find the definitive HOWTO here:

http://www.debian.org/doc/manuals/apt-howto/index.en.html

3.7 Summary

Which to use? Like many things in the computer world, there isn't a clear and simple answer—it's in large part a matter of personal preference and philosophy, and thus advocating one over another is prone to spark another so-called "religious war." Because you're just getting started with Linux, I'll discuss how to use the GUI for up2date in detail in this chapter. After you've gotten comfortable with the process and know your way around the block, you can check out alternatives like yum and APT.

4. The details—step by step

In this section, I'll discuss how to use the up2date mechanism in great detail, because it's the mechanism most applicable to a new Fedora Core user.

4.1 Are there updates available?

When you right-click the Red Hat Network Alert Notification Tool icon, you'll get a menu that offers a variety of update options, as shown in **Figure 5**.

***Figure 5**. Right-clicking the Red Hat Network Alert Notification Tool icon displays a number of update options.*

Because Fedora Core 1 is a transitional release—from Red Hat Linux to a distribution completely supported by the community—some of the items in the commercial version are still hanging around. The RHN Website option available in the Red Hat Network Alert Notification Tool is one example of commercial remnants. Clicking on it still takes you to the RHN Web site, but the information there doesn't have anything to do with Fedora Core updates.

In order to run up2date properly, you'll want to follow three steps. First, you'll configure up2date with the Red Hat Network Alert Notification tool. ##Whil, I see from the wizard that the title of the tool includes the word "Alert." It makes me wonder if the preceding discussion of the "Notification Tool" (and "Notification Tool icon") should also include the word "Alert." So I marked it a bunch of times. I think I caught them allJF## Second, you'll point up2date to a mirror different than the default that comes with Fedora Core. Third, you'll launch up2date and run the wizard. After you've done this process once, future up2date episodes will simply involve the third step of checking for (and installing, if any) updates.

4.2 Configuration

This menu option allows you to configure the RHN Alert Notification Tool, including proxy settings. Click the Configuration menu option to open the Welcome screen as shown in **Figure 6**.

***Figure 6**. The Welcome screen of the Red Hat Network Alert Notification Tool.*

Click Forward to move to the Terms of Service screen, as shown in **Figure 7**.

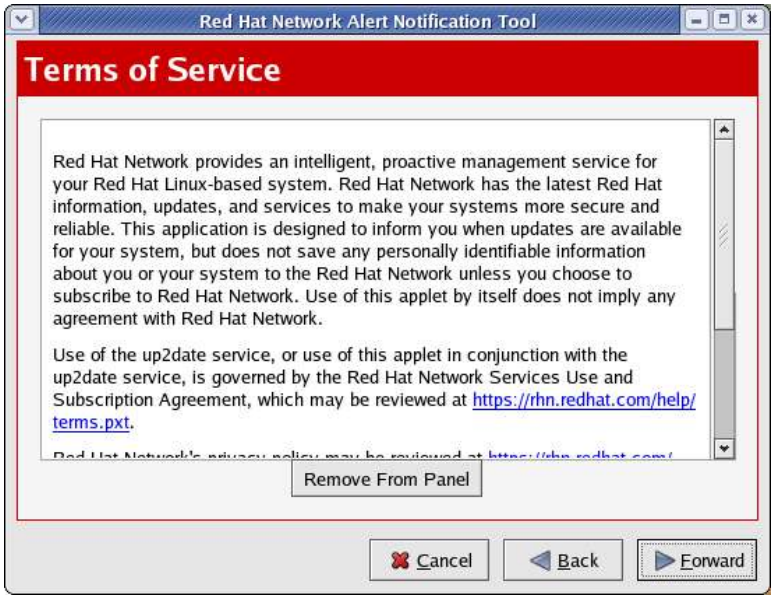

***Figure 7**. Reading the Terms of Service screen is a good idea, for a change.*

Unlike most licenses, this one contains some useful information. You should actually read through the Terms of Service paragraphs. You can also remove the Red Hat Network AlertNotification Tool from the panel by clicking the Remove From Panel button under the edit box. (If you do so, and later want to add the ##Alert??##Notification Tool back, click System Tools | Red Hat Network Alert Icon.) Then click the Forward button to open the Proxy Configuration screen, as shown in**Figure 8**.

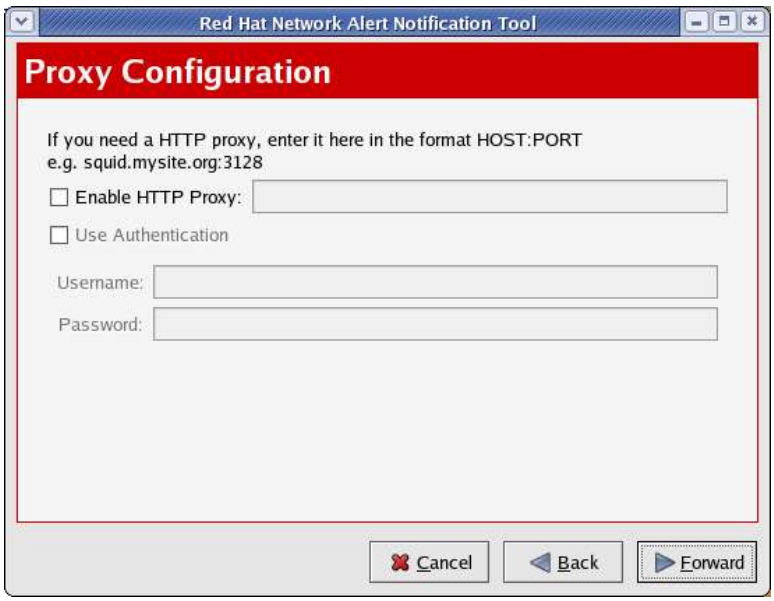

***Figure 8**. The Proxy Configuration screen of the Red Hat Network Alert Notification Tool.*

If you need to go through a proxy, you undoubtedly have configured settings like these before, and you know what to enter for your particular environment. If not, you can ignore this screen. In either case, when you're done, click Forward to move to the Configuration Complete screen as shown in **Figure 9**.

***Figure 9**. The Configuration Complete screen allows you to check for updates immediately.*

You can check for available updates simply by clicking the Apply button, or just close the tool by clicking the Close box in the upper right corner of the title bar. For purposes of this chapter, I'll discuss updates in the next section.

This wizard just gets you started. A number of advanced options are available through the Red Hat Network Configuration dialog. To get to this dialog, follow these steps:

1. Open a terminal window.
2. Issue the command

```
up2date --configuration
```

and enter the root password when prompted.

3. The Red Hat Network Configuration dialog will appear, as shown in **Figure 10**.

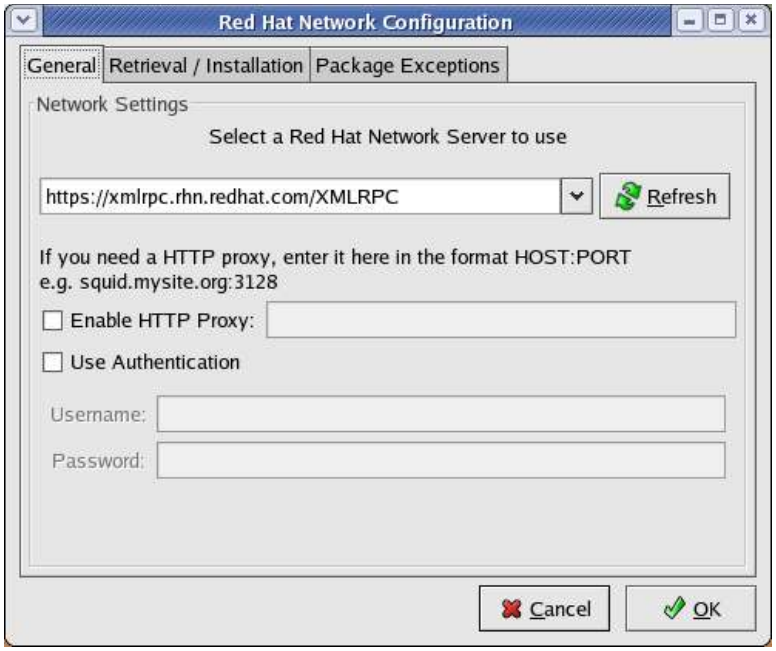

***Figure 10**. The General tab of the Red Hat Network Configuration dialog handles proxy settings.*

The General tab of the Red Hat Network Configuration dialog allows you to set which server you want to use, as well as proxy settings for your system, similar to the wizard discussed earlier.

The Retrieval / Installation tab displays a variety of customization capabilities, as shown in **Figure 11**. ##This sentence pretty much repeats the next caption. OK? JF##

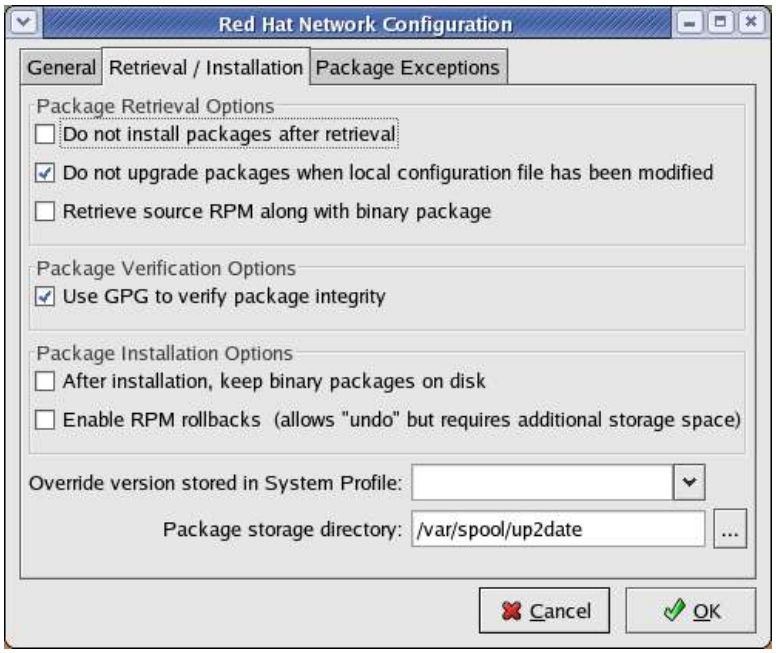

***Figure 11**. The Retrieval / Installation tab provides a variety of customization capabilities.*

Here you can specify whether or not to install packages automatically after downloading them, whether or not you want to upgrade packages after you've modified the configuration file for those packages (because upgrading would typically overwrite the changes you've made), and whether or not to retrieve the source code for the package.

The middle box allows you to automatically run the GNU Privacy Guard (GPG), a free Pretty Good Privacy (PGP) workalike, ##do we need to spell this out anywhere? JF##to verify that the downloaded packages arrived safely and are, in fact, the original packages you intended to download. If you select this option but you haven't already installed the Red Hat public key, the first time you attempt to download and install, you'll be prompted to fetch the key (see **Figure 12**).

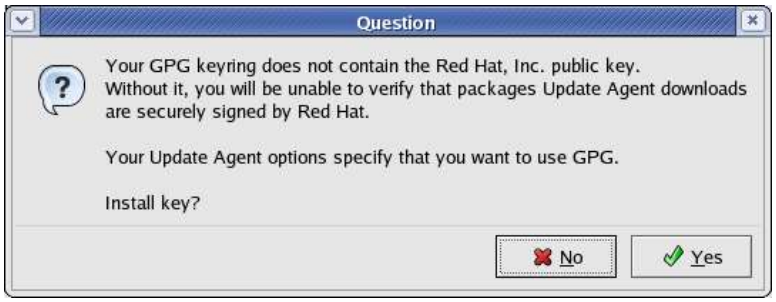

***Figure 12**. Installation prompts you to download and install Red Hat's public key if you don't have it and want to use GPG.*

The third box allows you to delete the RPMs automatically (or not) after installation, and specify whether you want to have rollback capability for the packages you're updating.

Finally, you can choose where you want to store the packages you're downloading—important if your system has limited hard disk space and the default partition in /var doesn't have room.

The third tab, Package Exceptions, allows you to specify which packages you don't want to include in a standard update operation, as shown in **Figure 13**.

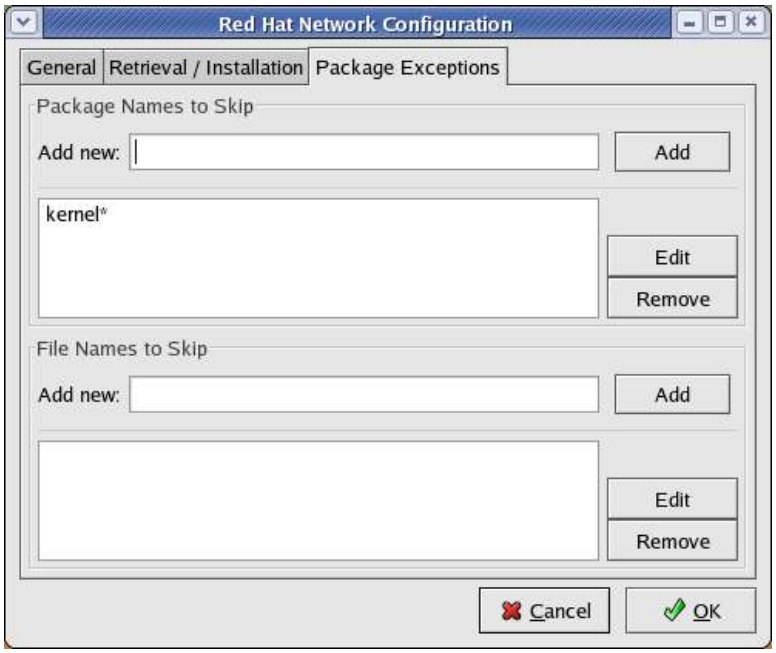

***Figure 13**. The Package Exceptions tab allows you to choose which packages you don't want to include in a standard update operation.*

By default, an update will grab all packages available on the server. You may know about some packages in advance that you aren't going to update regularly. For example, the Linux kernel is updated regularly. But many people don't want to install every single update, preferring to wait for updates that affect them. Thus, the packages that begin with the name "kernel" are by default part of the Package Exceptions group. (The asterisk denotes every package that begins with the six letters "kernel," including the kernel itself, the documentation, and the source code.)

You can use this feature to customize just exactly what shows up during the installation process, but you can also override your default choices in special cases.

When you're done with all your changes, just click OK.

4.3 Point up2date to a local mirror

As mentioned earlier, Fedora Core up2date is configured to point to a single Web site for fetching updates. As a result, this site is usually overloaded, which results in a couple of common problems. One is an abnormally slow transfer, as shown in **Figure 14**.

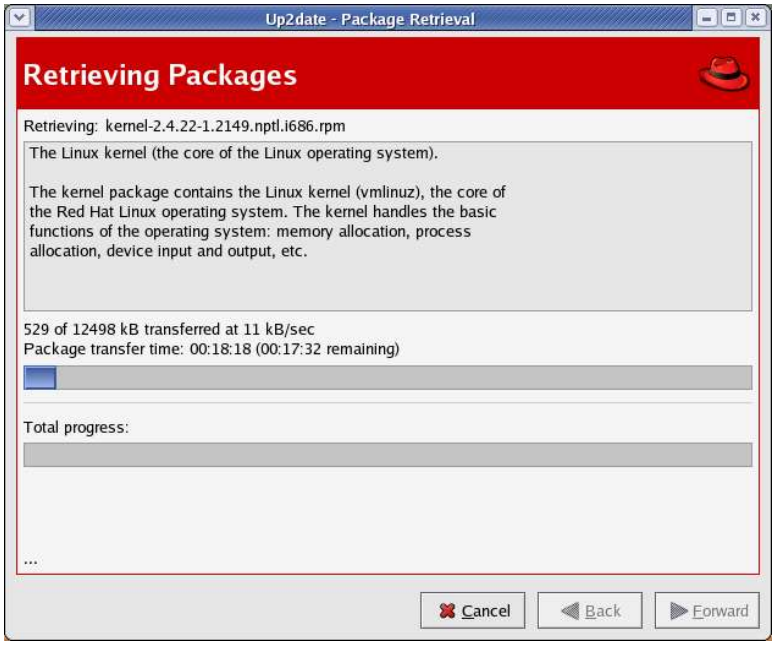

***Figure 14**. The Retrieving Packages dialog shows you the data transfer rate, among other things.*

You'll notice that the data transfer in this screen is 11 KB/sec, which is a little slow given that the receiving end is an unfettered T1 line. Another common error is shown in **Figure 15**, after the retrieval of a package.

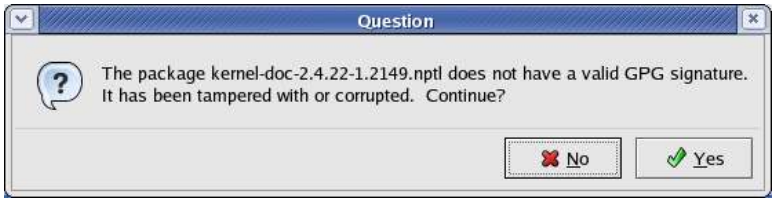

***Figure 15**. The "invalid GPG signature" error is often a result of an interrupted file download.*

What has likely happened is that the file hasn't been tampered with, but that it's simply corrupt, because the download was terminated and wasn't resumed, leaving a bad file.

These two problems are posted on the Fedora Core mailing list on a daily basis. You can avoid them by hitting a different mirror. up2date gets its configuration information from a plain text file called /etc/sysconfig/rhn/sources. In this step, you'llmodify this file and change the URL for the Web site to grab files. Here's how, in detail. ##I converted following URLs to regular text; they had been listed as code. OK? yup JF##

First, look for a mirror that is physically close to you. Browse the list of mirrors at **http://fedora.redhat.com/download/mirrors.html**. I'll use **http://www.gtlib.cc.gatech.edu/pub/fedora.redhat/linux/core/** in this example, but encourage you to use a different one, so that all of the readers of this book don't overload *that* server too! When you go to that URL, you won't see a list of files. Instead, you'll see a directory structure similar to that shown in **Figure 16**.

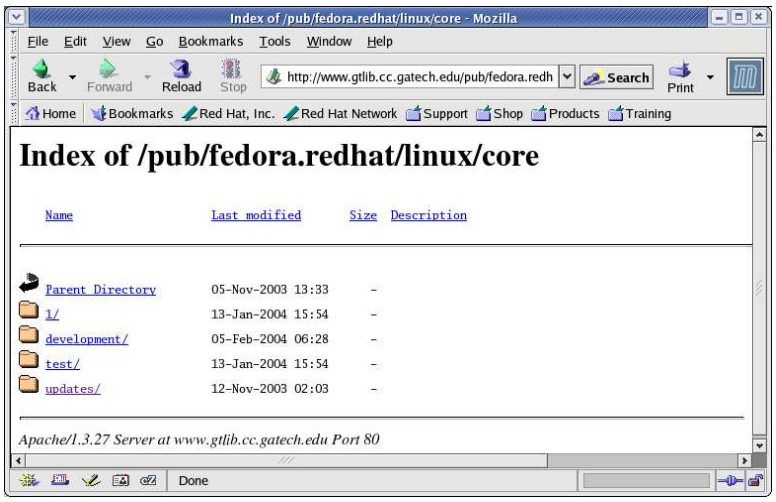

***Figure 16**. Drilling down through the updates node of a directory listing.*

Navigate through the following directories—updates, 1, and i386—until you arrive at the **http://www.gtlib.cc.gatech.edu/pub/fedora.redhat/linux/core/updates/1/i386/** directory, as shown in **Figure 17**. The URL in the address bar is the information you'll put in the sources file shortly.

Yeah, this isn't terribly friendly yet, but it'll get better in future Fedora Core releases.

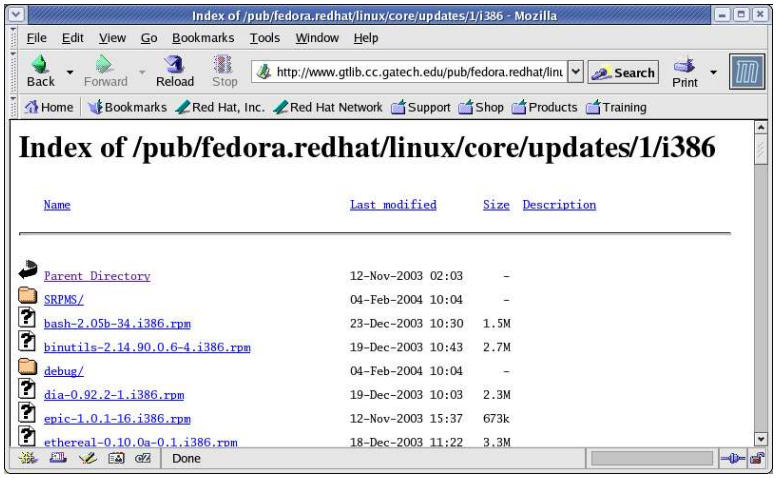

***Figure 17**. The final destination of drilling down through a directory structure is a list of update files.*

Second, open up a terminal window and switch to superuser as shown in **Figure 18**.

***Figure 18**. Editing the Red Hat Network sources file as superuser.*

In the sources file, search for the line that says

```
yum updates-released http://fedora.redhat.com/updates/released/fedora-core-1
```

Although the first word is “yum,” this works for both up2date and yum. Replace it with a line like so (this line wraps onto two lines for publication in this document):

```
yum updates-released http://www.gtlib.cc.gatech.edu/pub/fedora.redhat/linux/core/updates/1/i386
```

You’ll note that there are two other lines, one that says

```
yum fedora-core-1 http://fedora.redhat.com/releases/fedora-core-1
```

and another that says

```
yum updates-testing http://fedora.redhat.com/updates/testing/fedora-core-1
```

You can make similar changes to these lines if you’re interested in changing the servers for the core files and for the testing updates files.

In either case, save the file, and you’re ready to launch up2date.

4.4 Launch up2date

up2date is the actual mechanism that goes to the mirror site, fetches files, and installs them on your machine, all in one handy interface.

Click the “Launch up2date” menu option from the Red Hat ##Alert??## Notification Tool. You’ll be greeted with the dialog in **Figure 19** that requests you to log in as root.

***Figure 19**. First, log in as root in order to run up2date.*

Upon successful login, you’ll be greeted with the Welcome screen to the Red Hat Update Agent wizard, as shown in **Figure 20**.

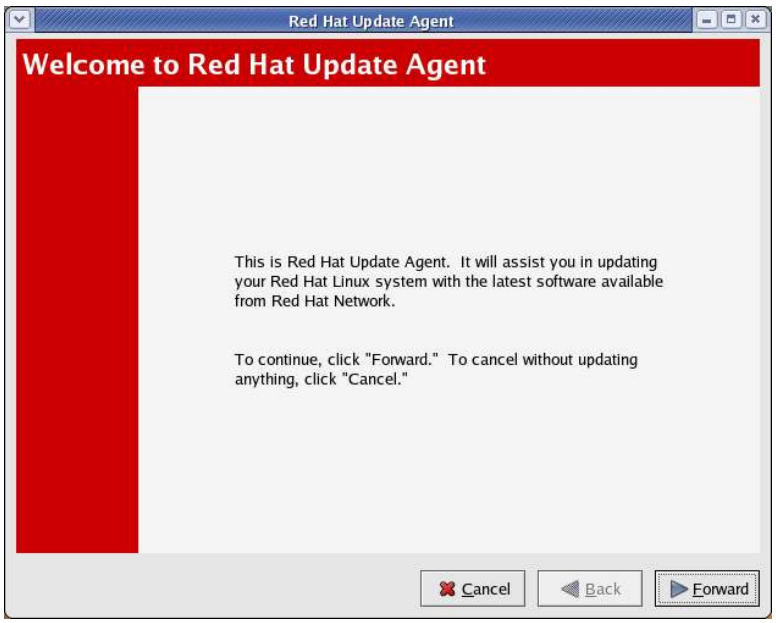

***Figure 20**. The Welcome screen to the Red Hat Update Agent.*

Nothing to see here. Move along, move along. And click the Forward button to get to the Channels screen, as shown in **Figure 21**.

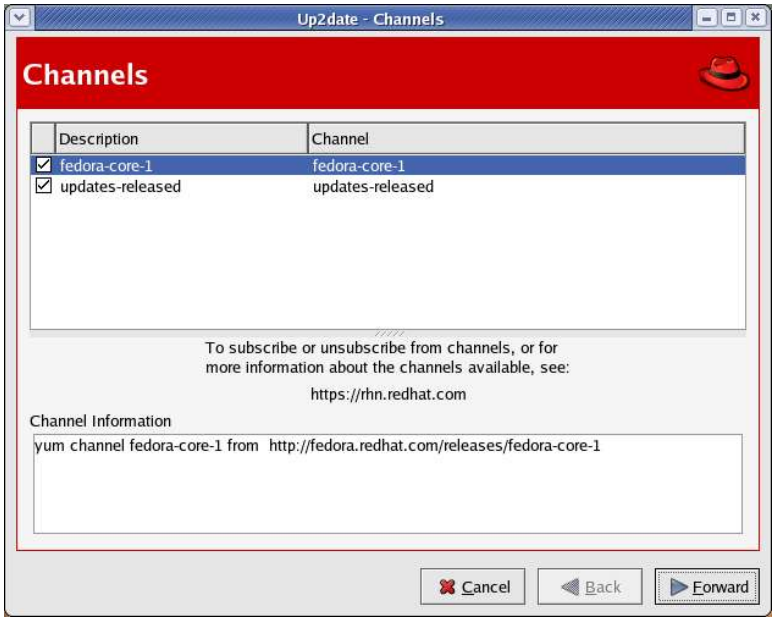

***Figure 21**. The Channels screen in the Update Agent wizard shows you which channels are associated with which type of software update.*

A channel is a mechanism that holds a certain type of software. For example, Figure 21 lists two channels in the list box in the top half of the screen. The first channel is for the original Fedora Core package, while the second channel is for updates.

In other words, the original ISOs for Fedora Core are found in the fedora-core-1 channel, while updates are found in the updates-released channel. There are other channels as well, such as “beta” builds for testing prospective releases of updates.

Highlighting a channel in the list box shows which Web site the channel is associated with. You can see in Figure 21 that the fedora-core-1 files are found at the fedora.redhat.com Web site. Selecting the “updates-released” line in the list box displays the gatech.edu Web site that we entered in the sources file earlier, as shown in **Figure 22**.

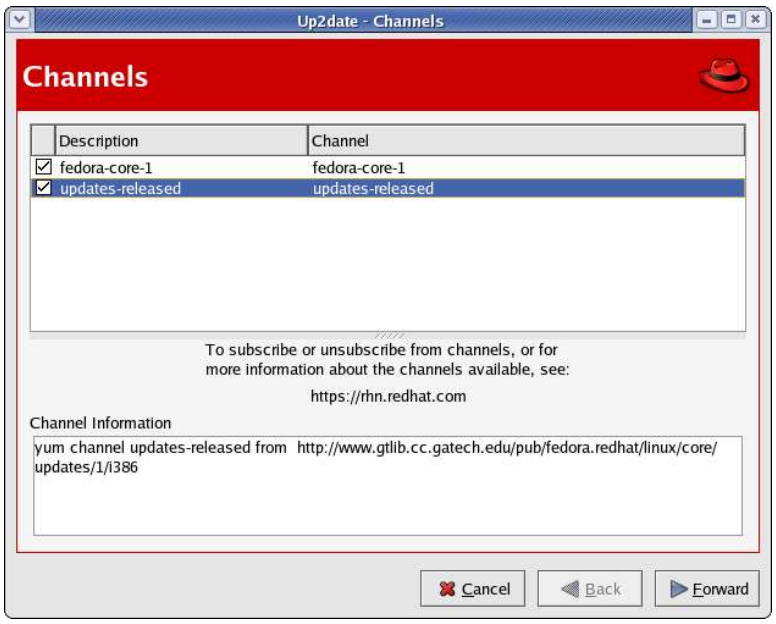

***Figure 22**. The updates-released channel points to the gatech.edu Web site.*

Click the Forward button to start the process. The first thing that the Update Agent will do is download headers from the Web site identified in the Channels screen, as shown in **Figure 23**.

***Figure 23**. The first Progress Dialog provides feedback on header retrieval.*

Once the header info has been brought down, the header for each available update is downloaded. The progress of this operation appears in another Progress Dialog, as shown in **Figure 24**.

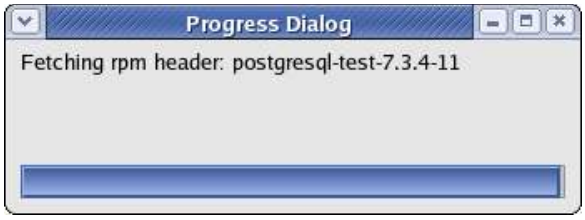

***Figure 24**. The second Progress Dialog provides feedback on individual header retrieval.*

Once all of the headers have been retrieved successfully, one of several screens will appear, depending on how your system is configured.

If you have selected some packages or files in the Package Exceptions tab of the Configuration dialog (see Figure 13), the next screen you see will be similar to **Figure 25**, which lists the packages that were flagged to be skipped.

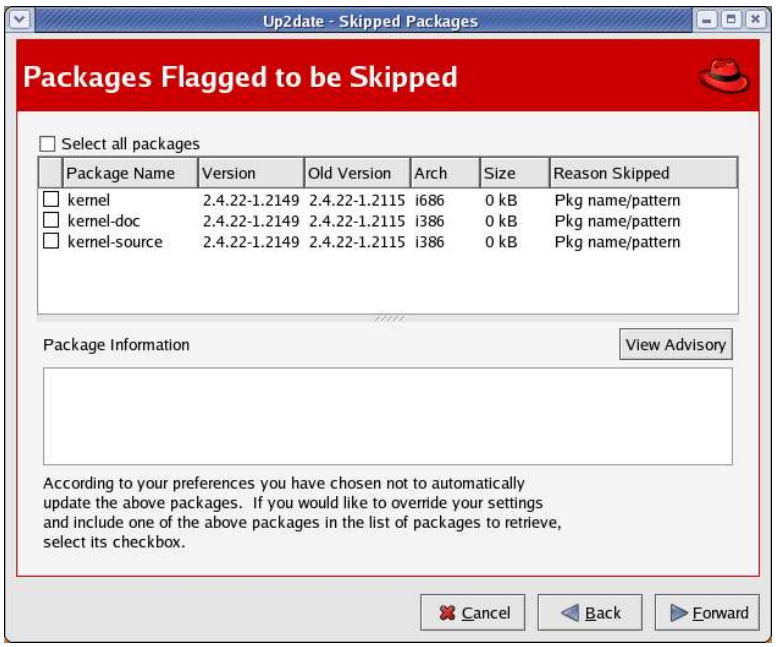

***Figure 25**. The Packages Flagged to be Skipped screen allows you to override the default settings.*

In this screen, you can choose to override the default choices and select one or more packages for installation during this run. Whether or not you choose any, click Forward to go to the Available Package Updates screen, as shown in **Figure 26**.

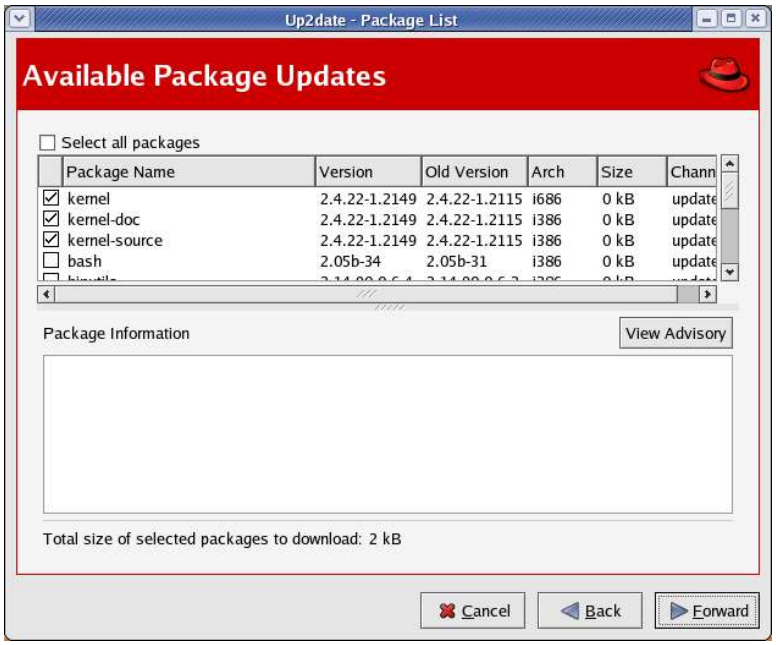

***Figure 26**. The Available Package Updates screen allows you to select which packages of those available to be installed.*

The Available Package Updates screen will appear in two situations. The first is if no packages were listed in the Package Exceptions tab of the Configuration dialog (Figure 13). In this case, Figure 24 will be skipped completely. The second situation is upon completion of the Packages Flagged to be Skipped screen.

In either case, use this screen to select which packages you want to download and install, and click the Forward button. A progress bar indicating that the package set and dependencies are being dealt with appears next, as shown in **Figure 27**.

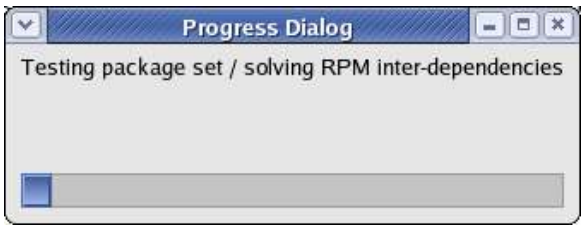

***Figure 27**. The wizard will display a progress bar while evaluating the package set and determining if any dependencies need to be dealt with.*

In short order, this dialog will disappear and be replaced by the Retrieving Packages screen, as shown in **Figure 28**.

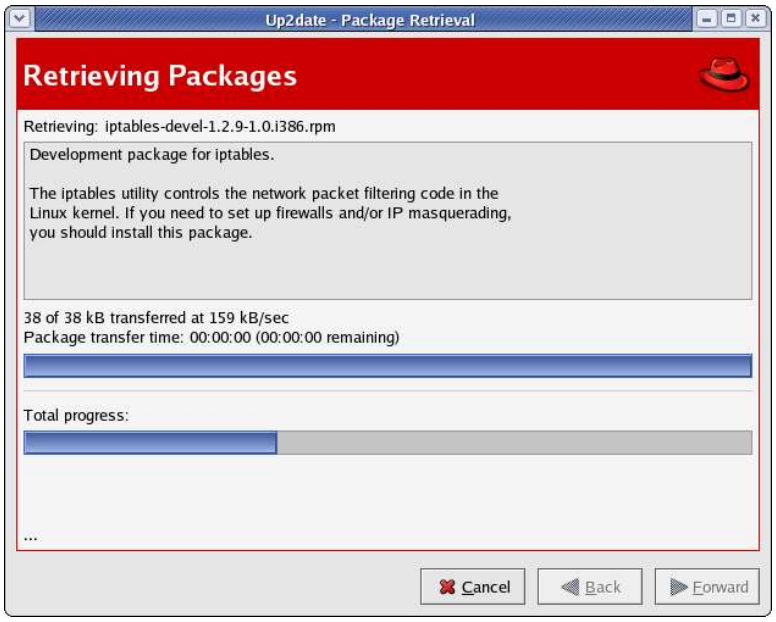

***Figure 28**. The Retrieving Packages screen displays progress on each package as well as overall download progress.*

The Retrieving Packages screen has two thermometer bars. The top bar displays the progress of the download of a single package, while the bottom bar displays the progress of the entire download process.

Occasionally, in order to solve some sort of complex dependency, other packages need to be fetched and installed even though they were not part of the original set of files. In this situation, you'll see a screen listing those packages, as shown in **Figure 29**.

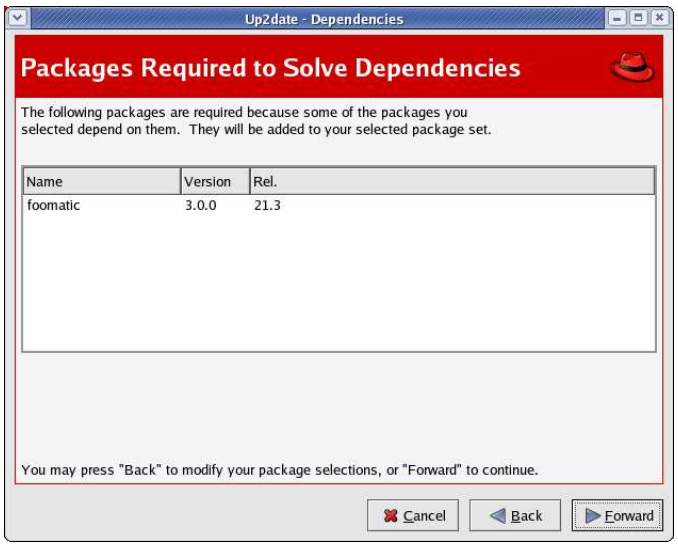

***Figure 29**. If additional packages are needed to solve dependencies, they'll be listed in an interim screen.*

If you don't want to install those packages (for example, if you know they would conflict with other packages on your system), you can click the Back button and attempt to deselect other packages that require them.

When all packages are finished downloading, an "All Finished" message appears, as shown in **Figure 30**.

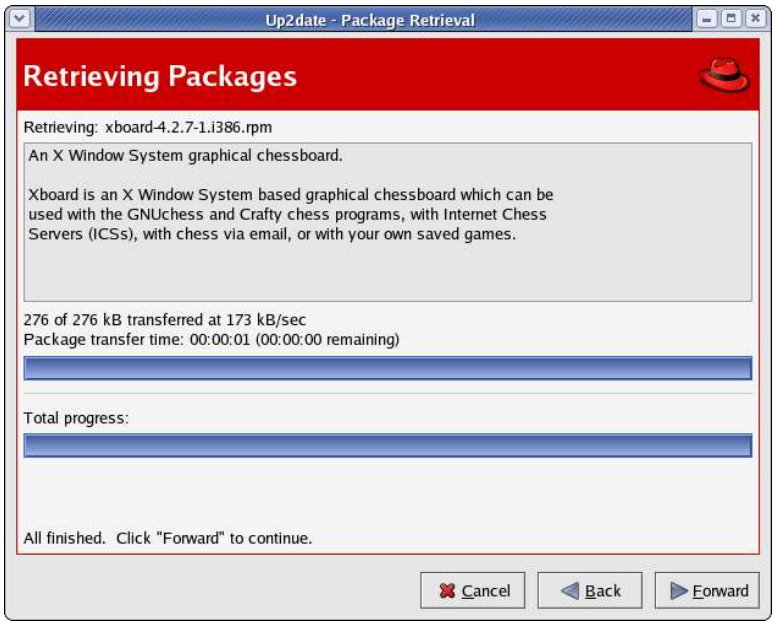

***Figure 30**. The Retrieving Packages screen displays an "All finished" message when downloading is complete.*

Click the Forward button to start installing packages, as shown in **Figure 31**.

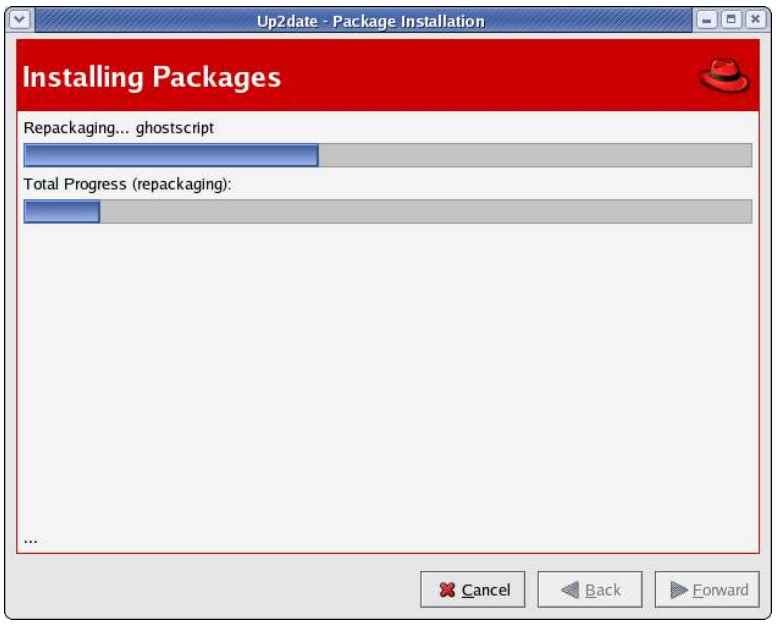

***Figure 31**. The Installing Packages screen displays progress similarly to the Retrieving Packages screen.*

Once repackaging is complete (what is repackaging? It must be important because they're doing it, but it sure seems like a detail we users don't need to know about, huh?), installation will start, as shown in **Figure 32**.

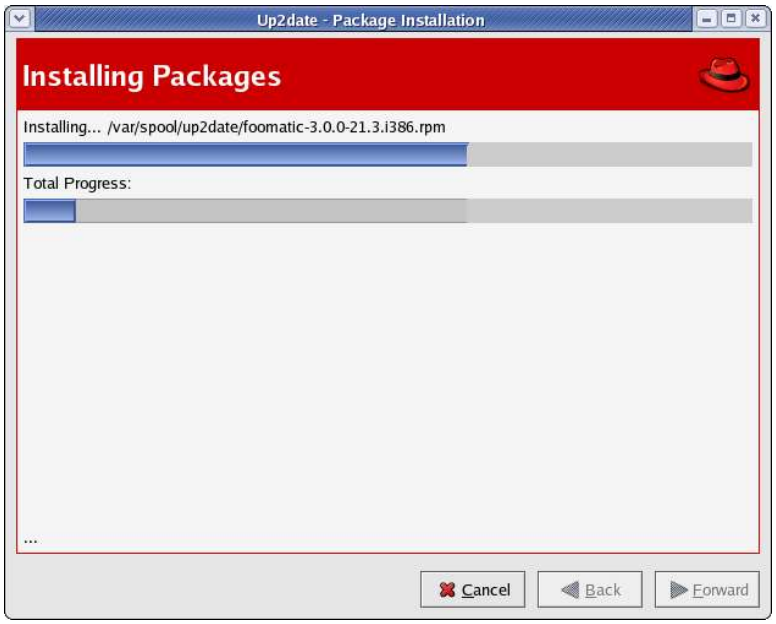

***Figure 32**. After repackaging, the Installing Packages screen displays the progress.*

Finally, the All Finished screen appears (see **Figure 33**).

***Figure 33**. The All Finished screen displays a list of all packages that were installed successfully.*

##Deleted b/c it repeats the caption exactly. JF##

If additional packages were required (refer back to Figure 29), those packages will be displayed at the bottom of the list (as opposed to being in alphabetical order), as shown in **Figure 34**.

***Figure 34**. The foomatic package is listed out of order because it was installed due to a dependency requirement.*

In either case, click the Finish button and you're all done!

Unless you're updating the kernel itself, the updates you've applied are immediately available the next time you use the application that was updated. If you've updated the kernel, you'll need to reboot your machine and select the new kernel from the boot menu screen as shown in **Figure 35**.

If you have any servers running, you'll need to restart them by clicking Main Menu | System Settings | Server Settings | Services. Or if you can't bear the thought of going days and weeks without a reboot, rebooting your machine will work too.

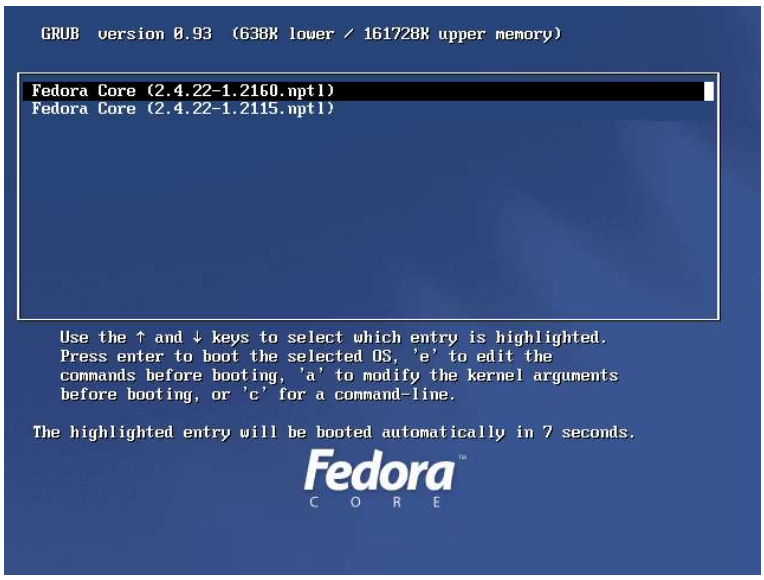

***Figure 35**. After you update your kernel, you will see multiple options in the boot menu.*

The preceding discussion is all based on the various options that occur when you right-click the Red Hat Network Alert Notification Tool icon and then select one of the menu options. If you simply click the icon, you'll see a pair of dialogs, one overlaid on the other, as shown in **Figure 36**.

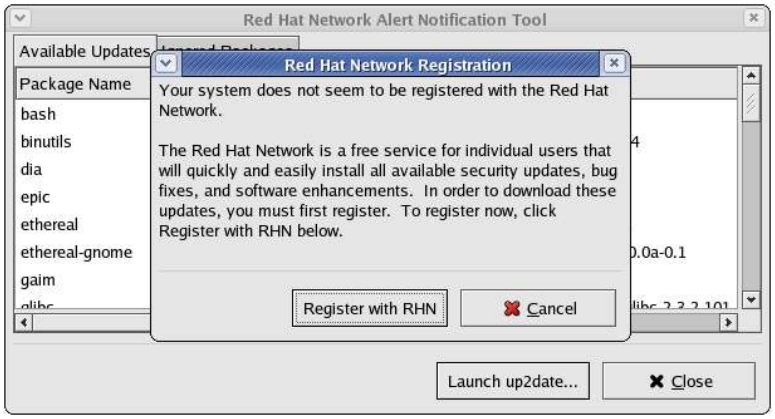

***Figure 36**. Dialogs that appear when you (left) click the Red Hat Alert Notification Tool icon.*

The Red Hat Network Registration dialog on top is a funny little beast—a throwback to commercial versions of Red Hat Linux. If you click the "Register with RHN" button, you'll be prompted for the root password, as in Figure 19, and then you'll see the "Welcome to Red Hat Update Agent" screen, as shown in Figure 20.

If, on the other hand, you click Cancel, the Available Updates tab of the Red Hat Network Alert Notification Tool will appear, as shown in **Figure 37**.

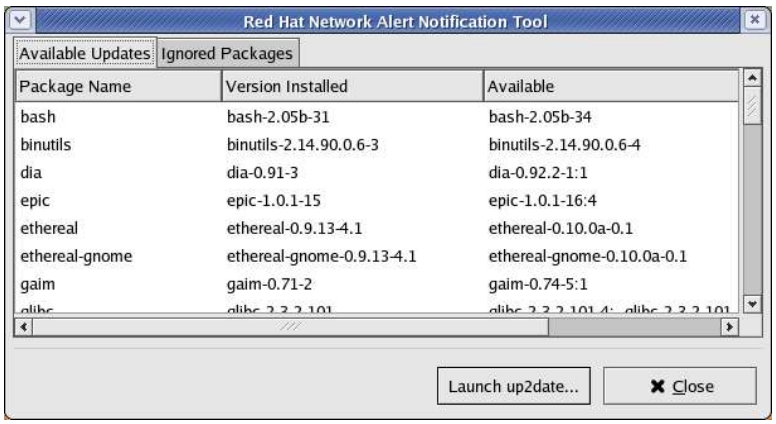

***Figure 37**. The Red Hat Network Alert Notification Tool is a remnant of the commercial version of Red Hat Linux.*

Then, when you click the Launch up2date button, you'll be prompted for the root password, and then you'll see the same "Welcome to Red Hat Update Agent" screen—as if you'd clicked the Register with RHN button in the previous screen. Go figure.

5. Conclusion

The mechanisms for updating Fedora Core are possibly one of its best-kept secrets. I've heard Windows users claim superiority despite the security holes constantly found in that operating system—because "Windows is easy to update; just click 'Windows Update' and that's that!"

Fedora Core's mechanism is actually better, both because updates require rebooting only if the kernel is modified, and because the source of updates is distributed across many servers on the Internet. If you take the time to configure up2date for your location, updating couldn't be simpler!

6. Where to go for more information

This free whitepaper is published and distributed by Hentzenwerke Publishing, Inc. We have the largest lists of "Moving to Linux", OpenOffice.org, and Visual FoxPro books on the planet.

We also have oodles of free whitepapers on our website and more are being added regularly. Our Preferred Customer mailing list gets bi-monthly announcements of new whitepapers (and gets discounts on our books, first crack at special deals, and other stuff as we think of it.)

Click on "Your Account" at www.hentzenwerke.com to get on our Preferred Customer list.

If you found this whitepaper helpful, check out these Hentzenwerke Publishing books as well:

Linux Transfer for Windows® Network Admins:
A roadmap for building a Linux file and print server
Michael Jang

Linux Transfer for Windows® Power Users:
Getting started with Linux for the desktop
Whil Hentzen