Red Hat Interchange 4.8: Getting Started Guide

Draft

Table of Contents

CHAPTER 3   Frequently Asked Questions

PART II: **INSTALLING AND SETTING UP INTERCHANGE**

CHAPTER 4   Installing Interchange 4.8

CHAPTER 5   Building and Setting up a Catalog

CHAPTER 6   Maintaining Interchange

List of Tables

Part I:
Before You Begin

CHAPTER 1   Introducing Red Hat Interchange 4.8

The purpose of the Interchange 4.8 Getting Started Guide is to provide you with a high level understanding of the Red Hat Interchange platform and the process involved in setting up and running Red Hat Interchange. This guide will also provide you with style and user formats used throughout the Red Hat documentation set.

Style Conventions

The following style conventions are used throughout this and all Red Hat e-business documentation.

Italic Text—A term defined in the context of the sentence, or the title of other reference material.

Bold Text—A file name, window name, or function.

`Courier New Text`—User input or system output.

!!! Information that requires strict observance to ensure proper operation or to avoid system failure.

Note: Helpful or supplemental information, as well as tips and shortcuts.

Interchange 4.8 Documentation Set

The **Getting Started Guide** is the first guide in the Interchange 4.8 set. Table 1 lists the complete Interchange 4.8 documentation set.

TABLE 1. Interchange 4.8 Documentation Set

DOCUMENT	DESCRIPTION
Red Hat Interchange 4.8: Getting Started Guide	Gives an overview of Interchange, lists the prerequisites and system requirements, and describes the procedures for installing, setting up, and maintaining Interchange.
Red Hat Interchange 4.8: Administration Guide	Describes how the Interchange Administration Tool can be used and customized to administer Interchange.
Red Hat Interchange 4.8: Tag Reference Guide	Lists and describes the Interchange Tag Language (ITL) tags.
Red Hat Interchange 4.8: Reference Guide	Provides a directory of reference material to help you customize your Interchange platform.

These documents are included in the Interchange installation package and are also available on the Interchange developer resource site at http://developer.akopia.com/cgi-bin/ic/dev/index.html.

Interchange System Requirements and Prerequisites

This section describes the minimum software and hardware requirements need to run Interchange 4.8.

Minimum System Requirements

The following are required of your computer for proper installation of Interchange.

- 400 MHz Pentium II or higher computer
- 128 MB of RAM

Prerequisites

Interchange requires the following to be installed and configured on your computer.

- UNIX or Linux operating system.
- Web server (Apache recommended).
- Perl 5.005 or later.
- Database or appropriate Perl module (GDBM, DB_File library, or SQL database recommended).
- Perl module bundle (Can be installed with Interchange).
- Encryption Package (GPG or PGP recommended) or web server that supports Secure Socket Layer (SSL) encryption. This is necessary only if you are accepting sensitive credit card information.
- An in depth working knowledge of your operating system.

Note: Setting up an SQL database is beyond the scope of this document. For assistance in setting up your SQL database, contact customer support.

Getting Help

This section describes the options that are available to you for receiving help.

Contacting Interchange Experts

The following support options are available to you:.

- **Interchange User Group support**—email interchange-users@akopia.com to post a question to Interchange users. To subscribe to the Interchange User Group, go to http://lists.akopia.com/mailman/listinfo/interchange-users. The Interchange User Group should be your first resource for obtaining support, since it allows you to contact thousands of Interchange users for advice or assistance.
- **24-hour phone support**—contact Customer Support for fee-based phone support at 1-703-456-2911.
- **Email support**—email support@akopia.com to request fee-based support from Red Hat's professional support staff.

Reporting a Bug

Red Hat Interchange is an open source e-commerce software platform. *Open source* means that the source code is delivered with the software and can be viewed and manipulated by the user. This format allows the product's users to become the product's developers. Each new version of Interchange is rigorously tested prior to release. However, if the occasion arises where an Interchange user finds a bug in the software, he or she can report that bug to the Red Hat development team for resolution. A *bug* refers to an error or deficiency in the code that affects the performance of the software. To report a bug, go to **Bugzilla** at http://developer.akopia.com/bugs/ and enter the necessary information. The Red Hat core development team monitors Bugzilla regularly and the bugs are fixed as promptly as possible. Each subsequent release of Interchange is accompanied by a list of resolved bugs.

> *Note:* The fastest way to get your bug fixed is to give as much detailed information as possible when reporting it. Bugs that are reported with

supporting research, bug recreation steps, and a proposed resolution will get attention faster than bugs that are vague and unsubstantiated.

About Red Hat Interchange

Red Hat Interchange is the industry's leading open source e-commerce software platform. It allows all merchants, both large and small, to manage their Internet storefront, merchandising, and backend processes. Interchange's extensive feature set is based on years of evolution and experience in the open source community.

This chapter discusses the benefits of using Red Hat Interchange, describes how Interchange works, and provides you with a checklist that outlines the steps involved in installing, setting up, and running Interchange.

The Open Source Advantage

Red Hat Interchange leverages the advantages of open source technology to form a superior product. This is because Red Hat Interchange users work together to improve the product. Over the past five years, the software has been downloaded over one million times by developer customers all over the world. These developers find what works for them, what doesn't work, what could make Interchange better, and they change it. These changes or enhancements are reviewed and tested by the Red Hat core Interchange development staff. Once approved, these enhancements are included in the next release of the product code. The

contributions from the development community give Interchange users the advantage of an extremely rapid improvement and enhancement life cycle, as well as a truly superior product.

Feature enhancements that take many months in a traditional development environment to research, develop, and implement can take only a few weeks in an open-source development environment. This is because our developer customers take advantage of the Interchange community to exchange ideas and suggestions. One user might find a bug that could be fixed within hours by a team of developers working on the problem together.

The Interchange Community

Once you download Red Hat Interchange, you are instantly a member of the Interchange community. The Interchange community shares ideas and helps members with Interchange related problems. The members of the Red Hat development team are also active members in the Interchange community. Red Hat considers input from the Interchange community, whether submitted by a technical user or a non-technical user, to be very important in the development of Interchange.

The following is a list of the opportunities to become involved in the Interchange community:

Announcement Mailing List—provides you with email announcements of new releases and other news about Interchange. To subscribe to the Announcement Mailing List, go to http://lists.akopia.com/mailman/listinfo/interchange-users.

User Group Mailing List—provides a high volume email forum for questions, suggestions, advice, and so on. To subscribe to the User Group Mailing List, go to http://lists.akopia.com/mailman/listinfo/interchange-announce.

Feature Wish List—provides a place for users to list and describe features that they would like to see included in future releases. To submit to the feature wish list, go to http://developer.akopia.com/cgi-bin/ic/docfly?mv_session_id=UjYp9iEF&mv_arg=wishlist0

Bugzilla—stores all resolved and yet to be resolved Interchange bugs. Browsing this database allows you to search for problems that may have recently been resolved by the

development team or submit a bug that you have discovered. For information on how to submit a bug, see *Reporting a Bug, page4* .

Interchange 4.8 Components

Once installed, Interchange 4.8 provides two tools for getting started with your e-business site, the Foundation Store and the Administration Tool. The *Foundation Store* provides a skeletal site structure that demonstrates some of the fast functionality of Interchange and gives you a base for building your own store. The *Administration Tool* provides an interface for managing the backend functions of Interchange, such as uploading pages and images, administering users, and managing product merchandising and customer orders. For more information on the Administration Tool, see *Red Hat Interchange 4.8: Administration Tool Guide*.

Understanding the Foundation Store

The Foundation Store was designed to help you to understand some of the functionality of Interchange by seeing it in use. Also, by replacing the graphics, logos, and text, you can Through example, you can better understand how the Interchange model functions. , shows the Foundation Store as it looks when installed.

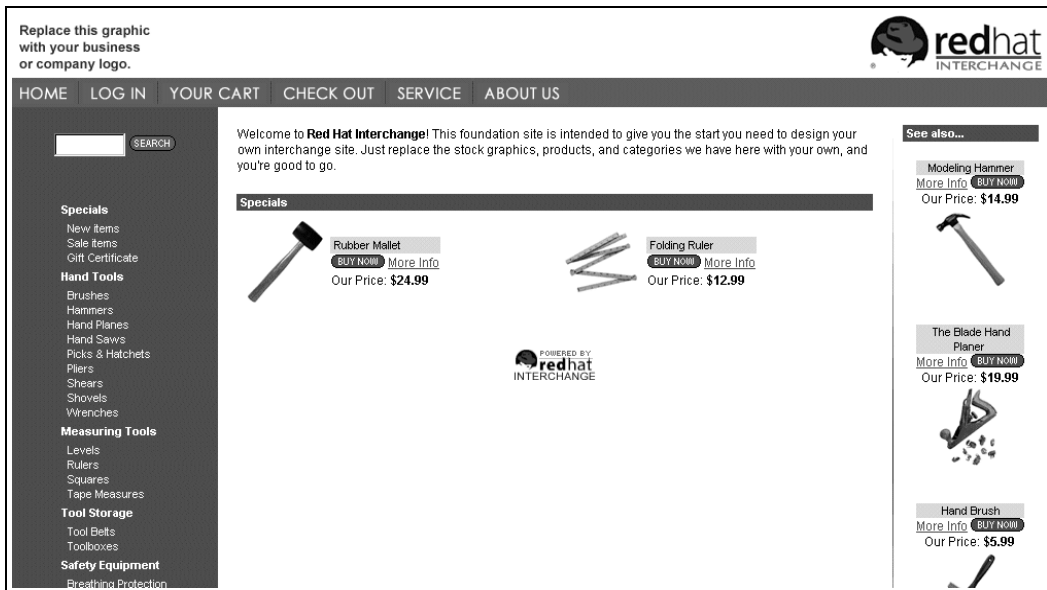Table 1 describes some of the features for displaying products on the Foundation Store.

TABLE 1. Features of the Foundation Store

FEATURE	DESCRIPTION
Specials	Select products to be displayed in this section.You can control how many products are displayed at one time and how they are displayed. You can also have a selection of products rotate through the specials section, so that each time a customer loads a page the products accented in this section rotate. Promotion (new items, something as part of a promotion until expired)
See also...	Select complimentary products to be displayed with a product or catagory of products. When a product is displayed, the complimentary or "cross-sell" products are featured in this section. For example, you might choose to cross-sell nails with hammers.

TABLE 1. Features of the Foundation Store (Continued)

FEATURE	DESCRIPTION
Search	Click Search to envoke the Interchange search feature. You can set the search function to match partial words and phrases. .
Services	Showcases the different customer service functions of the Foundation Store. The following is a list of these customer functions: • Help—accesses your **Help** page. • Account Maintainance—alter account information. • Order Tracking—track the status of orders. • Additional Shipping Addresses—add a shipping address to an account. • New Account—add a new user account. • Order Returns—return a purchase after it has been shipped. • Saved Carts/Recurring Orders—explained below.
Log In	Allows customer log in to your site to access historical information, such as shipping addresses and pending orders.
Save Cart (not shown)	The Save Cart feature is new to Interchange 4.8. It allows customers to save items in a named shopping cart for a one-time future purchase. When the saved cart is accessed again the contents are displayed and are ready for purchase. This allows the customer to continue a shopping experience and complete the purchase at a later date.
Recurring Order (not shown)	The Recurring Order feature is new to Interchange 4.8. It allows customers to save items in a named shopping cart for future use. The Recurring Order shopping cart items can be accessed and purchased multiple times until the cart is deleted. This allows the customer to purchase a recurring product or set of products when necessary.

Note: Red Hat is constantly working to improve Interchange. Therefore, technical releases of Interchange are made public every few weeks containing minor improvements or patches. Check the Interchange developer site at http://developer.akopia.com to download the latest version of Interchange. Red Hat also sends an email to the user group mailing list when a new version of Interchange is released.

Understanding the Administration Tool

The Interchange Administration Tool allows you to access and administer Interchange from a web browser. Table 2 shows an example of the Administration Tool.
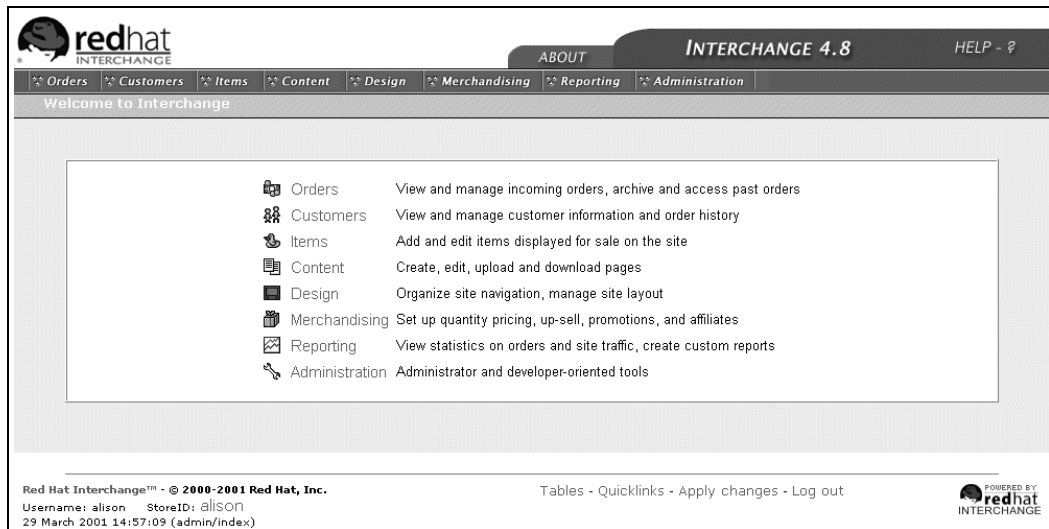

FIGURE 2: THE INTERCHANGE ADMINISTRATION TOOL

The Administration Tool allows you to do the following:

- View, enter, and archive customer orders.
- Upload and download site content.
- View and change customer information as well as change customer status.
- Organize site information, create search mechanism, build links and buttons, and create static pages.
- Edit existing products, add new products, and specify inventory tracking.
- Generate reports and analyze the trends of your site and affiliate traffic.
- Implement merchandising features such as cross-sell, up-sell, promotion, and quantity pricing.

- Access administration information such as shipping, payment, tax, and access preferences as well as the Interchange data tables.

Understanding Red Hat Interchange

This section describes some of the components and functions of Interchange.

Commonly Used Terms

To help you to better understand the language used throughout the Interchange documentation to describe the unique functionality of Interchange, Table 2 provides a list of terms.

TABLE 2. Interchange Terminology

TERM	DEFINITION
Session	A temporary storage area that allows Interchange to maintain information about a specific website visitor.
Session ID	A temporary identifier embedded in a URL or customer cookie that associates a user with a session.
ITL	Interchange Tag Language. A library of tags used to format text, define action, or perform a function. An ITL tag is designated by square brackets (**[ ]**). The ITL is exclusive to the Red Hat Interchange product.
Database	A repository containing product, template, and user information in the form of tables or database files.
Template	A base structure for building web pages.
Catalog	A compilation of your database, web pages, and variables that come together to make your web site.

The Interchange System

Figure 3, page 15, shows a high-level overview of how your systems interact with each other and with Interchange.
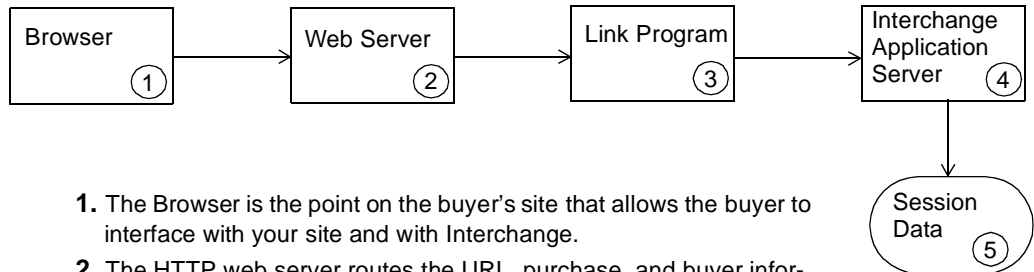
1. The Browser is the point on the buyer's site that allows the buyer to interface with your site and with Interchange.
2. The HTTP web server routes the URL, purchase, and buyer information through the Link program.
3. The Cgi-bin secures the integrity of your Interchange application, allows for load balancing, and allows more flexibility with the HTTP server you decide to use.
4. Interchange creates a session for the buyer's user experience. Interchange also checks for a cookie. If the buyer is new to your site, a cookie is created with a session ID. If the buyer has cookies disabled, a session ID is appended to the URL for use throughout the session.
5. As the buyer makes purchases, Interchange uses the session data

FIGURE 3: THE INTERCHANGE SYSTEM

About the Interchange Components

The three main components that comprise Interchange are the catalog, the template, and the Interchange tags. This section gives an overview of these components.

Catalog. The catalog that you create represents your entire store. The catalog contains your database (which consists of product information, customer information, and so on), web pages, and variables. You can create multiple catalogs, or multiple stores, using one installation of Interchange.

Templates. The templates are the skeletal structure used to build your web pages. Your web pages are compiled by inserting variables, Interchange tags, and HTML into the template. One template can be used to build an infinite number of web pages. A user that has the same banner at the top of every web page on his or her site can build that element into the template. As long as that particular template is used to build the web page the banner will be displayed.

Interchange Tag Language. The Interchange Tag Language (ITL) was created specifically for use in Interchange. There are more than 80 distinct Interchange tags that, either alone or combined, support hundreds of functions. Interchange also allows user-defined tags to be created. User-defined tags are as powerful as regular Interchange tags and are used for addressing specific customer needs.

The following is an example of how an Interchange tag is used to create a link from a page. The ITL tag is included in square brackets [ ] in the HTML code.

<A HREF="[href specials]">See our Specials!</A>

The [href specials] tag represents the specific URL of the page to be linked to. If the URL changes, the user only needs to change the [href specials] tag without having to edit many HTML pages, saving hours of effort. The Interchange tags also allow Interchange to potentially generate thousands of pages from a small number of templates.

Interchange Functions

Using the components described in *About the Interchange Components, page 15*, Interchange can perform many different functions. This section describes some of the most popular functions that showcase the use of ITL tags.

User Form Input. Interchange captures input from a user so that it can be easily passed between forms. The value of any form variable is "remembered" and inserted upon finding a [value input_field] tag, where input_field is the name of the HTML form field. For example, a user who typed his or her warehouse receiving address in one form will see it automatically populated another form using the [value input_field] tag.

Database Contents. Interchange can support an unlimited number of attached databases, either in one of its own internal formats, one of the SQL databases, or in any database with an ODBC interface. The contents of a database can be referenced with tags, for example:

[data table=products column=name key=334-12] or [query sql="select * from products where category = 'Computers'"]

Session Parameters. Session parameters include information specific to a particular user and his or her current interaction with Interchange. The parameters include the location

where the user originally found the catalog [data session referrer], the domain they are from [data session host], the source of the hit in a partner program [data session source], which is created by passing mv_pc=something on the URL, the time of their last access [data session time], and so on.

Embedded Perl, ASP, and External Programs. Interchange has a powerful object system that allows direct access to Perl and to external programs. An ASP-like syntax can be used, or the traditional Interchange tag approach can be employed.

File Contents or Program Output. The contents of a file can be inserted with [file directory/file] or [include directory/file]. It is also possible to include the output from another program.

Searches of Files. Interchange supports a variety of search engines, including *Glimpse*, a popular full-text search engine. It can also process the output from custom SQL database queries. Interchange itself has a search capability that can be used to search files.

Session Management

Because on the Internet each request for a web page is processed independently, it is normally not possible to identify if the next request comes from the same browser or even from the same user on that machine. Interchange automatically keeps track of each user session by one of two methods.

Cookies. Interchange issues cookies that contain the user session ID. If the user returns the cookie, he or she is presented pages without accompanying session information.

URL Re-writing. If the user blocks the use of cookies, Interchange includes the session ID in the URL. This is a string of text that is unique for each customer browsing the catalog. The text allows Interchange to sort information presented to the system from simultaneous users.

Red Hat Interchange Checklist

Table 3 explains the steps involved in preparing for and successfully installing, configuring, and running Red Hat Interchange, as well as where you can find further information on each step.

> ***Note:*** For information about system requirements and prerequisites, see *Minimum System Requirements, page 3*.

> ***Note:*** You can print this table to be used as a checklist when installing your Interchange system.

TABLE 3. Steps for Setting up Interchange

STEP	DESCRIPTION	REFERENCE	STEP COMPLETE
1	Install Perl 5.005 or newer. This installation or upgrade is required before Interchange can be installed.	www.perl.com	
2	Install Web Server. Apache 1.3.14 or higher is recommended.	www.apache.com	
3	Install Interchange 4.8. This also installs the bundle of Perl modules needed to run Interchange.	*Installing Interchange 4.8, page 1*.	
4	Set up your catalog. This includes building your product tables, shopping cart, and templates.	*Building and Setting up a Catalog, page 1*.	
5	Customize Interchange using your Administration Tool. This includes setting up Administration user permissions, merchandising, product information, affiliate relationships, and special pricing.	Red Hat Interchange 4.8: Administration Tool Guide	

TABLE 3. Steps for Setting up Interchange (Continued)

STEP	DESCRIPTION	REFERENCE	STEP COMPLETE
6	Configure payment processing solution. This allows you to accept payment from customers.	Your payment processing network or mechanism.	
7	Advanced customization. Fully take advantage of the Interchange platform by implementing more advanced features.	Red Hat Interchange 4.8: Reference Guide	

Frequently Asked Questions

This chapter provides some frequently asked questions about the Interchange e-business platform. They are intended to help you to understand Interchange as you get started with using it. For more frequently asked questions about Interchange, see the Red Hat development website at:
http://developer.akopia.com

What is the history of Red Hat Interchange?

Interchange is based on the convergence of two products, MiniVend and Tallyman. Both of these products had an area of superior functionality, and Red Hat combined the best of each to create the Interchange product. Since the inception of the Interchange software, it has evolved even further to make it the industry's most widely distributed and implemented open source e-commerce platform.

What type of platform is Interchange?

Interchange is a combination of many components. In addition to internal components such as the application server, link program, ITL tags, and templates; Interchange depends on third party components such as Perl libraries and modules, a web server, and links to several other subsystems such as SQL databases, CyberCash, PGP, and the Glimpse search engine. Interchange is a complete access and retrieval application. It uses no more memory than a large

database server. It is optimized for catalogs of more than 100 items and catalogs that will change and grow over time.

Can I use Interchange with my existing static catalog pages?

Yes, but it is recommended that you convert to the database-driven model since Interchange is designed to build pages based on templates from a database. Static pages are difficult to maintain and are easy to convert to the Interchange database-driven model.

If you decide to use your existing static catalog pages use the following format to create links to the Interchange order system from within a static page, replacing `/cgi-bin/simple` with the path to your Interchange link.

```
<a href="/cgi-bin/simple/order?mv_order_item=SKU_OF_ITEM">Order!</a>
```

Will Interchange run at an ISP?

Most Internet Service Provider (ISP) web servers provide some CGI service. However, most place limitations on the kind of software you can run. The system requirements for Interchange are not extensive, but the software is sophisticated and beyond the scope of web counters and simple shopping carts. Check with your ISP for more information. Keep in mind that virtual servers that do not provide shell access will not operate with Interchange without support from your ISP.

Also keep in mind that Interchange requires a stable platform to run properly. Many ISP servers are heavily loaded and may frequently run out of memory and file descriptors. This environment will affect your Interchange system performance. It is recommended that you find an ISP that already provides Interchange hosting, a good virtual server, or dedicated server provider. Red Hat provides both virtual (shared) and dedicated hosting that is tailored to Interchange users.

Why is Interchange a daemon?

A *daemon* is a program that is always running in the background of a system. Because Interchange has many configuration options and can serve hundreds of catalogs, it was created as a daemon to avoid having to load and compile it each time it is used.

Because Interchange is always running in the background, its rich set of features can be accessed quickly. However, when a configuration file is modified, Interchange must be notified so the necessary information can be reloaded and affected operations can be

reconfigured. This is accomplished by restarting the server or using the **reconfig** script to reconfigure an individual catalog.

Can I use an HTML editor to build my Interchange web pages?

Because of the nature of ITL, some HTML editors will reject or reformat the tags because they consider it to be foreign code. It is recommended that you edit your Interchange pages manually or by using the Interchange Administration Tool.

Part II:
Installing and Setting up Interchange

Installing Interchange 4.8

This chapter describes the process involved in installing Interchange.

Installing Interchange

The Red Hat Interchange 4.8 platform can be downloaded from the Red Hat web site or installed from the CD and can be installed from the **RPM** or the **tarball**. Interchange can be downloaded from the Red Hat web site or installed from the CD. If you are installing Interchange on a virtual or shared server, it is recommended that you perform the install from the **tarball**. The **tarball** allows you to specify the directories where the files should go. If you are installing Interchange on a dedicated server running Red Hat Linux or Linux Mandrake, it is recommended that you perform the install from the **RPM**. The **RPM** installs the files to default directories on your server.

> **Note:** There are many ways to install a program. The procedures described below are recommendations only.

Performing an RPM Installation

To install Interchange from the RPM:

1. Locate the **interchange-4.8-i386.rpm** file.

2. Download the file to your home directory.

3. Run **rpm -Uvh interchange-4.8-i386.rpm**

4. Type 'cd/usr/lib/interchange'.

5. Access Interchange.

6. Run **'bin/makecat'**.

Performing a tarball Installation

To install Interchange from the tarball:

1. Locate the **interchange-4.8.tar.gz** file.

2. Download the file to your home directory.

3. Run **tar xzvf interchange-4.8.tar.gz** to untar the package.

4. Access the **interchange-4.8** directory.

5. Run **./configure**

6. At the prompt, enter the path to the directory where Interchange should be installed.

7. CD to the directory where you installed Interchange

8. Run '**bin/makecat'**

The Interchange Directory Structure

Depending on how you installed Interchange, the location where the files were installed will be different. This section lists and describes some of the more common files and directories.

dist/ the distribution files, exclusive of executable files and modules. Includes:

> **foundation/** the new catalog demo application
>
> **interchange.cfg.dist** distribution interchange.cfg
>
> **lib/** User interface
>
> **src/** C and Perl code for CGI links
>
> **usertag/** some fairly-standard usertags

extra/ some not-always-needed Perl libraries
lib/ the library modules needed to run Interchange
pod/ manual pages, viewable with perldoc
scripts/ the executable files, relocated to bin/ in the run directory
test.pl the installation test script

Building and Setting up a Catalog

After Interchange is installed, you need to build your catalog. A *catalog* represents your store and contains all of your product and customer information as well as your templates, static web pages, images, and database tables. You can build as many catalogs as you want using one application of Interchange.

Running Makecat

Included in the installation of Interchange is the makecat program. *Makecat* is designed to establish your system and catalog settings by asking a series of questions. The questions asked during the initial run of the makecat program are different on each subsequent run. This is because the initial run of makecat asks some base configuration questions that configure your server. Also, depending on the values that you enter during the catalog build, the questions asked may vary.

> ***Note:*** Before you begin building your catalog you must have the catalog template you are using downloaded to the same directory that contains the makecat file. The **Foundation** template is included with Interchange. Other templates are available online at the following location:

http://developer.akopia.com

To run the makecat application, type **./makecat** and press enter. The **makecat** file is located in your **bin** directory. Default answers are supplied. If you make a mistake during the program, enter a @ at the prompt to return to the previous question.

> *Note:* The answers you provide during the **makecat** process can be changed by editing the configuration script.

Running Makecat From the Initial Installation

The following questions are asked during the initial Interchange catalog build.

Catalog Name?
Enter the name you will be using for this catalog. The typical naming convention used for catalogs is to give it the same name as your store, or a name that you associate with your store. This makes it easier to identify the appropriate catalog if dealing with multiple stores.

> *!!!* It is strongly recommended that your catalog name(s) be lower-case. The catalog name will be included in the URL and should therefore be easily remembered and typed by your customers.

Httpd conf
Enter the directory location of the configuration file of your http server.

Default Server
Enter the directory location of the server on which Interchange is located

Do you have a CGI directory?
Enter a **y** if you have a dedicated cgi-bin directory.

CgiDir?
Enter the location of your cgi-bin directory. A file with your store's name is created in this directory. This folder establishes a link between your web server and the Interchange server.

CgiBase?
Enter the base name of your CGI directory.

DocumentRoot?
Enter the path to your HTML directory. This directory is where your static HTML pages and images will be served from.

ShareDir?
Enter the path to your shared HTML directory. This directory is where the images and files that are shared across all catalogs are kept. An example of this information would be Administration Tool graphics.

ShareUrl?
Enter the path to your shared URL directory. This directory is where URL information that is shared across all catalogs are kept.

VendRoot?
Enter the path where the Interchange directory should be located.

BaseDir?
Enter the path where the catalog directory containing your catalog(s) should be located.

InterchangeUser?
Enter the user name of the system administrator responsible for Interchange.

Server Name?
Enter the Server name on which you are running Interchange. It is recommended that you enter this value as an IP address. For example, 172.133.256.13

DemoType?
Enter the type of demo template you will be using. The default is **Foundation**. If you want to use another template you must have already downloaded it to your Interchange directory from the website.

permtype?
Enter the appropriate letter that represents your Interchange administration structure.

> *Note:* This question is asked only during the **tar** installation.

MailOrderTo?
Enter the email address that will be used to receive order information when submitted.

CatRoot?
Enter the location where the catalog configuration files and catalog working data reside.

CgiDir?
Enter the location of your cgi-bin directory. A file with your store's name is created in this directory. This folder establishes a link between your web server and the Interchange server.

CgiUrl?
Enter the location of the CGI program for this catalog. The catalog CGI program is where the pages are served from.

Aliases?
Enter an additional URL location for the CGI directory. This may be useful if you are calling different sections of your e-business from different servers.

DocumentRoot?
Enter the path to your HTML directory. This directory is where your HTML pages will be served from.

SampleHtml?
Enter the directory where sample HTML files will be installed. This should be a subdirectory of your HTML directory.

ImageDir?
Enter the directory path where your business images should be copied. This directory is where the images are served from.

ImageUrl?
Enter the URL base where sample images should be displayed from.

INET or UNIX mode?
Enter the socket type you or your ISP is using. UNIX is the most common socket type.

Do you use CGIWRAP or SUEXEC?
Enter **Yes** if you are using either of these programs. If you are unsure of the answer, enter **No**.

Restarting Interchange

After you finish building your catalog using **makecat** you must restart Interchange to implement the settings. The method used to restart interchange depends on where you installed Interchange and the permissions you have on that server. To restart Interchange:

- **From the RPM installing as root:** /usr/lib/interchange/bin/interchange -r
- **From the tarball installing as the Interchange user:** /usr/local/ interchange/bin/interchange -r
- **From the tarball installing as root:** su interch -c '/usr/local/interchange/ bin/interchange -r'
- **From the tarball installing as a regular user:** ~/interchange/bin/ interchange -r
- **From the RPM installing as root on a Red Hat system during start up:** / etc/rc.d/init.d/interchange start

After restarting Interchange, it is recommended that you view your catalog in a browser to ensure it is running properly. To view your catalog, type the following URL in a browser, substituting your server name and catalog name below:

http://url.catalogname

You now have a functioning catalog that contains the working Foundation demo web site and Administration Tool site. The next step is to set the regularly occurring maintenance tasks to make sure your systems are running smoothly. For information on how this is done, see *Maintaining Interchange, page 33*. For information on accessing the Interchange Administration Tool, see *Interchange 4.8: Administration Tool Guide*.

CHAPTER 1 **Maintaining Interchange**

For Interchange to run smoothly, certain maintenance functions need to be put in place. The following is a list of the most important maintenance functions to implement.

- Rotate the error logs. (catalog and interchange)
- Rotate directories.
- Purge the temp directory.
- Purge the session directory.

!!! This list is not intended to be all inclusive. Depending on your system and your style of business operation your maintenance functions may vary.

Rotating the Error Logs

The *error logs* are written to every time there is a problem with Interchange or with your catalog. They are important tools in helping you diagnose problems. The error information should be rotated out of the working error logs and stored in an archive. This keeps your error log from getting too large while keeping important error information for future reference.

There are two types of error logs, the *catalog specific error log* and the *Interchange specific error log*, and they should both be rotated to an archive directory, renamed, and compressed. The frequency by which you rotate your error logs depends on how often an error is written to your error log. It is recommended that you archive your error logs at least once a week.

> ***Note:*** If you are running Red Hat Linux you can also use **logrotate.d** to rotate the log files.

The Catalog Specific Error Log

The catalog specific error log reports errors that occur in a particular catalog. A typical catalog error log is located in /home/merchant/catalogs/storename/error.log.

To rotate your catalog error log you need to create a cronjob similar to the following.

```
/home/merchant/catalogs/*/error.log {
rotate 4
weekly
compress
nocreate
}
```

The Interchange Specific Error Log

The Interchange specific error log reports errors that occur with Interchange. The Interchange specific error log is located in your **interchange** directory. This directory should also contain the **interchange.cfg** file.

To rotate your Interchange error log you need to create a cronjob similar to the following:

```
/var/log/interchange {
rotate 4
weekly
compress
nocreate
}
```

Rotating Other Log Files

The following directories should be rotated:

- **tracking.asc**—a record of product orders. Tracking.asc is written to every time a product is ordered. A rotated archive of this directory serves as a back up of product orders in case of system failure.
- **usertrack**—a record of user clicks on your site. Usertrack creates a log of each user's navigation path through your web site. This directory should be rotated frequently since it grows quickly. The archived information is useful to evaluate your site layout and usability.
- **orders**—a directory of order files. The orders directory saves a file for every order number issued. A rotated archive of this directory serves as a back up of all orders received in case of system failure.

These directories can be rotated by creating a cronjob and building your own script. They are located in /var/lib/interchange/construct/logs.

> ***Note:*** If you are running Red Hat Linux you can also use **logrotate.d** to rotate the log files.

Purging the Temp Directory

The *temp* directory contains information used during daily Interchange operation. This directory is located in your **catalog** directory and is named **tmp**. This temporary information, including PGP encrypt files and search results, should be purged on a regular basis to keep your temp directory from becoming too large. The frequency with which you purge this directory depends on the amount of traffic that visits your site. It is recommended that you purge at least once a day.

To purge your temp directory, you need to create a cronjob similar to the following:

```
44 4 1 * * find /var/lib/interchange/yourcatalogname/tmp -mtime
+30 -type f | xargs rm
```

> *!!!* Be sure to set your purge time to after business hours. Purging the temp directory while in use by a customer could affect his or her shopping experience.

Purging the Session Directory

The *session* directory contains saved information from a customer visit your site. This information should be purged on a regular basis to keep your session directory from becoming too large. The frequency with which you purge this directory depends on the amount of traffic that travels to your site. It is recommended that you purge at least once a day.

To purge your session directory, you need to create a cron job similar to the following:

```
44 4 1 * * find /var/lib/interchange/yourcatalogname/session
-mtime +30 -type f | xargs rm
```

> *!!!* Be sure to set your purge time to after business hours. Purging the session directory while in use by a customer could effect his or her shopping experience.

You can also manually purge your session directory by using the **dump** utility located in your **catalog/bin** directory. The **dump** utility purges the session file for that catalog. The **expire** utility will expire all sessions for that catalog.

Finally, you must customize your catalog to meet your business needs by adding your products, customers, graphics, and text copy. This can be done using the Interchange Administration Tool. For more information on the Administration Tool, see *Red Hat Interchange 4.8: Administration Tool*.

Index