Novell NetWare® 6

www.novell.com

GETTING RESULTS WITH NETWARE
WEB SERVERS AND TOOLS

Novell®

Legal Notices

Novell, Inc.
1800 South Novell Place
Provo, UT 84606
U.S.A.

www.novell.com

Getting Results with Novell Web Services
February 2002
103-000133-001

Online Documentation: To access the online documentation for this and other Novell products, and to get updates, see www.novell.com/documentation.

Novell Trademarks

BorderManager is a trademark of Novell, Inc.

ConsoleOne is a trademark of Novell, Inc.

eDirectory is a trademark of Novell, Inc.

GroupWise is a registered trademark of Novell, Inc. in the United States and other countries.

iFolder is a trademark of Novell, Inc.

NetWare is a registered trademark of Novell, Inc. in the United States and other countries.

NetWare Loadable Module and NLM are trademarks of Novell, Inc.

NIMS is a trademark of Novell, Inc.

NMAS is a trademark of Novell, Inc.

Novell is a registered trademark of Novell, Inc. in the United States and other countries.

Novell Client is a trademark of Novell, Inc.

Novell Distributed Print Services is a trademark of Novell, Inc., and NDPS is a registered trademark of Novell, Inc. in the United States and other countries.

Remote Console is a trademark of Novell, Inc.

ZENworks is a trademark of Novell, Inc.

Third-Party Trademarks

All third-party trademarks are the property of their respective owners.

Contents

Preface

As an integral part of NetWare® 6, several Web products combine to make Novell's One Net (http://www.novell.com/news/onenet/index.html) vision a reality by providing the enabling Web technologies for many of Novell's Net services.

These key Web components include

- NetWare Enterprise Web Server
- Apache Web Server for NetWare
- Tomcat Servlet Engine for NetWare
- NetWare Web Search Server
- FTP Server
- WebDAV

What's in This Documentation?

This documentation briefly describes each of the components listed above and includes some additional information about NetWare 6 and the underlying Web infrastructure. This book includes documentation for

- NetWare Web Manager
- NetWare Enterprise Web Server
- NetWare Web Search Server
- Related concepts and procedures for using these products with NetWare 6

Documentation Conventions

In this documentation, a greater-than symbol (>) is used to separate actions within a step and items in a cross-reference path.

Also, a trademark symbol (®, ™, etc.) denotes a Novell trademark. An asterisk (*) denotes a third-party trademark.

Path References

By default, the NetWare Web Search Server is installed to the /NSEARCH directory located at the root of your server's volume. However, during installation, you can customize the location where you want Web Search installed.

Because of this, when referring to the root directory where the NetWare Web Search Server is installed, the variable name */searchroot* will be used.

Overview

This section introduces you to the Web technologies integrated with NetWare® 6 that provide the underlying Web architecture for net services software. It also shows how these products fit in to the One Net architecture, and then introduces you to each of them.

1 Introducing the Web Enabling Technologies of NetWare 6

The Web technologies included in NetWare® 6 enable Novell's Net Services products to function.

For example, the Enterprise Web Server is one of the key Web components. But it is also the enabling technology for iLogin™ and might also be used to serve up your department or company Web site.

The marriage of Novell's industry leading net services software with these Web technologies offer unparalleled opportunities for business exchanges, both within and between companies.

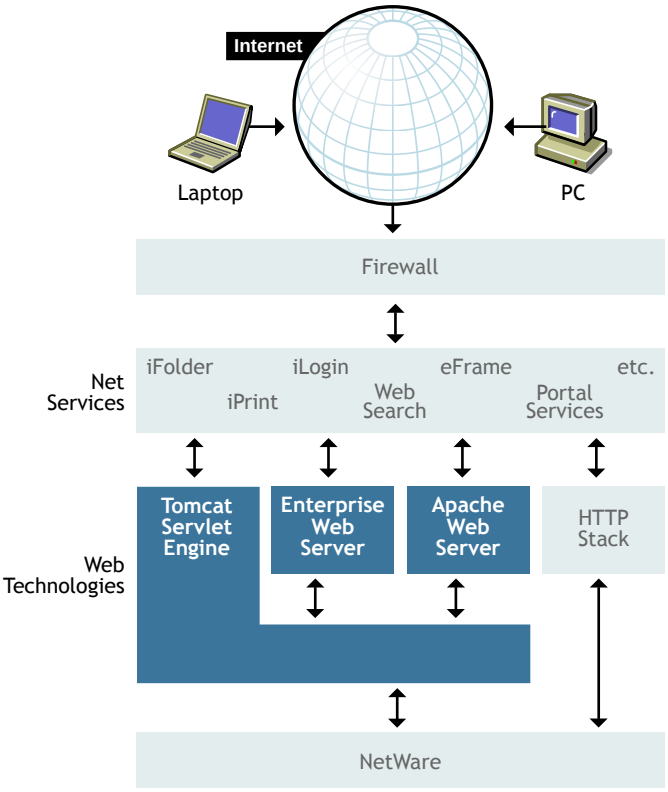
There is no better medium or infrastructure in place to enable, enhance, and encourage open communications than the Web. What e-mail has done for person-to-person communication, the Web does for person-to-department, department-to-department, company-to-person, and company-to-company communications.

While your expertise might be in networking technologies, the Web technologies are simple to use and ready to run, right out of your NetWare 6 box.

Introducing the Web Technologies

The following diagram is a simplified depiction of the role that these technologies play as the Web-enabling technologies for Net Services Software, making the One Net vision a reality.

Figure 1 **The Role of Web technologies in Enabling Net Services Software**



NetWare 6 Net Services that Depend on Web Technologies

Novell is the leading provider of Net Services Software. Net Services Software is the glue that holds together disparate networks and technologies for the purpose of simplifying business processes and communications.

As illustrated in Figure 1 on page 18, Web technologies provides the infrastructure for the following net services software included in NetWare 6:

- iFolder™
- iLogin
- iPrint
- NetWare Web Manager
- NetWare Web Search Server

- NetWare Web Access
- GroupWise Web Access
- Novell iManager

For details about each of these services, refer to their documentation on the Novell Documentation Web site (http://www.novell.com/documentation).

NetWare Enterprise Web Server

The NetWare Enterprise Web Server is an HTTP server used to serve up Web pages to the Internet, an intranet or extranet. It is optimized to run in the NetWare environment and is a critical component to building One Net solutions where all kinds of networks work together to achieve business results.

You can create a Web to enhance departmental communication, or you can create a Web that spans your location or company. You can also provide spaces for your customers, suppliers, vendors, consultants, or any other entities outside your company who would benefit by having access to specified areas on your Web. In addition, you can publish information on the Internet so that the world can see it and contribute to it.

There are traditional categories of Webs, each fulfilling a unique role. While each Web type is unique in some way, each of them facilitates a single network and the sharing of information.

- The most popular of all Webs is the World Wide Web, available through the Internet. Hosting a Web server with Internet access requires a persistent Internet connection. In this environment, information published to the Web server will be accessible by anyone with a browser. Most companies host Internet Web sites to publish information about their products or services and to provide a point of contact with customers.

  A firewall is required to provide security for Web servers with Internet connections. A firewall prevents users on the Internet from accessing a company's network-based resources.

- Many companies also host their own private intranets. The target audience of an intranet is the employees. An intranet is the ideal environment for employees and departments to publish information that can benefit others in the company.

  A company can host both Internet and intranet Web sites. Employees inside the company can access both, while people outside the company can't get through the firewall to access the intranet.

- An extranet is a combination of public and private Web sites. Extranets are usually created among companies to expedite communication and cooperation. For example, a company that relies on a vendor to fill its orders could create an extranet to allow the vendor to access customer orders. This way, the vendor could automatically fill orders without having to wait for user intervention.

For more information about using the Enterprise Server, see Chapter 4, "Putting the Web Server to Work," on page 43.

Apache Web Server for NetWare

Apache Web server is an open-source Web server originally developed by the not-for-profit Apache Group. The Apache Web server is in use by more than 60% of all Web hosting companies. It is extremely stable and it's free!

As an integral part of the Web infrastructure of NetWare 6, Apache is installed by default during your NetWare 6 installation and you should not have to manage any part of it. Apache is a critical component of NetWare's Web technologies and is used by the following NetWare 6 Web services:

- NetWare Web Manager
- NetWare Web Search Manager
- NetWare Web Search Print and Search Services
- NetWare Web Access
- iFolder
- Novell iManager

To host Web sites and Web applications, see "NetWare Enterprise Web Server" on page 19. For additional information about the Apache Web server, visit http://www.apache.org (http://www.apache.org), or refer to the Apache documentation available on your server after NetWare 6 installation and located at *volume*:\Apache\htdocs\manual.

Tomcat Servlet Engine for NetWare

Also developed by the Apache Group, Tomcat is a servlet engine used to serve up Web applications. It is also used by several NetWare 6 components, including the NetWare Web Search Server. If you are a developer working to create solutions, you will likely work a great deal with Tomcat. If you are not a developer, you may never have to do anything with Tomcat.

For additional information about Tomcat, including how to migrate IBM WebSphere applications to Tomcat, see "About Tomcat for NetWare" on page 111.

NetWare Web Search Server

Make your data searchable in minutes! From simple search solutions to complex, revenue-generating search services, NetWare Web Search bridges all types of networks, from file servers, to intranets, extranets, and the Internet, by bringing critical information to busy people. It is one of the industry's fastest and most accurate search engines available today.

Most users care less about where information is stored and more about finding it, and finding it fast! With staggering terabytes of information on the Internet and on huge information stores found in most corporations today, Web Search offers an easy way for users to find highly relevant information at record speeds.

Installed by default when you install NetWare 6, Web Search is ready to run, right out of the box. Just point Web Search at the Web or file servers to be searched, and Web Search generates keyword indexes used to perform time-saving searches.

Through the implementation of a powerfully simple template-based architecture, you can customize the search forms and search results pages to get the results you need. By using included parameters, variables, and basic HTML, Web Search lets you build your own templates from scratch. From simple search solutions to complex, revenue-generating search services, NetWare Web Search offers the power to change the way you do your business by making one net out of file servers, intranets, extranets, and the Internet.

FTP Server

The NetWare FTP Server provides FTP service for transferring files to and from NetWare volumes. FTP Server can be used to post new Web content to your Enterprise Web Server, or to post or retrieve documents from your NetWare file server.

WebDAV

Web-distributed Authoring and Versioning (WebDAV) is an industry standard protocol. It is an enhancement to the HTTP protocol, turning the Web into a document database that enables collaborative creation, editing and searching from remote locations. Whereas HTTP only supports the reading of files, WebDAV enables documents to be written using HTTP.

Because of WebDAV's versioning control, Web users can use a Web browser to write, edit and save shared documents without overwriting each others' work.

HINT: You must be using eDirectory as your directory service and Internet Explorer as your browser if you want to use WebDAV. For more information about publishing content to your Web server, see Chapter 6, "Managing Server Content," on page 63.

NetWare Web Manager

NetWare® Web Manager is the tool you use to manage all Web technologies and to access other Web-based management tools. Using any 4.x or newer Web browser such as Netscape* Navigator* or Communicator*, you can manage Novell® Web technologies and services from any place on the Internet.

With Novell® eDirectory™, you can create and manage user and group authentication to sensitive information on your server.

Requirements for Managing Web Technologies

To manage any of the Web technologies of NetWare 6, you need a Web browser, such as Netscape Navigator or Navigator Gold 3.0 or later, Netscape Communicator, Internet Explorer or any other browser that supports Java* and JavaScript*.

Web Browser Requirements

You must enable Java or JavaScript in your Web browser because all of the configuration forms in Web Manager and other management tools require one or both of these forms of Java to function.

To enable Java in Internet Explorer 4 or later:

1. From the Internet Explorer browser window, click Tools > Internet Options.
2. Select the Advanced tab.
3. Under Microsoft VM, check JIT Compiler for Virtual Machine Enabled (Requires Restart).
4. Click OK.

To enable Java in Netscape Navigator:

1. From the browser window, click Options > Network Preferences.
2. Select the Language tab and make sure Java and JavaScript are checked.
3. Click OK.

To enable Java in Netscape Communicator:

1. From the Communicator browser window, click Edit > Preferences.
2. Select the Advanced category in the left column.
3. Check the Enable Java and Enable JavaScript check boxes.
4. Click OK.

2 Introducing NetWare Web Manager

NetWare® Web Manager is a browser-based management tool used to configure and manage the NetWare Enterprise Web server. But it also serves as a front door to other NetWare browser-based management tools, such as NetWare Remote Manager. It can be likened to a Web site's home page with links to other resources and tools.

HINT: Web Manager and many other Web-based management tools used for managing NetWare 6 rely on the industry leading Apache Web server. Therefore, when viewing Web Manager access or error log files, or when shutting down or restarting Web Manager, you are actually affecting the Apache Server, not the NetWare Enterprise Web Server.

Using a workstation and Web browser, you can access Web Manager either locally (from within your WAN or LAN), or from remote locations where you have Internet access. Web Manager lets you

- Manage the Enterprise Web Server
- Monitor Web server activity
- Set up and manage user authentication and access to information on your server using Novell® eDirectory™ or local database modes
- Access other browser-based management tools such as NetWare Remote Manager or NetWare Web Search Server (see Table 1, "NetWare 6 Web-based Management Tools," on page 27)

Requirements for Using Web Manager

To use Web Manager, you must be using a 4.*x* or newer Web browser such as Internet Explorer or Netscape Communicator.

If you have a firewall, you need to configure it to allow the Web Manager port number to be made available. By default, the port number assigned is 2200. However, this is configurable during and after NetWare 6 installation. When changing port numbers, refer to Appendix F, "Port Number Assignments," on page 295.

When to Use Web Manager

There are several management tools included with NetWare 6. Some are Web or browser-based, and others require a Windows client, as with NetWare Administrator. And while you can perform basic object management tasks in eDirectory, Web Manager's primary purpose is to provide you with a tool for configuring and managing the various Web technologies.

In addition, Web Manager is a home page to other NetWare management tools, providing you with one-click access to them.

To help you decide when to use NetWare Web Manager or one of the other management tools, the following table offers a description of each tool and its intended use.

Table 1 NetWare 6 Web-based Management Tools

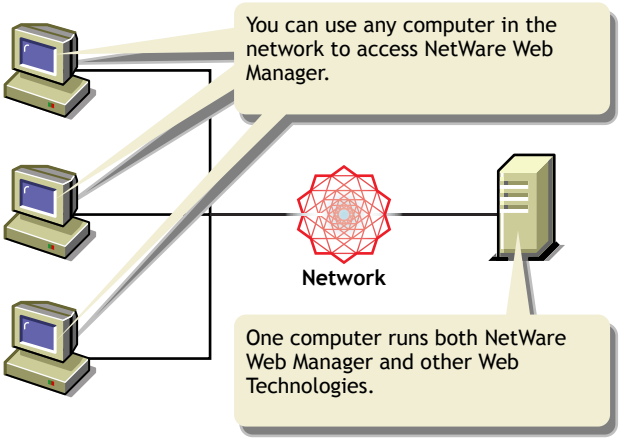
Management Tool	When to Use This Tool	How to Access This Tool
NetWare Web Manager	When you need to manage the Enterprise Web Server or modify Web Manager settings.	Enter your Web server's domain name or IP address, followed by a colon and the port number, which by default is 2200. *Example:* https://mycompany.com:2200 Or https://123.456.789.456:2200 To manage the Web server, click your servername located under NetWare Enterprise Web Server. To manage NetWare Web Manager settings, click the Admin Preferences icon in the top frame of the Web Manager home page.

Management Tool	When to Use This Tool	How to Access This Tool
NetWare Web Manager	When you need to manage eDirectory trees or objects from a remote location.	Enter your Web server's domain name or IP address, followed by a colon and the Web Manager port number, which by default is 2200. You can change the port number during and after NetWare 6 Installation. Refer to Appendix F, "Port Number Assignments," on page 295 for more information. *Example:* https://mycompany.com:2200 Or https://123.456.789.456:2200 From the Web Manager home page, click your servername located under Novell Directory Services. To manage users an groups specific to the Enterprise Server, click the servername under NetWare Enterprise Web Server, and then click the Users and Groups icon in the top frame of Web Manager.
NetWare Remote Manager	When you need to perform basic functions on your NetWare server, such as performance monitoring, restarting your server, and so forth.	Enter your Web server's domain name or IP address, followed by a colon and the port number, which by default is 8008. *Example:* https://mycompany.com:8008 Or https://123.456.789.456:8008

Management Tool	When to Use This Tool	How to Access This Tool
Novell iManager	When you need to configure or manage NDPS or DHCP.	Enter your Web server's domain name or IP address, followed by a colon and the port number. *Example:* https://mycompany.com:*port_number* Or https://123.456.789.456:*port_number*

One of the primary advantages of using NetWare Web Manager is that you can easily configure various services from a remote workstation in your network or even from a client computer outside of your firewall, provided that you have dialup access to your network.

Figure 2 Remotely Configuring a Web Technology



NetWare Web Manager also allows you to manage user authentication to your Enterprise Server using eDirectory™, local database, or LDAP modes.

NOTE: eDirectory mode handles both authentication and access rights. While you can use LDAP, we recommend that you use eDirectory. If your Web server will contain mostly public information and you have little need for authentication, you

can also use local database mode. For more information, see Chapter 7, “Using a Directory Service to Control User Access to Network Resources,” on page 85.

When you install additional Web technologies, they can be configured and managed from within NetWare Web Manager. NetWare Web Manager is installed when you install NetWare.

After installing NetWare, use a Web browser from a client computer in your network to access NetWare Web Manager. As you make changes to your services using Web Manager, modifications are made to various configuration files on your NetWare server.

HINT: To access Web Manager from outside of your firewall, you would first need to open the TCP port 443 (HTTPS) to the IP address of your Web server (typically port 80). You might already have done this to allow regular HTTP traffic to your company’s Web server. Then you would need to open the TCP port you have assigned to Web Manager. For example, of Web Manager’s port number was 5500, you would then open TCP port 5500 to the same IP address. In short, you would have set a TCP Port to 80 for HTTP traffic, another TCP port to 443 for HTTPS access, and then another port to 5500 for Web Manager access. Refer to your firewall documentation for more details.

Using Web Manager

The URL you use to navigate to Web Manager depends on the server hostname and its port number. For example, if the domain name you specified during NetWare 6 installation was MYSERVER, the URL you would use in your Web browser would be similar to

https://myserver.mycompany.com:2200

or

https://137.95.65.150:2002

HINT: The default port number for Web Manager is 2200. You can change the port number from within Web Manager by clicking Admin Preferences from the Web Manager home page and then entering a new port number. For more information about available ports, see Appendix F, “Port Number Assignments,” on page 295.

After entering the URL in your browser and sending the HTTP request to your server, Web Manager prompts you for your username and password. The administrator username and password are the same ones used when you installed NetWare 6.

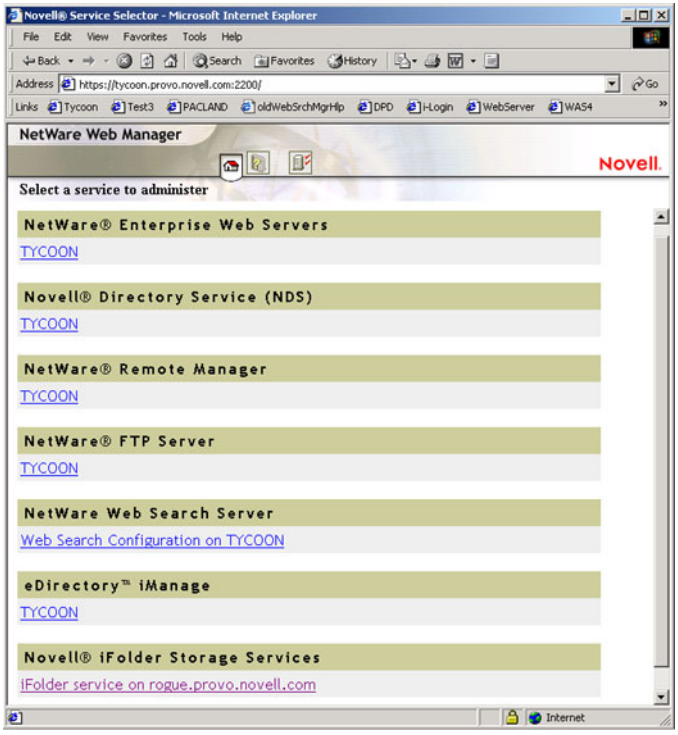
IMPORTANT: After accessing Web Manager for the first time, we recommend that you create a new user for managing the Web technologies. This will help to ensure the security of your servers from unwanted intruders.

The first page you see when you access Web Manager is called the NetWare Web Manager home page. It is similar to a home page you might see on the World Wide Web in that it is a type of front door, or portal, to information and services. The Web Manager home page links to other Web pages for the Web technologies (such as the Enterprise Web Server) that you have installed and are broken down into several service categories (see "The NetWare Web Manager Home Page" on page 31).

Depending on the Web technologies and Web-based management tools you have installed, the Web Manager home page includes the following default categories:

- NetWare Enterprise Web Server
- Novell eDirectory
- NetWare Remote Manager
- NetWare Web Search Server

Figure 3 The NetWare Web Manager Home Page



Admin Preferences: This appears as a button in the top frame of the Web Manager home page. It lets you configure settings that apply to Web Manager, such as changing it's default port number or working with error and access logs.

NetWare Enterprise Web Server: Appearing as the first category on the NetWare Web Manager home page, this category offers a link to the configuration pages for your Web server where you can manage everything from eDirectory user authentication to programs and content management.

Novell eDirectory: This link lets you perform basic eDirectory functions such as creating, changing or deleting user and group objects, and setting access rights to volumes and directories on your NetWare server. The purpose for this link is to provide a convenient Web-based access method for eDirectory in addition to using the client-based ConsoleOne™.

NetWare Remote Manager: NetWare Remote Manager lets you perform operations and monitor your NetWare server from a Web browser, anywhere where you have Internet access.

Your Web Manager might also include NetWare FTP Server, NetWare Web Search Server, and Novell iManager, if you chose to install them.

For more information about configuring Web Manager preferences, see Chapter 3, "Modifying Web Manager Preferences," on page 35.

Accessing the Web Manager Home Page

1. At the server console, type **nsweb**. The nsweb command executes an NCF file that runs the server.

   Once the NetWare Enterprise Web Server is running, you can use any browser that supports frames and JavaScript* and has access to the NetWare Web Server Manager to configure your servers.

2. Open a browser and highlight the URL

   http://*servername:port number*/

3. Substitute *servername* with the name you gave your server during installation. Substitute *port number* with the number assigned during installation.

4. When prompted, enter the username and password you chose during installation.

5. Click OK.

NOTE: The default installation modifies the AUTOEXEC.NCF to load the Web server whenever NetWare is restarted.

To disable autoloading, remove NSWEB from AUTOEXEC.NCF. To load and unload the Web server, type NSWEB and NSWEBDN respectively, at the system console.

3 Modifying Web Manager Preferences

NetWare® Web Manager has a few preference settings that you can customize, which include encryption for securing Web Manager, the Web Manager port number, and working with Web Manager access and error log files.

Securing Web Manager

Keeping intruders out of Web Manager is best accomplished through the use of encryption and server certificates. By default, encryption is enabled on your server and a server certificate is created when you install NetWare 6.

But you can take additional steps to help ensure the safety and security of your servers and data. For example, keep your administrator username and password well hidden. You might also occasionally change your Web Manager port number. Using a port number of 49152 or higher is a safe choice.

IMPORTANT: Some ports below 49152 may be in use by other services and should not be used. Using a four-digit port number between 49152 and 65000 is the safest choice. However, for a complete list of registered port numbers, visit the Internet Assigned Numbers Authority (http://www.iana.org/assignments/port-numbers).

To change NetWare Web Manager's port number, do the following:

1. From the NetWare Web Manager home page, click Admin Preferences.
2. In the Web Manager Port field, type the port number you want NetWare Web Manager to use.
3. Click OK.
4. Restart the server for the settings to take effect.

Using Encryption

Secure Sockets Layer (SSL), enabled by default when NetWare Web Manager is installed, is used to secure NetWare Web Manager by applying encryption to information going out or coming in to Web Manager. When enabled, you must use HTTPS to access Web Manager.

Once encryption is enabled, you can then use ConsoleOne to install Public Key Infrastructure Services (PKIS). When you install the Novell Certificate

Server (during the NetWare installation), a Key Material Object (KMO) was created by default. A KMO, also called a Server Certificate Object, includes a server certificate and key pair files.

To enable or disable encryption in Web Manager, do the following:

1. From the NetWare Web Manager home page, click Admin Preferences.
2. Under Encryption, click On to enable, or Off to disable SSL.
3. From the Server Certificates drop-down list, select the Server Certificate object you want to use for SSL encryption.
4. Click OK.

For more information on installing and configuring the Novell Certificate Server, refer to the NetWare 6 NetWare 6 Overview and Installation Guide or Setting Up Novell Certificate Server in the *Novell Certificate Server Administration Guide*.

Working with Log Files

The Apache Web Server logs Web Manager activity, as well as activity by other Web technologies and services that depend on the Apache Server, such as NetWare Remote Manager.

Access and error log files indicate who and what has been accessed, and what errors have occurred on your Apache Web server.

HINT: The Enterprise Web Server maintains its own log files. To view Enterprise Web Server log files, click your Web server link under NetWare Enterprise Web Server on the Web Manager home page, and then click the Server Status icon.

The access log files created by Apache are in the default Common Log Format (CLF) that provides a fixed amount of information about Apache Web Server activity.

The ERROR log file, located in ADMIN/LOGS in the server root directory, lists all the errors the server has encountered.

The ACCESS log file, located in ADMIN/LOGS in the server root directory, records information about requests to the server and the responses from the server.

To configure logging options for NetWare Web Manager, do the following:

1. From the NetWare Web Manager home page, click the Admin Preferences icon > Log Settings.
2. In the Access Log field, type a path to the directory where you want NetWare Web Manager to store the ACCESS log file.

   You can type either an absolute path or a path relative to your server root directory. Leaving this field blank deactivates access logging.
3. Click OK.

Viewing an Access Log File

You can view Web Manager's active and archived access log files under Global Enterprise Server Settings.

1. From the NetWare Web Manager home page, click the Admin Preferences icon > View Access Log.
2. In the Number of Entries field, type the number of lines you want the access log to display.
3. In the Only Show Entries With field, type the particular word you want to filter the access log entries for.

   HINT: Case is important. If you use this search feature, the Only Show Entries With field determines how many entries to search, not how many will display.
4. Click OK.

The following is a sample of an access log in the common log file format:

```
a.nov.com - [16/May/1999:21:18:26 -0800] "GET /WebAdmin/
  icons/dot.gif HTTP/1.0" 200 2575

a.nov.com - [17/May/1999:11:04:38 -0800] "GET /WebAdmin/bin/
  frames?index+pref HTTP/1.0" 204 342

a.nov.com - [20/May/1999:14:36:53 -0800] "GET /WebAdmin/
  manual/ag/config.htm HTTP/1.0" 200 890

arrow.a.com -[20/May/1997:14:36:53 -0800] "GET /WebAdmin/
  manual/ag/so.gif HTTP/1.0" 401 571
```

Table 2 Descriptions of Each Field in an Access Log File

ACCESS Log Field	Example
Hostname or IP address of client	user.novell.com In this case, the hostname is shown because the server is using DNS lookups; if DNS cannot resolve the name or if DNS lookups are disabled, the client's IP address would appear.
Username	john (username entered by the client for authentication)
Date/time of request	29/Mar/1998:4:36:53 -0800
Request	GET /WebAdmin/https/ReadAccessLog.jsp
Protocol	HTTP/1.0
Status code	401
Bytes transferred	571

Viewing the Error Log File

The ERROR log file contains errors the server has encountered after the log file was created. It also contains informational messages about the server, such as when the server was started and who tried unsuccessfully to login to the server.

1. From the NetWare Web Manager home page, click Global Enterprise Server Settings > View Error Log.
2. In the Number of Entries field, type the number of lines you want to see.
3. In the Only Show Entries With field, type the particular word that you want to filter the error messages for.

   HINT: As with the Access Log file, case is important. If you use this search feature, the Only Show Entries With field determines how many entries to search, not how many will display.

   This field is case sensitive.
4. Click OK.

The following is an example of an error log:

```
[13/May/1999:16:56:51] info: successful server startup

[13/May/1999:16:56:51] info: NetWare Web-Administrator
  97.117.0455

[13/Mar/1999:19:08:52] security: for host user.mozilla.com
  trying to GET /admin-serv/bin/index, acl-state reports:
  access of /usr/suitespot/bin/admin/admin/bin/index denied
  by ACL admin-serv directive 3

[13/May/1999 20:05:43] failure: for host ceo.mozilla.com
  trying to POST /admin-serv/bin/distadm, cgi-parse-output
  reports: the CGI program /usr/suitespot/bin/admin/admin/
  bin/distadm did not produce a valid header (program
  terminated without a valid CGI header. Check for core dump
  or other abnormal termination)
```

Working with the NetWare Enterprise Web Server

A Web server is a fundamental building block for bridging disparate networks together into one net. With the NetWare® Enterprise Web Server, you can host Internet, intranet, or password-protected extranets that serve as secure portals to your company's business processes and information.

The Enterprise Web Server is a key component in building true, one net, eBusiness solutions. It is a powerful NetWare-ready HTTP server that is ready to run right out of the box.

You can use the Enterprise Web Server to host all types of Web content, from simple Web sites containing static HTML files to complex, dynamically generated Web pages and Web applications that can automate your business processes and improve communications.

Among other things, the NetWare Enterprise Web Server lets you

- Deploy a departmental intranet server using the existing NetWare network backbone
- Publish existing data stored on a NetWare network for consumption by clients with a browser over the WAN
- Use the NetWare server with Microsoft* Office 2000 productivity applications without the use of client software (other than your Web browser)
- Establish file access control using eDirectory™ and SSL-based security
- Allow users to quickly and easily create and manage their own home pages
- Manage security of sensitive folders and files

- Run server-side applications using Java servlets, CGI, scripting, Java Server Page and Active Server Page technologies

The information in this section will help you become familiar with how to manage Web server settings, how to publish content to it, and how to add your own Web applications.

4 Putting the Web Server to Work

NetWare® Enterprise Web Server provides an important ingredient in bridging disparate networks. The Web Server is installed by default; but if you chose not to install it and you want to build One Net solutions, you will need to install it using NWCONFIG or the NetWare server GUI. For NetWare 6 installation information, see the *NetWare 6 Overview and Installation Guide*.

The Enterprise Web Server provides all the functionality necessary for companies, both small and large, to build Web solutions. It offers security and authentication using directory services.

The Enterprise Web Server also serves as a host for the Tomcat servlet engine, which is a portable environment for deploying Java applications on NetWare.

IMPORTANT: Tomcat replaces IBM's WebSphere Application Server that was included with NetWare 5.1. If you have applications deployed using WebSphere on NetWare, you can use the migration utility included with NetWare 6 to move your applications to Tomcat. For more information, see "About Tomcat for NetWare" on page 111.

The Enterprise Web Server allows Web developers to select development tools with which to create Web-based applications on the NetWare platform.

Creating Your Own Web Site

You can use any HTML editor to create a Web site, although most functional corporate Web sites are created by professional designers. But depending on your needs and resources, your implementation tool can range from any of the readily available Web site creation programs (some of which are free) to a team of programmers. Another avenue is to out-source the creation of your Web site.

Creating personal and departmental Web sites can be simple, requiring only minutes to assemble. You can use any HTML editor to create the pages of your Web site.

When you create your home page, save the file as INDEX.HTM or .HTML and that file will automatically appear when your Web site is accessed. You can then create links to other pages and graphics with any filenames.

HINT: You can configure the Enterprise Server to recognize a specific filename and extension so that when a user enters your Web server's URL, it will automatically display your home page. See "Setting the Primary Document Directory" on page 63.

Hosting Multiple Web Servers

You can configure your NetWare 6 server to host multiple Web servers. This way, a single NetWare 6 server running NetWare Enterprise Web Server can host all the Web server needs of your company; or, if you are an Internet Service Provider (ISP), you can host Web sites for your customers. This makes it easy to allow two or more departments to create their own Web sites without requiring that they each have a server.

You can host two types of Web servers on your NetWare server:

- Hardware virtual servers
- Software virtual servers

Each type has its strengths and weaknesses; you should choose the one that's right for your situation.

For information about setting up hardware virtual servers, see "Setting Up Hardware Virtual Servers" on page 74. For information about setting up software virtual servers, see "Setting Up Software Virtual Servers" on page 75.

Accessing Your Web Site

If you have already successfully installed NetWare 6 and the Enterprise Web Server was included in the list of products to install, you can access it right now. A sample Web page and some sub-pages have been included. You can remove these pages and replace them with your own content.

HINT: Before replacing the sample Web site, you might want to look through it first. It is a good place to start for an introduction to all of the Web-based management tools included with NetWare 6 and includes links to the actual management tools.

To view the sample Web site, open a client Web browser on a workstation in your network and enter your NetWare server's IP address or DNS name. For example:

http://*server_IP_address*

or

http://*domain_name*

Adding Content to Your Web Site

NetWare Enterprise Web Server has a document root or primary document directory. By default, the path to the primary document root directory is SYS:\\NOVONYX\SUITESPOT\DOCS. This is where all of the content for the sample Web site is stored.

All content placed in this folder is visible to your Web site audience. If necessary, you can easily specify another directory as the primary document root directory. (See "Setting the Primary Document Directory" on page 63.)

Once your Web server is running, you can start posting content for the world (or your department or company) to see by placing files in the Web server's primary or additional document directories.

For example, suppose you created a new HTML file called WELCOME.HTM that included some information about your department that you wanted to share with other departments in your company. You would then copy the file to SYS:\\NOVONYX\SUITESPOT\DOCS, which is the Web server's default primary document directory.

If your server's domain name were SALES.MYCOMPANY.COM, people in your company would enter the following URL to view the new document.

HTTP://SALES.MYCOMPANY.COM/WELCOME.HTM

You could also then edit the document from the document root directory and the moment you save your changes, users would see the changes when they refresh their Web browser or when they view the page for the first time.

As mentioned above, a number of files are stored in the \DOCS root directory. You should replace these files with your own files. The page you choose as your home page should be named INDEX.HTML. By default, the Enterprise Server recognizes this name and will serve it up automatically when a user points at your server's domain name. However, you can specify an alternate

filename as the default index page. (See “Specifying a Default Home Page” on page 69.)

You can also create additional document directories, which is a good idea if departments want to publish their own content to the company Web site but when you don’t want to give users control of the primary document root. (See “Setting Additional Document Directories” on page 64.)

You can follow the same procedures when creating a company Internet site, intranet site, departmental site, or even a personal site. What differentiates each of Web site is whether the Web site is placed on the Internet outside the firewall or on the intranet inside the firewall. Departmental Web sites are typically a software virtual server where personal Web sites are easily created by each user.

How to Publish Content to Your Web Server

A Web site on the Internet is typically the place for you to publish information you want visitors to read. However, a Web site on an intranet is most effective if employees can participate and share information with others. This makes it possible for users to communicate within a department, for departments to share information with other departments, and for company leaders to communicate with the entire company.

Web content contributors have three options for publishing content to your Web server:

- Mapping a network drive and creating or copying the content to the desired directory
- Using Internet Explorer 5.0 or higher
- Using Novell® NetDrive to map a drive using FTP or WebDAV

In each case, the administrator must first do something to allow the user to access directories on your server.

Publish Content Using a Mapped Drive

1. Use ConsoleOne™ to set up access rights to your NetWare server for each person who will be contributing content to your Web server.
2. Provide users with the correct network path to your server and to the folders with which they will be working.

HINT: If users want to map a drive without having to install or use the NetWare client, they can use NetDrive, which is included on the root of your *NetWare 6*

Client CD. (See Installing Novell NetDrive in the *Novell NetDrive Administration Guide*.) Once NetDrive is installed, you can map a drive across the Internet to folders on your NetWare server. You must have WebDAV enabled. See "Web Publishing through WebDAV" on page 68.

Publishing Content Using Internet Explorer

1. Make sure WebDAV is enabled on your Web server. (See "Web Publishing through WebDAV" on page 68.)
2. On a client computer, open Internet Explorer.
3. In the Address field, enter your server's domain name, followed by "My Network". For example:

   `https://digital.airlines.com/My Network`

   You can also use your server's IP address. For example:

   `https://157.168.179.200/My Network`
4. Press Enter and, when prompted, enter your directory service username and password.

To publish content using NetDrive, refer to the *Novell NetDrive Administration Guide*.

Creating Personal Web Sites

Users on your network can also create their own personal Web sites. This requires no administrative interaction other than making sure you specified a home directory for users who want to publish their own Web content. If you have not created home directories, you can easily go into each user directory and add one using ConsoleOne.

1. Create a PUBLIC_HTML directory in your personal directory.
2. Copy or create a Web page and place the INDEX.HTML file in this directory.
3. Access the Web site by entering the following URL in your browser's Address field:

   `http://servername/~username`

   or

   `http://IPaddress/~username`

Securing Your Web Site

Because information published on a Web site can be viewed by anyone, sensitive information should be guarded. Most Web sites on the Internet are designed for general access, but a company intranet is an ideal environment for Web site security. Likewise, extranets also demand tight security controls.

Using a directory service, such as eDirectory™, you can control access to the entire server or to parts of the server, such as directories, files, or file types.

NetWare Enterprise Web Server is configured, by default, to run in eDirectory mode, but you can modify it to run in either local database or LDAP modes. With eDirectory, you manage access control through the NetWare file system trustees. When running in local database mode, usernames and passwords are stored in a simple configuration file and, therefore, are not as secure as using eDirectory. Running in LDAP mode requires that you have an LDAP server running and configured in your network.

Controlling Access Using eDirectory Mode

Running in Novell eDirectory mode allows you to restrict access to folders on your server.

eDirectory mode allows you to restrict access to files, but it does not allow you to restrict access based on IP address or other criteria. If access must be restricted based on IP address or other parameters, you must either change modes and use LDAP or find an alternative method to restrict access, such as a firewall.

Managing users and groups while running in eDirectory mode is best accomplished using ConsoleOne. However, you can perform basic eDirectory tasks from the Users and Groups section of the Enterprise Server pages of Web Manager, or from the eDirectory link on the Web Manager home page.

Access control is accomplished by restricting access to files in the same way that you control access to files stored on the file server. For more information, see Chapter 7, “Using a Directory Service to Control User Access to Network Resources,” on page 85.

Additional Web-Based Services

NetWare 6 goes beyond simply providing a Web server and includes all the functionality for hosting a complete Web environment, including NetWare FTP server and NetWare Web Search Server.

Using the NetWare FTP Server

The NetWare FTP Server provides File Transfer Protocol (FTP) service for transferring files to and from NetWare volumes. You can use the FTP command from a workstation with FTP access to log in to an eDirectory tree. You can also perform file transfers from any FTP client by using the FTP Server to log in to the eDirectory tree. After logging in to an eDirectory tree, you can navigate to other NetWare servers (in the same eDirectory tree) that might not be running FTP service.

FTP servers provide fast Internet download capabilities and are known most for their ability to download files that are linked to a Web-based URL.

Another common use for FTP is using its Internet access to provide collaboration between organizations or companies. For example, Company A could use FTP to provide a shared rights area with Company B, while maintaining rights via IP addresses. A set of addresses would be assigned to a given area and only those addresses would have access to that area. This goes one step further than user authentication, since you can restrict access on a user's location as well.

The most common use for FTP today is the automated archival ability. Its strengths are that it is fast and lightweight, and that archives can be scripted by many FTP clients. The main purpose is to move data from the live environment to an archive environment where it is less expensive to store. Types of information being archived range from Human Resources information to product code backups. With the emergence and growth of the Web environment, FTP has been the main medium for backing up large Web contents or moving content from a staging server to a production environment.

Information on using FTP to transfer files can be found in the Overview section of the *NetWare FTP Server Administration Guide*.

HINT: The FTP server can be loaded from the NetWare console using the command **nwftpd**. Users can start the FTP session from a workstation running FTP client software using the command **ftp** *server_name*.

Using the NetWare Web Search Server

With the NetWare Web Search Server you can create an enterprise-wide index of all the information on your intranet or NetWare file servers. You can organize information spanning multiple servers and file types into a knowledge base. Using a single interface accessible from any Web browser, users can access online information whether it exists in HTML, Word*, Excel*, WordPerfect*, or several other file formats.

For more information, see Chapter 11, “Introducing NetWare Web Search Server,” on page 133.

5 Managing the Web Server

This chapter describes how to configure NetWare® Enterprise Web Server preferences.

Starting and Stopping the Web Server

Once installed, the Web server runs constantly, listening for and accepting requests. You can start and stop the server using Web Manager, NetWare Remote Manager, or the NetWare system console.

1. From the Web Manager home page, click Enterprise Web Server *servername*.
2. Click Server On or Server Off.

After you shut down the server, it might take a few seconds for the server to complete its shutdown process and for the status to change to Off.

If your NetWare is taken offline for any reason, the Web server stops and any requests it was servicing will be lost.

Setting the Termination Time-out

When you stop the Web server, it stops accepting new connections. Then it waits for all outstanding connections to complete. The time the server waits before timing out is configured in the MAGNUS.CONF file. By default it is set to 3 seconds. You probably do not need to change this value. If you do need to change the value, add **`TerminateTimeout`** ***`seconds`*** to MAGNUS.CONF, where *seconds* represents the number of seconds you want the server to wait before timing out.

The advantage to configuring this value is that you can wait longer for connections to complete. However, because most servers have connections open from nonresponsive clients, if you increase the time the server waits, you will almost always have to wait the full time before your server shuts down.

Viewing Server Settings

From View Server Preferences, you can view your server's technical and content settings and see if your server is running. The technical settings come from MAGNUS.CONF and the content settings come from OBJ.CONF. These files are located in the server root, in the directory HTTP-*servername* CONFIG. For more information about the MAGNUS.CONF and OBJ.CONF files, see the Novell Developer Kit Web site (http://www.developer.novell.com/ndk/nscomp.htm).

The following explains the server's technical settings:

- Server Root: The directory where the server binaries are kept. You first specified this directory during installation.
- Hostname: The URL that clients use as a hostname to access your server.
- Port: The port on your system that the server monitors for HTTP requests.
- Error Log: The name and path of the server's error log file.
- MTA Host: The name of the mail server (used by agents).
- NNTP Host: The name of the news server (used by agents).
- DNS: Indicates whether DNS is enabled or disabled.
- Security: Indicates whether SSL is enabled or disabled.
- Asynch DNS: Indicates whether asynchronous DNS is enabled or disabled.

The server's content settings depend on its configuration. Common server content settings include the server's document directory, its index filenames, name and location of its access log, and default MIME type.

Restoring Backup Configuration Files

You can view or restore a backup copy of your configuration files, which include the following:

- HTTPS-SERVER_ID.ACL
- MAGNUS.CONF
- OBJ.CONF
- WEBPUB.CONF
- AGENT.CONF
- MIME.TYPES
- ACL files
- RDM.CONF
- CSID.CONF
- PROCESS.CONF
- ROBOT.CONF
- FILTER.CONF

To Restore (and View) a Backup Copy of Configuration Files

1. From the Web Manager home page, click Enterprise Web Server *servername* > Server Preferences > Restore Configuration.
2. In the Set Number of Sets of Backups field, enter the number of backups displayed on the form and click Change.
3. To restore a backup version, click Restore.

   To restore all files to their states at a particular time, click Restore to Date, which lists the specific time to which you want to restore.
4. Click OK > Save and Apply.
5. To view a backup version, click View next to that version.

Tuning Web Server Performance

You can configure the server's technical options, including the number of maximum simultaneous requests, listen queue size, and DNS usage.

To get the number of simultaneous requests, the server counts the number of active requests, adding 1 to the number when a new request arrives and subtracting 1 when a request is finished. When a new request arrives, the server checks to see if it is already processing the maximum number of requests. If it has reached the limit, it defers processing new requests until the number of active requests drops below the maximum amount.

Configuring Maximum Simultaneous Requests

You can set the number of maximum simultaneous requests, which is the number of active requests allowed for the server at one time. If your site is processing many requests that take many seconds, you might need to increase the number of maximum simultaneous requests. However, for general Internet or intranet use, you probably will not need to change the default value (128 requests).

If you need to change the number of maximum simultaneous requests, set the number before starting the server.

1. From the Web Manager home page, click Enterprise Web Server *servername* > Server Preferences > Performance Tuning.
2. In the Maximum Simultaneous Requests field, enter the number of requests.
3. Click OK > Save and Apply.

Enabling Domain Name System Lookups

You can configure the server to use Domain Name System (DNS) lookups during normal operation. By default, DNS is not enabled; if you enable DNS, the server looks up the hostname for a system's IP address. Although DNS lookups can be useful for server administrators when looking at logs, they can affect performance. When the server receives a request from a client, the clients's IP address is included in the request. If DNS is enabled, the server must look up the hostname for the IP address of each client that makes a request.

IMPORTANT: If you turn off DNS lookups on your server, hostname restrictions won't work and hostnames won't appear in your log files. Instead, you'll see the IP addresses.

You can also specify whether to cache the DNS entries. If you enable the DNS cache, the server can store hostname information after receiving it. In the future, if the server needs information about the client, the information is cached and available without further queries. You can specify the size of the DNS cache and an expiration time for DNS cache entries. The DNS cache can contain from 32 to 32768 entries; the default value is 1024 entries. Values for the time it takes for a cache entry to expire can range from 1 second to 1 year (specified in seconds); the default value is 1200 seconds (20 minutes).

To modify DNS settings, do the following:

1. From the Web Manager home page, click Enterprise Web Server *servername* > Server Preferences > Performance Tuning.
2. Select No or Yes to enable DNS.
3. Select No or Yes to enable Async DNS.
4. Select No or Yes to cache DNS entries.
5. (Conditional) If you cache DNS entries, enter the number of entries that you want cached in the Size of DNS Cache field. In the Expire Entries field, enter the number of seconds at which a cache entry will be deleted.
6. Click OK > Save and Apply.

Configuring Listen Queue Size

The listen queue size is a socket-level parameter that specifies the number of incoming connections the system will accept for that socket. The default setting is Incoming Connections.

Normally, you should not change the listen queue size. The default setting is sufficient in most cases.

If you manage a heavily used Web site, you should make sure your system's listen queue size is large enough to accommodate the listen queue size setting from the Server Preferences form. If you do change the listen queue size, make sure that your system supports the new size. The listen queue size set from the Server Preferences form changes the listen queue size requested by the server. If the server requests a listen queue size larger than the system's maximum listen queue size, the size defaults to the system's maximum.

IMPORTANT: Setting the listen queue size too high can degrade server performance. The listen queue size was designed to prevent the server from becoming overloaded with connections it cannot handle. If your server is overloaded and you increase the listen queue size, the server will only fall further behind.

To modify the listen queue size, do the following:

1. From the Web Manager home page, click Enterprise Web Server *servername* > Server Preferences > Performance Tuning.
2. In the Listen Queue Size field, enter the listen queue size you want.
3. Click OK > Save and Apply.

Configuring the HTTP Persistent Connection Time-out

With HTTP 1.1, a connection can be set to be persistent (similar to Keep Alive in HTTP 1.0). However, even if a connection is persistent, it still needs to have a time-out setting or it might consume system resources.

Normally, you should not change the persistent connection time-out. The default setting is sufficient in most cases.

1. From the Web Manager home page, click Enterprise Web Server *servername* > Server Preferences > Performance Tuning.
2. In the HTTP Persistent Connection Time-out field, enter a number (representing seconds).
3. Click OK > Save and Apply.

Configuring MIME Types

Multipurpose Internet Mail Extension (MIME) types control what types of multimedia files your e-mail system supports. You can also use MIME types to specify what file extensions belong to certain server file types (for example, to designate what files are CGI programs).

Adding a New Mime Type

1. From the Web Manager home page, click Enterprise Web Server *servername* > Server Preferences > MIME Types.
2. From the Category drop-down list, select a category.

Type is the file or application type, *Enc* is the encoding used for compression, and *Lang* is the language encoding.

3. In the Content-Type field, enter the context type that will appear in the HTTP header.

   The receiving client uses the header string to determine how to handle the file. The standard strings are listed in RFC 1521.

4. In the File Suffix field, enter the file suffix.

   This is the file extension that maps to the MIME type. To specify more than one extension, separate the entries with a comma and do not include any spaces. Do not map one file extension to two MIME types.

5. Click New Type.

Editing a Mime Type

1. From the Web Manager home page, click Enterprise Web Server *servername* > Server Preferences > MIME Types.
2. Click Edit next to the category you want to edit.
3. In the Content-Type field, enter the context type.
4. In the File Suffix field, enter the file suffix.
5. Click Change MIME Type > Save and Apply.

IMPORTANT: Do not enter spaces between the file suffixes when you add or edit a MIME type. If you put a space between them, you might receive an error or your server might not restart. If this happens, edit your MIME.TYPES file to delete the space. The MIME.TYPES file is in your server root in the HTTPS-*servername*/CONFIG directory. After you have edited the file, from Server Preferences, click Apply.

Removing a Mime Type

1. From the Web Manager home page, click Enterprise Web Server *servername* > Server Preferences > MIME Types.
2. Click Remove next to the category you want to remove.
3. Click Save and Apply.

Configuring Network Settings

You can change your server's network settings by using the Server Preferences page. Following is a brief introduction to each setting and instructions to modify your network settings.

Changing the Server Name

The server name is the full hostname for your server. When clients access your server, they use this name. The format for the server name is *servername.yourdomain.domain*.

For example, if your full domain name is novell.com, you could install a server with the name www.novell.com.

If you have set up a DNS alias for your server, use that alias.

Changing the Server Port Number

The server port number specifies the TCP port that the server listens to. The port number you choose can affect your users. If you use a nonstandard port, then anyone accessing your server must specify a server name and port number in the URL. For example, if you use port 8090, users wanting to access your server from their Web browsers would specify the following:

```
http://www.novell.com:8090
```

The standard unsecure Web server port number is 80; the standard secure Web server port number is 443. Technically, the port number can be any port from 80 to 65535, but because many other services are using other ports, you should carefully choose your ports.

IMPORTANT: Before changing the Web server's port number, refer to Appendix F, "Port Number Assignments," on page 295.

Changing the Server Binding Address

At times you'll want the server to answer to two URLs. Your system must already be set up to listen to multiple IP addresses. For information on configuring multiple IP addresses, refer to "Setting Up Hardware Virtual Servers" on page 74.

Changing the Server's MTA Host

The server's Message Transfer Agent (MTA) host is the name of the Simple Mail Transfer Protocol (SMTP) mail server. See the following section, "Modifying Network Settings," for instructions on where to modify your server's MTA host.

Modifying Network Settings

1. From the Web Manager home page, click Enterprise Web Server *servername* > Server Preferences > Network Settings.
2. In the Server Name field, enter the full hostname or DNS alias of your server.
3. In the Server Port field, enter the port number of your server.
4. In the Bind to Address field, enter the IP address that is associated with the specified hostname.
5. In the MTA Host field, enter the name of your SMTP mail server.

   You must enter a valid MTA if you want to use the agent e-mail function.
6. Click OK > Save and Apply.

Customizing Error Responses

You can specify a custom error response that sends a detailed message to clients when they encounter errors from your server. You can specify a file to send or a CGI program to run.

Instead of sending back the default file, you might want to send a custom error response. For example, if a client repeatedly tries to connect to a part of your server protected by access control, you might return an error file with information on obtaining an account.

What Are the Errors?

You can customize the response to the following kinds of errors:

- Unauthorized: Occurs when users without access permission try to access a document on the server that is protected by access control.
- Forbidden: Occurs when the server doesn't have file system permissions to read something, or if the server is not permitted to follow symbolic links.

- Not Found: Occurs when the server can't find a document or when it has been instructed to deny the existence of a document.
- Server Error: Occurs when the server is not configured properly or when a catastrophic error occurs, such as the system running out of memory or producing a core dump.

Setting Up the Response

Before you can set up the response, you need to write the HTML file to send or create the CGI program to run.

1. From the Web Manager home page, click Enterprise Web Server *servername* > Server Preferences > Error Responses.
2. From the Editing drop-down list, select the server resource you want to configure.
3. To choose a specific part of your server, click Browse.
4. To browse files and directories on your server, click Options.
5. To return to the Custom Error Responses form, click Back.
6. To enter the wildcard pattern to edit, click Wildcard.
7. Select the error response you want to customize.
8. In the appropriate field, enter the absolute pathname to the file or CGI script that you want to return for that error code.
9. Check the CGI box if the file is a CGI program that you want to run.
10. Repeat this process for each error response you want to customize.
11. Click OK.

To remove a customization, return to the form and delete the filename from the field next to the error code.

Restricting Access

Use the Restrict Access form to configure several features.

When you use Public Directory Designation, you're actually specifying what files and directories you want to allow public access to. The Public Directory Designation box lists directories and files that are currently public with associated prefixes.

For more information on working with additional or virtual document directories, refer to "Setting Additional Document Directories" on page 64 and "Setting Virtual Document Directories" on page 65. For information on User Document Directories, refer to "Configuring User Document Directories" on page 66.

The Password Redirection File allows you to create and display a file that alerts users their passwords have expired and that they are using grace logins. When users access this URL, the specified file appears rather than INDEX.HTML.

When you are in eDirectory™ mode, file access is determined by eDirectory rights granted to users. Rights Checking Mode allows you to check rights at a more granular level. If you have the system check rights at the file level, system performance will be affected.

Making a File or Directory Public

1. From the Web Manager home page, click Enterprise Web Server *servername* > Server Preferences > Restrict Access.
2. Click Insert File to insert a file or directory.
3. Click Save to save your changes.

Displaying a File Indicating Expired Passwords

1. In the Password Expiration Redirection File field, enter the path to the location where you have saved (or will) your password expiration notification file.

   The default is \NOVONYX\SUITESPOT\DOCS\NDSDIRECT.HTML.
2. Click Save to save your changes.

If you are using eDirectory, you can use Rights Checking Mode to determine at what level you want rights checked. Checking File will affect performance.

Establishing Security

When you install the Novell® Certificate Server (during the NetWare installation), a Key Material Object (KMO) is created by default. A KMO, also called a Server Certificate object, includes a server certificate and key pair files.

For related information on securing the NetWare Web Manager, refer to Chapter 3, "Modifying Web Manager Preferences," on page 35.

For more information on installing and configuring the Novell Certificate Server, refer to the *Novell Certificate Server Administration Guide* (http://www.novell.com/documentation/lg/crtsrvad/docui/index.html).

If you created a KMO, you can use the following procedure to enable security:

1. From the Web Manager home page, click Enterprise Web Server *servername* > Server Preferences > Encryption On/Off.

   This option is available only if you have created a KMO.

2. Under Encryption, click On.

   The Port Number field displays 443.

3. Select the KMO you want to use for encryption from the KMO drop-down list.

4. Click OK.

To affect the changes, restart the Web server by going to the Server Preferences page and clicking Server Off and then Server On.

HINT: Once you have enabled security on the NetWare Web Manager or Enterprise Web Server, you must use https:// in the URL to access them.

6 Managing Server Content

You can use the NetWare® Web Manager to help manage Web server content. You can create HTML pages and other files such as graphics or text and then store those files on your server. When users connect to your server, they can view your files provided they have access to them.

This chapter describes how your users can contribute content to your Web server and how you can configure and manage content files and folders.

Setting the Primary Document Directory

You probably don't want to make all the files on your file system available to remote clients. An easy way to restrict access is to keep all of your server's documents in a central location, known as the document root or primary document directory.

Another benefit of the document directory is that you can move your documents to a new directory (perhaps on a different disk) without changing any of your URLs, because the paths specified in the URLs are relative to the primary document directory.

For example, if your document directory is SYS:NOVONYX\SUITESPOT\DOCS, a request such as http://www.novell.com/products/info.html tells the server to look for the file INFO.HTML in SYS:NOVONYX\SUITESPOT\DOCS\PRODUCTS\.

If you change the document root (by moving all the files and subdirectories), you only have to change the document root that the server uses, instead of mapping all URLs to the new directory or telling clients to look in the new directory.

To set your server's primary document directory, do the following:

1. From the Web Manager home page, click Enterprise Web Server *servername* > Content Management > Primary Document Directory.
2. In the Primary directory field, enter the full pathname of the directory that you want to make the primary document directory.
3. Click OK > Save and Apply.

Setting Additional Document Directories

Most of the time you keep all of your documents in the primary document directory. But sometimes you might want to serve documents from a directory outside of your document root. You can do this by setting additional document directories. By serving from a directory outside of your document root, you can let someone manage a group of documents without giving them access to your primary document root.

1. From the Web Manager home page, click Enterprise Web Server *servername* > Content Management > Additional Document Directories.
2. In the URL Prefix field, enter the URL prefix or keyword you want to use to represent the path.

   For example, the URL prefix could be *docs*.
3. In the Map to Directory field, enter the absolute path of the directory you want the URL prefix to map to.

   The command syntax is *volume*:*directory**subdirectory*.

   For example, the path could be

   \NOVONYX\MARKETING\PUBDOCS\INDEX.HTML
4. (Optional) Select a configuration style to apply to this directory's configuration.
5. Click OK.

HINT: When you update information but don't save and apply changes, your information is retained so that you can view and edit it, even though the changes have not taken effect.

Setting Virtual Document Directories

A virtual document directory allows you to serve documents from directories that do not reside on the file server where your Enterprise Web Server is running but that do exist in the same tree as the server where your Web Server is running.

IMPORTANT: If your eDirectory™ User objects for users who need access to their Virtual Document Directories are in a different container than the eDirectory Server object (for the NetWare server running the Enterprise Web Server), the containers that include the User objects with Virtual Document Directories must be in a partition that is replicated (master or read/write) on the server that the Enterprise Web Server is running on.

Establishing the Path to the Directory

1. From the Web Manager home page, click Enterprise Web Server *servername* > Content Management > Additional Document Directories.
2. In the URL Prefix field, enter a key word (for example *text*) to represent the path to the virtual directory.
3. In the Map to Directory field, specify the path to your documents in the following format:

   *servername**volume*:*directory**subdirectory*

Providing Public Access

To provide public access to the virtual directory in eDirectory mode and restart the server:

1. From the Web Manager home page, click Enterprise Web Server *servername* > Server Preferences > Restrict Access.
2. Click Insert File and enter the path (or any portion of the path you want to be public) using the following syntax:

   *servername**volume*:*directory**subdirectory*

3. Click OK > Save Changes.
4. To effect the changes, click On/Off under Server Preferences to restart the server.

Setting Server Access

To give the file server that is running the Enterprise Web Server access to the directory structure of the server where the index file resides, the Enterprise Web Server file server must be configured as a trustee of that directory. Use ConsoleOne™ to set the rights.

Configuring User Document Directories

User Document Directories allows you to set up document directories or home directories for each user in your directory. A great advantage to setting up home directories is that users can then access their own files using a Web browser.

For every user that you want to provide a home page for, complete the following tasks:

- Create a home directory for each User object.
- Create a PUBLIC_HTML directory in the user's home directory and copy an INDEX.HTML file to it.
- Add the user's context to the Search Contexts List.
- Restart the NetWare Web Manager.
- Activate User Document Directories in the Enterprise Web Server.
- Make the PUBLIC_HTML directory public.

See the following sections for details on completing the above tasks.

HINT: If you are running the Enterprise Server in eDirectory mode and you have user objects for users who need access to their document directories, the user objects should be kept in the same container as the eDirectory server object.

If they are not, then the containers that include these user objects must be in a partition that is replicated (Master or Read/Write) on the server where the Enterprise Server is running.

Creating a Home Directory

Using ConsoleOne, create new users in their appropriate contexts. Click Create Home Directory in the lower portion of the form to create their user document directories.

Creating a PUBLIC_HTML Directory

Create PUBLIC_HTML directories in the users' home directories and copy INDEX.HTML files to them.

NOTE: You can change the name of the PUBLIC_HTML directory. Should you choose to change it, make sure all references to this directory name are consistent.

Adding Users' Contexts to the Search Contexts List

1. From the Web Manager home page, click Enterprise Web Server *servername* > Users and Groups.
2. Click Insert Context and enter the information for each new context in the New NDS® Context box. Use the following format:

   ou=*yourdepartment*.o=*yourcompany*

   This information is added to the Search Contexts List.

   If this context is already set in your AUTOEXEC.NCF file (set Bindery Context=) you don't need to add it here.
3. Click Save Changes.

Restarting the Enterprise Web Server

Restart the server at the system console. Use the command NSWEBDN to bring down the Web server and NSWEB to restart it.

Activating User Document Directories

This step activates your users' home directories so that when the URL is entered all that is required is a slash (/) followed by *~usersname* in order to reach a particular user's home page.

1. From the Web Manager home page click Enterprise Web Server *servername* > Content Management > User Document Directories.
2. To activate the service, click OK.

Providing Public Access

1. From the Web Manager home page, click Enterprise Web Server *servername* > Server Preferences > Restrict Access.
2. Click Insert File and enter the path (or any portion of the path you want to be public) in the following format:

   servername\volume:\directory\subdirectory
3. Click OK > Save Changes.
4. To effect the changes, click On/Off under Server Preferences to restart the server.

Web Publishing through WebDAV

There are various ways of publishing content to your Web server. (See "How to Publish Content to Your Web Server" on page 46.) Web-distributed Authoring and Versioning (WebDAV) is an industry standard protocol and is an enhancement to the HTTP protocol, turning the Web into a document database that enables collaborative creation, editing, and searching from remote locations.

For your users to benefit from WebDAV, it must first be enabled. WebDAV is enabled by default once you install NetWare 6. But if you need to enable or disable it, you can do so from NetWare Web Manager.

To enable WebDAV on your Enterprise Server, do the following:

1. From the NetWare Web Manager home page, click NetWare Enterprise Web Server *servername* > WebDAV.
2. Under WebDAV State, click On.
3. Click OK.
4. Click Save and Apply.

Configuring Document Preferences

You can configure the following document preferences from the Web Manager home page by clicking Enterprise Web Server *servername* > Content Management > Document Preferences.

Specifying a Default Home Page

If a document name is not specified in a URL, the Web server will look for a specific filename (or home page) such as INDEX.HTML, and return it to the Web browser. The filename that the Web Server looks for is configurable using the Document Preferences page of Web Manager. If the specified filename cannot be found, the Web browser will display a listing of files and folders located at the URL.

By default, the Enterprise Web Server defines INDEX.HTML and HOME.HTML as the default home page filenames, but you can set these to whatever filename you choose.

If more than one name is specified, the server searches in the order in which the names appear in this field until one is found. For example, if your index filenames are INDEX.HTML and HOME.HTML, the server first searches for INDEX.HTML and, if it doesn't find it, the server then searches for HOME.HTML.

Directory Indexing

In your document directory, you'll probably have several subdirectories. For example, you might create a directory called PRODUCTS, another called SERVICES, etc. It's common to provide an overview (or index) of these directories.

The server indexes directories using the following process:

1. The server first searches the directory for an index file called INDEX.HTML or HOME.HTML, which is a file you create and maintain as an overview of the directory's contents. (These defaults can be configured for the whole server, so your server's files might vary. For more information, see "Specifying a Default Home Page" on page 69.) You can specify any file as an index file for a directory by naming it one of these default names, which means you can also use a CGI program as an index if CGI is activated.

2. If an index file isn't found, the server generates an index file that lists all the files in the document root. The generated index has one of the following formats:

   - Fancy directory indexing is fairly detailed. It includes a graphic that represents the type of file, the date the file was last modified, and the file size.
   - Simple directory indexing is less detailed, but also takes less time to generate.
   - You can also specify that no dynamic directory listing be generated if the server looks for index files and cannot find any. If the server does not find any index files, it will not create a directory listing to show the user and will return an error message.

Server Home Page

When users first access your server, they usually use a URL such as http://www.novell.com/. When the server receives a request for this document, it returns a document called a home page. Usually this file has general information about your company and links to other documents.

By default the server finds the index file specified in the Index Filenames field and uses that for the home page. However, you can also specify a file to use as the home page by selecting the Home Page icon (by the Location field) and entering the filename for the home page in the field.

About the Temporary Web Site

By default, the Enterprise Web Server includes a temporary Web site. The purpose of the site is to verify when your Web server is running. But it also includes information about, and links to, all of the Web-based NetWare 6 management tools.

You can replace the default INDEX.HTM and with your own home page and remove all of the supporting pages as well. You might want to explore the site before replacing it with your own content.

To view the temporary Web site, point your Web browser at your Web server by entering your NetWare server's domain name or IP address. For example:

http://www.digitalairlines.com

or

http://120.140.160.180

Default MIME Type

When a document is sent to a client, the server includes a section that identifies the document's type, so the client can present the document in the correct way. However, sometimes the server can't determine the proper type for the document because the document's extension is not defined for the server. In those cases, a default value is sent. For information about maintaining your server's Multipurpose Internet Mail Extension (MIME) types, see "Configuring MIME Types" on page 56.

The default is usually Text/Plain, but you should set it to the type of file most commonly stored on your server. Some common MIME types include the following:

text/plain	text/html
text/richtext	image/tiff
image/jpeg	image/gif
application/x-tar	application/postscript
application/x-gzip	audio/basic

Parsing the Accept Language Header

When clients contact a server using HTTP, they can send header information describing the languages they accept. You can configure your server to parse this language information.

For example, if you store documents in Japanese and English, you could choose to parse the Accept Language header. When clients that have Japanese as the Accept Language header contact the server, they receive the Japanese version of the page. When clients that have English as the Accept Language header contact the server, they receive the English version.

If you do not support multiple languages, you should not parse the Accept Language header.

Setting Document Preferences

1. Click Enterprise Web Server *servername* > Content Management > Document Preferences.
2. In the Index Filenames field, enter a new index filename.
3. Select the kind of directory indexing you want.

4. Select whether you want users to see a specified home page or an index file when they access your server. If you select the home page option, enter the filename of the home page you want in the Index File field.
5. In the Default MIME Type field, enter the default MIME type you want the server to return if a client accesses a file with an extension that has not been set up as a MIME type on your server.
6. Select whether or not to parse the accept language header.
7. Click OK > Save and Apply.

Forwarding URLs

Redirection is a method for the server to tell a user that a URL has changed—for example, if you have moved files to another directory or server. You can also use redirection to send a person who requests a document on one server to a document on another server.

To map a URL to another server, you must first specify the prefix of the URL you want the server to redirect. Then, you need to choose which URL to redirect to. You can redirect to a URL prefix if the directory on the new server is the same as in the mapped URL; you can also redirect to a fixed URL (hostname, directory, and filename).

To forward URLs, do the following:

1. Click Enterprise Web Server *servername* > Content Management > URL Forwarding.
2. In the URL Prefix field, enter the URL prefix you want to redirect.

   For example, if the URL you want to map is http://www.netscape.com/info/movies, you would type **`/info/movies`** in the field.
3. Select whether you want to forward requests to a URL prefix or to a fixed URL.
4. Click OK > Save and Apply.

If you forward to a URL prefix, the forwarding keeps the full pathname and substitutes one prefix for another. For example, if you forward http://www.novell.com/info/docs to a prefix cambridge.com, the URL http://www.novell.com/info/docs redirects to http://cambridge.com/info/docs.

However, if the directory structure on the new server is not the same as in the mapped URL, you could forward the URL to a fixed URL. For example, you

could forward http://www.novell.com/info/docs to http://cambridge.com/new-files/info/docs.

Sometimes you might want to redirect requests for all the documents in one subdirectory to a specific URL. For example, if you had to remove a directory because it was causing too much traffic or because the documents were no longer to be served for any reason, you could direct a request for any one of the documents to a page explaining why the documents were no longer available. For example, a prefix on /info/docs could be redirected to http://www.novell.com/explain.html.

Setting Up Multiple Web Servers

There are two approaches you can take to set up multiple Web servers on your NetWare server:

- Hardware virtual servers
- Software virtual servers

Each approach has its strengths and weaknesses; you should choose the one that's right for your situation.

Hardware virtual servers allow you to map multiple IP addresses to multiple document roots. For example, if you have two IP addresses, you could map the first IP address to one document root and the second IP address to a second document root. Although hardware virtual servers take fewer system resources than multiple instances of the server, they must also share the same configuration information. For example, if one hardware virtual server has enabled security features or Web Publishing, they all must have it enabled.

Software virtual servers give you the ability to map a single IP address to multiple server names. Each software virtual server can have its own home page, which allows you to host multiple Web sites from one IP address. However, in order for software virtual servers to work correctly, the users accessing the server must use client software that supports the HTTP host header. Like hardware virtual servers, software virtual servers all must have the same configuration.

For more information, see "Setting Up Hardware Virtual Servers" on page 74 and "Setting Up Software Virtual Servers" on page 75.

Setting Up Hardware Virtual Servers

A hardware virtual server lets your server respond to multiple IP addresses without your having to install multiple servers. With hardware virtual servers, you map multiple IP addresses to multiple document roots. For example, if you have two IP addresses, you could map the first IP address to one document root and the second IP address to another document root.

Although hardware virtual servers take fewer system resources than multiple instances of the server, they must also share the same configuration information. For example, if one hardware virtual server has enabled security features, they all must have security features enabled.

To set up hardware virtual servers, dothe following:

1. Load and bundle all IP addresses.
2. Enter the following command at the NetWare server console:

   `add secondary IPaddress IP_address`
3. Add the above command to the AUTOEXEC.NCF file after the LOAD and BIND statements or after INITSYS.NCF if INETCFG is being used to configure the server.
4. Click Enterprise Web Server *servername* > Content Management > Hardware Virtual Servers.
5. In the IP Address field, enter the secondary IP address.
6. In the Document Root field, enter the document root. For example, SYS:NOVONYX\SUITESPOT\DOCS.
7. To secure your hardware virtual server, check the Encryption check box.

   See "About Securing a Hardware Virtual Server" on page 74 for additional information.
8. Click OK > Save and Apply.
9. Repeat the previous steps for each hardware virtual server.

About Securing a Hardware Virtual Server

For more information on security, refer to *Novell Certificate Server Administration Guide*.

While the Enterprise Web Server doesn't have to be secured for a hardware virtual server to be secured, you do have to specify a Key Material Object

(KMO) during installation to use encryption. Once the KMO is created, use Server Preferences > Encryption On/Off to select a KMO.

IMPORTANT: Once you have turned Encryption on, you must use HTTPS to contact this server rather than HTTP.

Setting Up Software Virtual Servers

A software virtual server is a way to host several Web sites on one computer without needing to have more than one IP address on the computer. For example, you can set up your system so that both www.novell.com and www.cambridge.com resolve to 192.3.4.5, and then set up software virtual servers to handle both server names (for example, http://www.novell.com/ and http://www.cambridge.com). The server can respond differently to requests depending upon the URL, even though the server only has one IP address.

For example, an Internet service provider (ISP) installs a Web server and then wants to set up a software virtual server for each of its customers (for example, customers aaa, bbb, and ccc) so that each customer can have an individual domain name.

The ISP first configures the Domain Name System (DNS) to recognize that a customer's URL, www.aaa.com, resolves to the ISP's IP address. The ISP then creates a subdirectory for each company (aaa, bbb, and ccc) in the document root. These subdirectories contain the files for that company, including the home page, aaa/HOME.HTML. Next, the ISP sets up software virtual servers. The URL host would be www.aaa.com and the home page would be aaa/HOME.HTML. The ISP would do this for each company it services.

Because software virtual servers use the HTTP host header to direct the user to the correct page, not all client software works with software virtual servers.

Only client software that supports the HTTP host header will work. In the previous example, the ISP would set up the INDEX.HTML file in the document root to be an index page that links to all the virtual servers hosted by the system, so all users could access the home pages.

To set up a software virtual server, do the following:

1. Click Enterprise Web Server *servername* > Content Management > Software Virtual Servers.
2. Create a directory under the DOCS directory. For example, SYS:\NOVONYX\SUITESPOT\DOCS\TEST.

3. In the URL Host field, enter the URL host whose custom home page you want to set up. For example, *test*/.
4. In the Home Page field, enter the path to the home page you want to use for this virtual server. For example INDEX.HTML.

   If you enter a full path, the server uses that specific document. If you enter a partial path, the server interprets it as relative to your primary document directory.
5. Click OK > Save and Apply.
6. If you want to modify preferences on the default home page, click Edit the Default Home Page at the top of the form.

About the Drop-Down Lists

When working on the International Characters, Document Footer, Parse HTML, and Cache Control Directives pages of Web Manager and your Web server, you will find a drop-down box at the top of each page. It works the same for each of these features.

The drop-down list and associated Browse button let you select specific resources to be configured.

From the drop-down list, you select a resource to be configured. You can click Browse to browse your primary document directory, Options to choose other directories, and Wildcard to configure files with a specific extension.

Wildcards Used in the Drop-Down List

In many parts of the server configuration, you specify wildcard patterns to represent one or more items to configure. Note that the wildcards for access control and text search might be different from those discussed in this section.

Wildcard patterns use special characters. If you want to use one of these characters without the special meaning, precede it with a backslash (\) character.

Table 3 Drop-Down Wildcard Patterns

Pattern	Use
*	Match zero or more characters.
?	Match exactly one occurrence of any character.

Pattern	Use
|	An *or* expression. The substrings used with this operator can contain other special characters such as an asterisk (*) or a dollar sign ($). The substrings must be enclosed in parentheses—for example—(a|b|c), but the parentheses cannot be nested.
$	Match the end of the string. This is useful in *or* expressions.
[abc]	Match one occurrence of the characters a, b, or c. Within these expressions, the only character that needs to be treated as a special character is the right bracket ( ]); all others are not special.
[a-z]	Match one occurrence of a character between A and Z.
[^az]	Match any character except A or Z.
*~	This expression, followed by another expression, removes any pattern matching the second expression.

Table 4 Drop-Down List Wildcard Examples

Pattern	Result
*.netscape.com	Matches any string ending with the characters .netscape.com.
(quark|energy).netscape.com	Matches either quark.netscape.com or energy.netscape.com.
198.93.9[23].???	Matches a numeric string starting with either 198.93.92 or 198.93.93 and ending with any three characters.
.	Matches any string with a period in it
~netscape-	Matches any string except those starting with netscape-
*.netscape.com~quark.netscape.com	Matches any host from domain netscape.com except for the single host quark.netscape.com.

Pattern	Result
*.netscape.com~(quark|energy|neutrino).netscape.com	Matches any host from domain netscape.com except for hosts quark.netscape.com, energy.netscape.com, and neutrino.netscape.com.
.com~.netscape.com	Matches any host from domain.com except for hosts from the subdomain netscape.com.

Assigning a Character Set

The character set of a document is determined in part by the language it is written in. For most Web browsers, you can override the default character set setting for a document, a set of documents, or a directory by selecting a resource and entering a character set for that resource.

Most Web browsers can use the MIME type charset parameter in HTTP to change its character set. If the server includes this parameter in its response, the Web browser changes its character set accordingly. The following are some character set examples:

```
Content-Type: text/html;charset=iso-8859-1
Content-Type: text/html;charset=iso-2022-jp
```

The charset names recognized by Netscape Communicator are specified in RFC 1700 (except for the names that begin with x-). These charset names include the following:

us-ascii	iso-8859-1
iso-2022-jp	x-sjis
x-euc-jp	x-mac-roman

Additionally, the following aliases are recognized for us-ascii:

ansi_x3.4-1968	iso-ir-6
ansi_x3.4-1986	iso_646.irv:1991
ascii	iso646-us
us	ibm367
cp367	

The following aliases are recognized for iso_8859-1:

latin1	iso_8859-1
iso_8859-1:1987	iso-ir-100
ibm819	cp819

To change the character set, do the following:

1. Click Enterprise Web Server *servername* > Content Management > International Characters.
2. Select the server resource you want to change the character set for from the Editing drop-down list.
3. To view the different server resources, click Browse .
4. To type the pattern you want to edit, click Wildcard .
5. In the Character Set field, enter one of the character sets previsously mentioned in this section.
6. Click OK > Save and Apply.

Specifying a Document Footer

You can specify a document footer, which can include the last-modified time, for all the documents in a certain section of your server without using server-parsed HTML. This footer works for all files except the output of CGI scripts or parsed HTML (.SHTML) files.

HINT: If you need your document footer to appear on CGI-script output or parsed HTML files, enter your footer text into a separate file and add a line of code or include another server-side to append that file to the page's output.

1. Click Enterprise Web Server *servername* > Content Management > Document Footer.
2. Select the resource that you want to apply the document footer to from the Editing drop-down list.
3. Click Browse to view the different server resources.
4. To enter the pattern you want to edit, click Wildcard .
5. In the For Files of Type field, enter the kind of files that you want to include in the footer. The default is text/html.

6. Select the time format from the drop-down list or enter a date in the Custom Date Format field.
7. In the Footer Text field, enter the footer text.

   The maximum number of characters for a document footer is 765. Type the string **:LASTMOD:** if you want to include the date the document was last modified.
8. Click OK > Save and Apply.
9. To change the footer text, click Deactivate Custom Trailer.

   When you change the document footer for an HTML page, the last-modified date doesn't change.

Customizing Parsed HTML

HTML is normally sent to the client exactly as it exists on disk without any server intervention. However, the server can search HTML files for special commands (that is, parse the HTML) before sending documents. If you want the server to parse these files and insert request-specific information or files into documents, you must first enable HTML parsing.

1. Click Enterprise Web Server *servername* > Content Management > Parse HTML.
2. Select the server resource to edit from the Editing drop-down list.
3. Click Browse to view the different server resources.
4. Click Wildcard to enter the pattern you want to edit.
5. Select whether or not you want to activate parsed HTML.

   If you activate it, you need to choose whether to activate it with or without the Exec tag. The Exec tag allows an HTML file to execute an arbitrary program on the server. You might not want to allow the Exec tag for security or performance reasons.
6. Select which files to parse.

   The default choice is to parse only files with the extension .SHTML. In this case, all files you want to parse must have the .SHTML extension. You can have the server parse all of its HTML files. Choosing this option can slow your server's performance.
7. Click OK > Save and Apply.

Using Cache-Control Directives

Cache-control directives are a way for the Enterprise Web Server to control what information is cached by a proxy server. By using cache-control directives, you override the default caching of the proxy to protect sensitive information from being cached and perhaps retrieved later. For these directives to work, the proxy server must comply with HTTP 1.1.

For specific directories in your server, you can set the cache-control directives to one of the following levels:

- Public: The response is cacheable by any cache.
- Private: The response is cacheable only by a private (non-shared) cache.
- No Cache: The response must not be cached anywhere.
- No Store: The cache must not store the request or response anywhere in nonvolatile storage.
- Must Revalidate: The cache entry must be revalidated from the originating server.
- Maximum Age (in seconds): The client does not accept a response that has a greater age than the maximum age.

To set the cache-control directives, do the following:

1. Click Enterprise Web Server *servername* > Content Management > Cache Control Directives.
2. Select the directory or directories that you want to set cache-control directives for from the Editing drop-down list.
3. Click Browse to view the different server resources.
4. Click Wildcard to enter the pattern you want to edit.
5. Select the level of control you want to set.

   The default is public.
6. Click OK.

   For more information on HTTP 1.1, see the Hypertext Transfer Protocol (http://www.ietf.org/html.charters/http-charter.html)(HTTP/1.1 specification [RFC 2068]).

Working with Configuration Styles

Configuration styles are an easy way to apply a set of options to specific files or directories that your server maintains. For example, you can create a configuration style that sets up access logging. When you apply that configuration style to the files and directories that you want to log, you don't have to individually configure access logging for all the files and directories.

Creating a Configuration Style

1. Click Enterprise Web Server *servername* > Styles > New Style.
2. In the Style Name field, enter the name you want to give the configuration style.
3. Click OK.
4. Select a configuration style to edit from the Style drop-down list and then click Edit This Style.
5. From the list of links available, click the category you want to configure for your style. You can configure the following information:
   - LCGI File Type: Lets you activate LCGI as a file type. For more information about working with Web applications, see Chapter 9, "Extending Your Server with Programs," on page 99.
   - Character Set: Lets you change the character set for a resource. For more information, see "Assigning a Character Set" on page 78.
   - Default Query Handler: Lets you set a default query handler for a server resource.
   - Document Footer: Lets you add a document footer to a server resource. For more information, see "Specifying a Document Footer" on page 79.
   - Error Responses: Lets you customize the error responses that clients see when they encounter an error from your server. For more information, see "Customizing Error Responses" on page 59.
   - Log preferences: Lets you set preferences for access logs. For more information, see "Working with Log Files" on page 115.
   - Restrict Access: Lets you restrict access to the entire server or parts of it. For more information, see "Restricting Access" on page 61.
   - Server Parsed HTML: Lets you specify whether the server parses files before they are sent to the client. For more information, see "Customizing Parsed HTML" on page 80.

6. Fill out the form that appears and then click OK.
7. Repeat Step 4 and Step 5 to make any other changes to the configuration style.
8. Click OK on the form you modified.
9. Click OK on the Edit a Style form.
10. Click Save and Apply.

Editing a Configuration Style

1. Click Enterprise Web Server *servername* > Configuration Styles > Edit Style.
2. Select a configuration style to edit from the Style drop-down list.
3. Click Edit This Style.
4. From the list of links available, click the category you want to configure for your style.

   For more information on these categories, see "Creating a Configuration Style" on page 82.
5. Fill out the form that appears > click OK.
6. Repeat Steps 4 and 5 to make any other changes to the configuration style.
7. Click OK on the form you modified.
8. Click OK on the Edit a Style form.
9. Click Save and Apply.

Applying a Configuration Style

Once you've created a configuration style, you can apply it to files or directories in your server. You can specify either individual files and directories or wildcard patterns, such as *.GIF.

1. Click Enterprise Web Server *servername* > Configuration Styles > Assign Style.
2. In the URL Prefix Wildcard field, enter the prefix of the URL that you are applying this configuration style to.

   If you select a directory inside the document root, only enter the path after the document root. If you enter **/*** after the directory, you apply the configuration style to all of the directory's contents.

3. Select the configuration style you want to apply from the Style drop-down list.
4. Click OK > Save and Apply.

Removing a Configuration Style

Before removing a configuration style, apply the None configuration style to any files or directories that had the configuration style applied to them. If you do not apply None before removing the configuration style, you must manually edit your OBJ.CONF file, search for the configuration style in the file, and replace it with None. If you don't do this search and replace, anyone who accesses the files or directories that the deleted configuration style was applied to will get a server configuration error message.

1. Click Enterprise Web Server *servername* > Configuration Styles > Remove Style.
2. Select the configuration style you want to remove from the Remove drop-down list.
3. Click OK > Save and Apply.

Listing Configuration Style Assignments

After you have created configuration styles and applied them to files or directories, you can get a list of the configuration styles and where you applied them.

1. Click Enterprise Web Server *servername* > Configuration Styles > List Assignments.
2. To edit a configuration style assignment, click Edit Style Assignment next to the configuration style name.

7 Using a Directory Service to Control User Access to Network Resources

Using a directory service with the Web technologies of NetWare® 6 lets you easily control which of your users can access sensitive data on your NetWare 6 server.

The NetWare Enterprise Web Server lets you choose which directory service mode you want the Enterprise Server to use. This chapter describes each of the available directory service modes and how to set them up.

The Directory Service

Directory services are a type of software that allow you to maintain information, such as contact information or identification information for the people in your organization. This information is also used when controlling access to a server and its resources by requiring a user to provide the necessary username and password when trying to access protected (or nonpublic) folders.

You can use three directory service modes with the Enterprise Web Server:

- eDirectory™ mode
- Local database mode
- LDAP mode

NOTE: If you choose local database or LDAP modes, users will be required to enter fully distinguished names (*.username.context.domain*) whenever accessing protected folders.

eDirectory Mode

Novell® eDirectory is installed during the NetWare 6 installation. The Enterprise Web Server provides a native eDirectory integration mode. While in this mode, users attempting to access any folder that is not designated as a public folder are required to enter a username and password. This is accomplished through eDirectory's use of native NetWare file system trustee assignments.

This means that anywhere that users have Internet access, they can log in from an HTTP client using their eDirectory usernames and passwords. There is no need for them to set up a dialup account into your company network. As long as they can get to the Internet, they can get to their data.

eDirectory is the default directory mode and is ready to go after installation of NetWare 6. After eDirectory mode is selected,

- Use ConsoleOne™ to maintain user and group information.

  HINT: You can also use the Users and Groups feature from within the Enterprise Server manager pages to perform basic eDirectory functions, such as adding, configuring, or removing User and Group objects. To access users and groups from the NetWare Web Manager home page, click your Enterprise *servername* > then Users and Groups.

- Users are required to log in from the Web browser to gain access to resources contained in nonpublic directories.
- Make sure SSL is enabled if you do not want eDirectory passwords visible to hackers.
- NetWare file system trustees assignments are available exclusively to control access to Web resources.
- NetWare Enterprise Web Server access control lists (ACLs) are disabled.

Local Database Mode

The local database is intended for sites running a public access Web site, or a site in which protected resources are limited and where you don't have a need to secure any or most of your information. Usernames and passwords are stored in a simple text file.

Local database mode has the following limitations when compared to eDirectory:

- Rights set up in local database mode don't carry over to your existing directory service; therefore, you have to maintain and synchronize two access control lists.

- The local database supports no more than 1,000 entries.
- The local database cannot be replicated.

LDAP Mode

If your company is built around LDAP, this mode might make sense for you. However, we recommend eDirectory as your directory service. eDirectory is the leading directory service available today and is included with NetWare 6.

For information about implementing LDAP with eDirectory, see *LDAP Services for Novell eDirectory* (http://www.novell.com/documentation/lg/nw6p/ndsedir86/index.html). Before using eDirectory as your LDAP server, you must first enable unencrypted passwords by opening the properties of your LDAP Server object using ConsoleOne.

Configuring Directory Services

This section describes how to configure a directory service (or the local database) for use with the Enterprise Web Server.

IMPORTANT: If you switch to or from local database mode, you need to restart all of the Web technologies that you have installed, including NetWare Web Manager. Keep in mind that Web Manager runs on the Apache Web server, which hosts other technology and services, including iFolder™. You might want to do this when you know user traffic is low.

Using eDirectory Mode

eDirectory is already configured by default. However, if you for any reason you need to reconfigure eDirectory mode, follow these steps.

1. From the NetWare Web Manager home page, click Enterprise Web Server *servername* > Users and Groups > Configure Directory Service.
2. Select eDirectory.

A dialog box appears to confirm that you want to use eDirectory.

3. Click OK.
4. To add a new search context, click Insert Context .

   HINT: This field is optional. However, by specifying the context of your admin user object, you will not be required to enter your fully distinguished name when prompted to authenticate to Web Manager or other protected resources.
5. Click Remove Context to remove one or more search contexts.
6. Click Float Context to move the selected context to a higher priority context.
7. Click Save Changes.
8. (Conditional) If you change directory service from a local or remote LDAP directory to eDirectory, you need to restart the Web server.

   NetWare Web Manager does not need to be restarted.

HINT: eDirectory does not allow public access to nonpublic folders or files. All users must be authenticated before receiving any content. Content that is placed in public directories do not require authentication. For more information on setting up public directories, see "Setting Additional Document Directories" on page 64.

Using Local Database Mode

1. From the NetWare Web Manager home page, click Enterprise Web Server *servername* > Users and Groups > Configure Directory Service.
2. Click Local Database.

   A dialog box appears to warn you that you will lose your directory service configuration information.
3. Click OK.
4. In the Base DN field, enter the distinguished name to be used as a suffix for your local directory and also as the point which directory lookups will occur from by default.

   An example of a suffix that you could enter here is

   `o= your_company_name, c=US`

   If you do not enter a value in this field, then your suffix will be a null string and all searches will begin from the top or root point of the directory.

5 Click Save Changes.

Using LDAP Mode

1 From the NetWare Web Manager home page, click Enterprise Web Server *servername* > Users and Groups > Configure Directory Service.

2 Click LDAP Directory Server.

A dialog box appears to confirm that you want to use a directory server.

3 Click OK.

4 In the Host Name field, enter the hostname where the directory server is running.

You must enter a hostname even if the directory server is running on the local machine.

5 In the Port field, enter the default number if your directory server is using a different port number than the default port number 389.

6 In the Base DN field, enter the distinguished name that will be the point which directory lookups will occur from by default and will be the location where all NetWare Web Manager's entries will be placed in your directory tree.

An example of a base DN that you could enter here is

`o= you_company_name, c=US`

7 In the Bind DN field, enter the bind DN that NetWare Web Manager will use to initially bind (or log in) to the directory server.

This bind DN requires only Read and Search access to the directory. Because this DN and the associated password (if any) are easily compromised, it is best to simply leave this field blank and then set up your directory server to allow anonymous search access. If you do not want to allow anonymous search access to your directory, then specify a

bind DN entry here that has only Read and Search access to your directory.

IMPORTANT: Do not specify your directory server's admin username in this field. This bind DN is used only to initially search for the username you entered in NetWare Web Manager authentication dialog box. Once the entry corresponding to this username is located, NetWare Web Manager rebinds to the directory server using the retrieved entry. Therefore, if the username you supplied when you first logged in to NetWare Web Manager does not have access to the directory server, you will not have any access to the directory server, regardless of the bind DN information provided in this field.

8. (Optional) In the Bind Password field, enter the password for the bind DN entry if you have entered a bind DN in the previous field.

9. Click Save Changes.

8 Understanding ACL Files

This chapter describes the access control list (ACL) files and their syntax. ACL files are text files that contain lists that define who can access resources stored on your Web server. By default, the Web server uses one ACL file that contains all of the lists for access to your server. However, you can create multiple ACL files and reference them in the OBJ.CONF file.

You need to know the syntax and function of ACL files if you plan to customize access control using the access control API. For more information on the API, see the Novell Developer Kit Web site (http://www.developer.novell.com/ndk/nscomp.htm).

Using either local database or LDAP directory modes, you manage access control through the access control lists (ACL). With Novell® eDirectory™, you manage access control through NetWare® file system trustees. For more information, see Chapter 7, "Using a Directory Service to Control User Access to Network Resources," on page 85 and "Controlling Access with NetWare Web Access Controls" on page 274.

ACL File Syntax

An ACL file is a text file containing one ACL or more. All ACL files must follow a specific format and syntax. All ACL files must begin with the version number they use. There can be only one version line and it can appear after any comment lines. For example:

```
version 3.0;
```

You can include comments in the file by beginning the comment line with the pound (#) sign.

Each ACL in the file begins with a statement that defines its type. ACLs can follow one of three types:

- Path ACLs specify an absolute path to the resource they affect.
- URI (Uniform Resource Indicator) ACLs specify a directory or file.
- Named ACLs specify a name that is referenced in resources in the OBJ.CONF file. The server comes with a default named resource that allows read access to anyone and write access to users in the local database or LDAP directory. Even though you can create a named ACL from the Server Preference forms, you must manually reference the named ACLs with resources in the OBJ.CONF file.

The type line begins with the letters *acl* and then includes the type information in double quotation marks followed by a semicolon. Each type information for all ACLs must be a unique name, even among different ACL files. The following lines are examples of several different types of ACLs:

```
acl "path=C:\Netscape\SuiteSpot\docs\mydocs\";

acl "*.html";

acl "default";

acl "uri=/mydocs/";
```

After you define the type of ACL, you can have one or more statements that define the method used with the ACL (authentication statements) and the people and computers who are allowed or denied access (authorization statements). The following sections describe the syntax for these statements.

Authentication Statements

ACLs can optionally specify the authentication method the server must use when processing the ACL. There are two general methods:

- Basic requires users to enter a username and password before accessing a resource.
- SSL requires the user to have a client certificate. For this method to work, the Web server must have encryption turned on.

By default, the server uses the basic method for any ACL that doesn't specify a method. You can change the default setting by editing the following line in the MAGNUS.CONF file:

```
Init fn=acl-set-default-method method=SSL
```

Each authenticate line must specify what list (users, groups, or both) the server should use when authenticating users. The following authentication statement, which would appear after the ACL type line, specifies basic authentication with users matched to individual users in the database or directory:

```
authenticate (user) {
       method = basic;
  };
```

The following example uses SSL as the authentication method for users and groups:

```
authenticate (user, group) {
       method = ssl;
  };
```

Any allow or deny statements must match the lists you specify in the authenticate line. If the line says authenticate (user), the allow or deny line must also specify users. The following example allows any user whose username begins with the letters *sales*:

```
authenticate (user)
  allow (all)
      user = sales*
```

If the last line was changed to group = sales, then the ACL would fail because there are no groups in the user lists.

Authorization Statements

Each ACL entry can include one or more authorization statements, which specify who is allowed or denied access to a server resource. Use the following syntax when writing authorization statements:

```
allow|deny [absolute] (right[,right...]) attribute qualifier
  expression;
```

Start each line with either *allow* or *deny*. It's usually a good idea to deny access to everyone in the first rule or command you enter and then specifically allow

access for users, groups, or computers in subsequent rules. This is because of the hierarchy of rules.

For example, if you allow anyone access to a directory called MY_STUFF, then you have a subdirectory MY_STUFF/PERSONAL that allows access to a few users. The access control on the subdirectory won't work because anyone allowed access to the MY_STUFF directory will also be allowed access to the MY_STUFF/PERSONAL directory. To prevent this, create a rule for the subdirectory that first denies access to anyone and then allows it for the few users who need access.

However, in some cases, if you set the default ACL to deny access to everyone, then your other ACL rules don't need a Deny All rule.

The following line denies access to everyone:

```
deny (all)
      user = "anyone";
```

Hierarchy of Authorization Statements

ACLs have a hierarchy that depends on the resource. For example, if the server receives a request for the document (URI) /MY_STUFF/WEB/PRESENTATION.HTML, the server first looks for an ACL that matches the file type or any other wildcard pattern that matches the request, then it looks for one on the directory, and finally it looks for an ACL on the URI. If there is more than one ACL that matches, the server uses the last statement that matches.

However, if you use an absolute statement, then the server stops looking for other matches and uses the ACL containing the absolute statement. If you have two absolute statements for the same resource, the server uses the first one in the file and stops looking for other resources that match.

For example, using the ACL hierarchy with the request for the document /MY_STUFF/WEB/PRESENTATION.HTML, you could have an absolute ACL that restricts access to the file type *.HTML then the server would use that ACL instead of looking for one that matches the URI or the path.

```
version 3.0;
  acl "default";
  authenticate (user,group) {
      prompt="Enterprise Server";
       };
       allow (write,delete)
            user="all";
```

```
acl "*.html";
      deny absolute (all)
           user="anyone";
acl "uri=/my_stuff/web presentation.html";
       deny (all)
            user="anyone";
     allow (all)
          user="anyone";
```

Attribute Qualifier Expressions

Attribute qualifier expressions define who is allowed or denied access based on their username, group name, hostname, or IP address. The following lines are examples of allowing access to different people or computers:

user = “anyone”

user = “smith*”

group = “sales”

dns = “*.organization.com”

dns = “*.organization.com” or “*.accounting_mail.com”

ip = “198.*”

You can also restrict access to your server by time of day (based on the local time on the server) by using the *timeofday* attribute qualifier. For example, you can use the *timeofday* attribute qualifier to restrict access to certain users during specific hours.

Use a 24-hour clock to specify times (for example, use 0400 to specify 4 a.m. or 2230 for 10:30 p.m.).

The following example restricts access to a group of users called Guests between 8:00 a.m. and 4:59 p.m.

```
allow (read)
       (group="guests") and
       (timeofday<800 or timeofday>=1700);
```

You can also restrict access by day of the week. Use the following three-letter abbreviations to specify days of the week: Sun, Mon, Tue, Wed Thu, Fri, and Sat.

The following statement allows access for users in the Premium group any day and any time. Users in the Discount group get access all day on weekends and on weekdays anytime except 8:00 a.m. to 4:59 p.m.

```
allow (read) (group="discount" and dayofweek="Sat,Sun") or
  (group="discount" and (dayofweek="mon,tue,wed,thu,fri" and
  (timeofday<0800 or timeofday>=1700)))
  or
  (group="premium");
```

Operators for Expressions

You can use various operators in attribute qualifier expressions. You can use parentheses to delineate the order of precedence of the operators. With user, group, dns, and ip qualifiers, you can use the following operators:

and

or

not

= (equals)

!= (not equal to)

With *timeofday* and *dayofweek* qualifiers, you can use the following additional operators:

> (greater than)

< (less than)

>= (greater than or equal to)

<= (less than or equal to)

Default ACL File

After installing the server, it uses the default settings in the file SERVER_ROOT/HTTPACL/GENERATED.HTTPS-SERVERID.ACL.

There is also a file called GENWORK.HTTPS-SERVERID.ACL that is a working copy that the server uses until you save and apply your changes when working with the user interface. When editing the ACL file, you might want to work in the GENWORK file and then use Server Preferences to save and apply the changes.

The following text is from the default file:

```
# File automatically written
  #
  # You may edit this file by hand
  #

  version 3.0;

  acl "agents";
  authenticate (user,group) {
            prompt = "Enterprise Server";
  };
  deny (all)
      user = "anyone"
  allow absolute (all)
      user = "all";

  acl "default";
  allow (read,execute,list,info)
      user = "anyone";
  allow (write,delete)
      user = "all";
```

The default ACL file is referenced in MAGNUS.CONF as follows:

```
ACLFile absolutepath/generated.https-serverid.acl
```

You can reference multiple ACL files in MAGNUS.CONF and then use their ACLs for resources in OBJ.CONF. However, the server uses only the first ACL file with the Web Publisher and with evaluation of access control for objects that don't have specific ACLs listed in OBJ.CONF. If you're using the Server Preference form to do some access control, the first ACL file in MAGNUS.CONF should point to the file GENERATED.HTTPS-SERVERID.ACL. See "Referencing ACL Files in OBJ.CONF" on page 98 for more information.

Input strings can contain the following characters:

- Letters A through Z
- Numbers 0 through 9
- Period (.) and underscore (_)

If you use any other characters, use double quotation marks (" ") around the characters.

A single statement can be placed on its own line and be terminated with a semicolon. Multiple statements are placed within braces. A list of items must be separated by commas and enclosed in double quotation marks.

Referencing ACL Files in OBJ.CONF

If you have named ACLs or separate ACL files, you can reference them in the OBJ.CONF file. You do this in the PathCheck directive using the check-acl function. The line has the following syntax:

`PathCheck fn="check-acl" acl="`*`acl_name`*`"`

The *acl_name* is a unique name of an ACL as it appears in any ACL file.

For example, you might add the following lines to your OBJ.CONF file if you want to restrict access to a directory using the ACL named *testacl*:

```
<Object ppath="/usr/ns-home/docs/test/*">
  PathCheck fn="check-acl" acl="testacl"
  </Object>
```

In this example, the first line is the object that states which server resource you want to restrict access to. The second line is the PathCheck directive that uses the check-acl function to bind the named ACL (testacl) to the object that the directive appears in. The testacl ACL can appear in any ACL file referenced in MAGNUS.CONF.

9 Extending Your Server with Programs

In addition to serving HTML documents, your server can run programs that interact with clients. These applications that run on the server are called server-side applications. Client-side applications are downloaded to the client and run on the client machine.

Your server can run these types of server-side applications:

- Local Common Gateway Interface (LCGI) programs
- JavaScript applications
- Plug-in programs that use the server plug-in APIs, such as the Netscape Server Plug-In (NSAPI)

This chapter describes how to install Java applets, CGI programs, and JavaScript applications onto your server. Plug-ins extend or replace your server's features. For example, you can use plug-ins to provide a different way to control access or to log in.

For information on writing and installing plug-ins, see the Novell Developer Kit Web site (http://www.developer.novell.com/ndk/doc.htm).

Additionally, your server can send server-side JavaScript programs to clients.This chapter deals mainly with the installation and configuration of server-side programs.

This chapter also describes the steps for specifying a default query handler CGI program. A query handler processes text sent to it via the ISINDEX tag in an HTML file.

Installing Server-Side Programs

JavaScript applications and CGI programs have different strengths and uses. CGI programs can be written in C, PERL, or other programming languages. All CGI programs have a standard way to pass information between clients and servers. JavaScript applications are written in JavaScript, an object-based scripting language that is easier to learn than an object-oriented programming language and lends itself to rapid application development.

Each type of program is installed onto the server differently. The following list summarizes the procedures:

- For CGI programs, configure your server to recognize certain files as CGI-all files with certain filename extensions or all files in specified directories.
- For JavaScript applications, check in each application individually through the Application Manager, which you can access from the Programs form or separately.

These installation instructions are described in the following sections.

Installing CGI Programs

Common Gateway Interface (CGI) programs can be created with any number of programming languages. On a UNIX* machine, you're likely to find CGI programs written as Bourne shell or PERL scripts. On a Windows computer, you might find CGI programs written in C++ or batch files. On NetWare®, you might find CGI programs written in NetBasic*, PERL, Novell Script for NetWare (NSN), or LCGI NLM™ applications.

Regardless of the programming language, all CGI programs accept and return data in the same manner.

There are two ways to store CGI programs on your server:

1. Specify a directory that contains only CGI programs. All files are run as programs regardless of the file extensions.
2. Specify that CGI programs are all a certain file type. They will all use the file extensions .CGI, .EXE, .NLM, or .BAT. The programs can be located in any directory that the server can serve from.

There are benefits to either implementation. If you want only a specific set of users to be able to add CGI programs, keep the CGI programs in specified directories and restrict access to those directories. If you want to allow anyone

who can add HTML files to be able to add CGI programs, use the file type alternative. Users can keep their CGI files in the same directories as their HTML files.

If you choose the directory option, your server will attempt to interpret any file you place in that directory as a CGI program. Similarly, if you choose the file type option, your server will attempt to process any files with the file extensions .CGI, .EXE, .NLM, or .BAT as CGI programs. If a file has one of these extensions but is not a CGI program, an error occurs when a user attempts to access it.

Specifying a CGI Directory

1. From the Web Manager home page, click Enterprise Web Server *servername* > Programs > CGI Directory.
2. In the URL Prefix field, enter the URL prefix you want to use for this directory.

   The text you enter appears as the directory for the CGI programs in URLs. For example, if you enter **`cgi-bin`** as the URL prefix, then all URLs to these CGI programs have the following structure:

   http://*yourserver.domain*.com/cgi-bin/program-name

   The URL prefix you specify can be different from the real CGI directory you specify in the next step.
3. In the CGI Directory field, enter the location of the directory as an absolute path.

   This directory doesn't have to be under your document root. This is the reason that you need to specify a URL prefix in the previous step.
4. Click OK > Save and Apply.

Editing an Existing CGI Directory

1. From the Web Manager home page, click Enterprise Web Server *servername* > Programs > CGI Directory.
2. Under Current CGI Directories, click Edit next to the directory you want to edit.
3. In the URL prefix field, enter the new prefix.
4. In the CGI directory field, enter the new directory.
5. Click OK > Save and Apply.

Removing an Existing CGI Directory

1. From the Web Manager home page, click Enterprise Web Server *servername* > Programs > CGI Directory.
2. Under Current CGI directories, click Remove next to the directory you want to remove.
3. Click OK > Save and Apply.

   Copy your CGI programs into the directories you've specified. Remember that any files in those directories will be processed as a CGI file, so you don't want to put HTML files in your CGI directory.

Specifying CGI as a File Type

1. From the Web Manager home page, click Enterprise Web Server *servername* > Programs > CGI File Type.
2. Select the resource you want to apply this change to from the Editing drop-down list.
3. Click Browse to choose a part of your server.
4. Click Options to browse files and directories on your server.
5. Click Back to return to the CGI as a File Type form.
6. Click Wildcard to enter the wildcard pattern to edit.
7. Click Yes to activate CGI as a file type.
8. Click OK > Save and Apply.

The CGI files must have the file extension .BAT, .EXE, .NLM, or .CGI. Any non-CGI files with those extensions will be processed by your server as CGI files and will cause errors.

Downloading Executable Files

If you're using .EXE as a CGI file type, users will not be able to download .EXE files as executables.

One solution to this problem is to compress the executable files that you want users to be able to download, so that the extension is not .EXE. This solution has the added benefit of shortening the download time.

Another possible solution is to remove .EXE as a file extension from the MAGNUS-INTERNAL/CGI type and add it to the APPLICATION/OCTET-STREAM type (the MIME type for normal downloadable files). You can do this by clicking Enterprise Web Server *servername* > Server Preferences > MIME Types. However, the disadvantage to this method is that after making this change, you cannot use .EXE files as CGI programs.

Another solution is to edit your server's OBJ.CONF file to set up a download directory, where any file in the directory is downloaded automatically. The rest of the server won't be affected. For directions on setting up this directory, visit the Netscape Technical Support Knowledge Base (http://help.netscape.com/kb/server/960513-130.html).

Scripting CGI

Refer to the Novell Developer Kit Web site (http://www.developer.novell.com/ndk/doc.htm) for information on PERL, NetBasic, NSN, and LCGIs.

Using the Query Handler

You can specify a default query handler CGI program. A query handler processes text sent to it via the ISINDEX tag in an HTML file.

ISINDEX is similar to a form text field in that it creates a text field in the HTML page that can accept typed input. Unlike the information in a form text field, however, the information in the ISINDEX box is immediately submitted

when the user presses Enter. When you specify your default query handler, you tell your server the program to direct the input to. For an in-depth discussion of the ISINDEX tag, see an HTML reference manual.

To set a query handler, do the following:

1. From the Web Manager home page, click Enterprise Web Server *servername* > Programs > Query Handler.
2. Select the resource you want to set a default query handler for from the Editing drop-down list.

   If you choose a directory, the query handler you specify runs only when the server receives a URL for that directory or any file in that directory.
3. Click Browse to choose a part of your server.
4. Click Options to browse files and directories on your server.
5. Click Back to return to Query Handler form.
6. Click Wildcard to enter the wildcard pattern to edit.
7. In the Default Query Handler field, enter the full path for the CGI program you want to use as the default for the resource you selected.
8. Click OK > Save and Apply.

Installing Server-Side JavaScript Programs

To install server-side JavaScript programs, you need to activate server-side JavaScript for your server and use the Application Manager. This section includes information on accessing and using the Application Manager to install server-side JavaScript applications as well as to perform other functions.

For more information about writing JavaScript applications, see the Novell Developer Kit Web site (http://www.developer.novell.com/ndk/doc.htm).

You must activate server-side JavaScript before you can use the Application Manager. Also, put JSAC.EXE and LIBESNSPR20.DLL in your system directory so that they are in the search path. These files are found in the NOVONYX/SUITESPOT/BIN/HTTPS directory.

Activating Server-Side JavaScript

If you are using server-side JavaScript applications, you must first activate server-side JavaScript for your server.

1. From the Web Manager home page, click Enterprise Web Server *servername* > Programs > Server Side JavaScript.
2. Select Yes to activate the server-side JavaScript application environment.
3. Click OK > Save and Apply.
4. When the Activate Server Side JavaScript form appears, click the link to use the Application Manager.
5. Enter the NetWare Web Manager username and password to use the Application Manager.

   For more information, see "Securing the Application Manager" on page 107.

For applications written in server-side JavaScript, you can perform the following administrative tasks with the server-side JavaScript Application Manager:

- Install a new JavaScript application. (You must add an application before users can run it.)
- Modify any of the attributes of an installed application. For example, its default home page, path to the .WEB file, and type of client-object maintenance.
- Stop, start, and restart an installed application.
- Run and debug an active application.
- Remove an installed application.

Running the Application Manager

To run the Application Manager, click Enterprise Web Server *servername* > Programs > Server Side JavaScript > Application Manager. You can also run the Application Manager by loading the following URL in your Web browser:

https://*server.domain*/appmgr

The Application Manager displays all applications currently installed on the server in a scrolling list in the left frame. Click an application in the scrolling list.

For the selected application, the right frame displays the following:

- Application name (at the top of the frame)
- Path of the application .WEB file on the server
- Default and initial pages for the application
- Number of built-in maximum database connections allowed
- External libraries used by the application (if any)
- Client object maintenance technique
- Status of the application: Active or Stopped (Users can run only active applications.)

To modify applications, do the following:

1. In the Applications list, select the application you want to modify.
2. Click the task button below the Applications list to perform the specified action:
   - Start activates the application.

     See "Starting, Stopping, and Restarting a Server-Side JavaScript Application" on page 109.
   - Stop stops the application.

     See "Starting, Stopping, and Restarting a Server-Side JavaScript Application" on page 109.
   - Restart restarts the application that was previously started and then stopped.

     See "Starting, Stopping, and Restarting a Server-Side JavaScript Application" on page 109.
   - Run: Retrieves application-Home form.

     See "Running a Server-Side JavaScript Application" on page 110.
   - Debug: Retrieves application-Home form.
   - Modify: Retrieves the specified application form.

     See "Modifying Installation Parameters" on page 109.
   - Remove: Removes the application.

     See "Removing a Server-Side JavaScript Application" on page 109.

The following explains the buttons and links on the green banner that runs across the top of the screen:

- Configure configures the default settings for Application Manager
- Add Application installs a new JavaScript application
- Documentation provides further documentation on server-side JavaScript
- Help provides instructions for using Application Manager

Securing the Application Manager

If you have disabled Web Manager's encryption, intruders can easily intercept your administrator username and password. Because Application Manager uses the same administrator username and password, intruders could also access Application Manager and add, remove, modify, start, stop, or delete your applications.

You should either enable Web Manager's encryption or avoid accessing it from outside of your company's firewall.

Installing Server-Side JavaScript Applications

You must install (add) an application with the Application Manager before you can run it. You can install up to 120 JavaScript applications on one server.

1. From the Web Manager home page, click Enterprise Web Server *servername* > Programs > Server Side JavaScript.
2. Click Application Manager.
3. Click Add Application at the top of the page.
4. In the Name field, enter the name of the JavaScript application.

   For specific information on application URLs, see "Application URLs" on page 108.

   IMPORTANT: Do not give any JavaScript applications the same names as any subdirectories of your primary document directory. If you do, the server will no

longer correctly process requests from the directory. For example, if you have a directory *server_root*/DOCS/BUG and a JavaScript application named Bug, all requests for any files in the BUG directory (or any of its subdirectories) will attempt to launch the JavaScript application Bug. The JavaScript application URI takes precedence.

5. In the Web File Path field, enter the absolute path to the .WEB file for the application.
6. In the Default Page field, enter the absolute path of the file to send to clients who do not indicate a specific page for the application.

   This page is analogous to INDEX.HTML for a standard URL. This is a required field.
7. (Optional) In the Initial Page field, enter the absolute path of the page to run when the application is first started.

   This page runs only once during the life of the application and is used to initialize values and establish database connections.
8. (Optional) In the Built-in Maximum Database Connections field, enter the maximum number of database connections that this application can have at one time if you are using the built-in Database object.
9. (Optional) In the External Libraries field, enter the absolute paths of any libraries to be used with the application.

   Libraries installed for one application can be used by all applications on the server.
10. In the Client Object Maintenance field, select the mode for maintaining the Client object.

    This can be Client-Cookie, Client-URL, Server-IP, Server-Cookie, or Server-URL.
11. Click OK > Save and Apply.

Application URLs

When you install a server-side JavaScript application, you must enter a name for it. This name determines the application URL, which clients use to access a JavaScript application. Application URLs are of the form http://*server.domain*/*appName*/*page*.html. *Server* is the name of the HTTP server, *domain* is the Internet domain (including the subdomains), *appName* is the application name you enter when you install it, and *page* is the name of the page in the application.

For example, if your server is named MYSERVER, your domain name is NOVELL.COM, and the application is called Hello World, the application URL is HTTP://MYSERVER.NOVELL.COM/WORLD/HELLO.HTML

This is a required field, and the name you enter must be different from all other application names on the server. The name must include only alphanumeric characters and cannot include spaces.

IMPORTANT: Before you install an application, make sure the application name you choose does not usurp an existing URL on your server. All client requests for URLs that match the application URL are routed to the directory specified for the .WEB file, circumventing the server's normal document root.

Controlling Access to a Server-Side JavaScript Application

When you install an application, you might want to restrict its use to only certain users. You can do this by applying a configuration style to the application. For more information, see "Working with Configuration Styles" on page 82.

Modifying Installation Parameters

You can change any of the parameters defined when you installed the application, except the application name. To change the name of an application, you must remove the application and then reinstall it.

If you modify the parameters of a stopped application, the Application Manager automatically starts it. When you modify parameters of an active application, Application Manager automatically stops and restarts it.

Removing a Server-Side JavaScript Application

Clicking Remove removes the application from the Application Manager, but does not delete files from the server. At this point, clients can no longer access the application.

If you delete an application and subsequently want to run it, you must install it again.

Starting, Stopping, and Restarting a Server-Side JavaScript Application

- Start starts an installed application that is stopped. If the application starts successfully, clients can run the application.
- Stop stops an active application. The application's status changes to stopped, and clients can no longer run the application. You must stop an

application if you want to move the .WEB file or update an application from a development server to a deployment server.

- Restart restarts a running application. For any changes you have made to take effect, you must restart an application after you compile it.

You can also start, stop, and restart an application by entering a special URL in the format

```
http://server.domain/appmgr/
  control.html?name=appName&cmd=action
```

where *appName* is the application name and *action* is either stop, start, or restart.

Running a Server-Side JavaScript Application

There are two ways to run an installed application:

- Select the application name in the Application Manager and click Run. A new Web browser window accesses the application.
- Enter the application URL in your Web browser.

If you attempt to run a stopped application (one that is not active), then the Application Manager tries to start it first.

WARNING: The server should not be unloaded while a server-side JavaScript application is running because the server can leave the application in an unpredictable state.

Configuring Default Settings

When you install a new application, the default installation parameters are used for the initial settings.

You can specify the following default settings:

- Installation parameters of .WEB file path, default page, initial page, maximum number of built-in database connections, external libraries, and client object maintenance technique. You can specify a default directory path for your development area and native executables libraries.
- Prompts to confirm your action when you remove, start, stop, or restart an application.
- The application trace to appear, when debugging an application, in the same window as the application, but in another frame or in a window separate from the application.

Installing Client-Side Programs

Installing client-side programs in your server is relatively easy. There are two types of client-side programs: Java applets and JavaScript programs.

Client-side Java applets are executable files identified in an HTML document, retrieved from the server, and executed on the client. The applets can reside anywhere under your server's primary document root.

Client-side JavaScript programs are created by lines of JavaScript code embedded in HTML files. The HTML files travel from the server to the client. Once the files reach the client, the Web browser interprets the JavaScript code and performs the specified actions.

With LiveConnect you can connect server-side Java and JavaScript applications or client-side Java and JavaScript applications. For more information on LiveConnect, on embedding JavaScript in HTML, and on using client-side JavaScript with other programs, see the Novell Developer Kit Web site (http://www.developer.novell.com/ndk/doc.htm).

About Tomcat for NetWare

Tomcat enables the NetWare Enterprise Web Server to execute Java servlets. A servlet can be thought of as a server-side applet without a user interface. Tomcat provides Web application developers with additional functionality. For example, a servlet could be written and deployed to process data obtained from a client via an HTML form and the server-side data processing could manipulate the data and store results in a database. Servlets provide an alternative to CGI.

For Tomcat documentation, refer to the HTML files found on your NetWare 6 server under SYS:\TOMCAT\33\DOC\INDEX.HTML.

You can also visit http://jakarta.apache.org (http://jakarta.apache.org) for the latest Tomcat news.

Migrating from WebSphere to Tomcat

If you have been using IBM* WebSphere Application Server for NetWare, you can migrate your existing Web applications to Tomcat using the migration utility included with NetWare 6. The Migration Utility creates Tomcat 3.3 Web applications from WebSphere Web applications.

The WebSphere-to-Tomcat Migration Utility is intended for use when upgrading from a NetWare 5.1 server to a NetWare 6.0 server where WebSphere 3.02 or WebSphere 3.5.1 has been already installed on the NetWare 5.1 server.

Step 1: Before Installing NetWare 6

1. At the NetWare console prompt, enter

   `xmlconfig -export volume:\websphere\migrate.xml -adminNodeName NodeName`

   If WebSphere was installed to another volume or directory other than *volume*:\WEBSPHERE, then specify that location instead.

   IMPORTANT: You must use the node's correct name.

   This step can be skipped if you are migrating from WebSphere Servlet Engine Only mode for WebSphere version 3.5.1.

2. Open the MIGRATE.XML file found in the *volume*:\WEBSPHERE directory to verify that the export was successful.

   Your Web applications should be listed in the MIGRATE.XML file.

Step 2: Upgrading to NetWare 6

Once you have completed the first step, you can proceed with your NetWare 6 server upgrade. (See the *NetWare 6 Overview and Installation Guide*).

IMPORTANT: All of the WebSphere directories must be preserved on the disk or made available after the upgrade is performed. Failure to save the directories could result in deleting your Web applications if you remove NetWare partitions during NetWare 6 installation.

Step 3: Running the Migration Tool

Once the upgrade is successfully completed and your server is running, start the migration utility.

1. Because the migration utility edits the Enterprise Web Server's OBJ.CONF file, we recommend that you make a backup copy of the file before running the utility.
2. If WebSphere was not installed to the default directory (SYS:\WEBSPHERE), edit the MIGRATE_TO_TOMCAT.NCF file with the correct path.
3. At the NetWare console prompt, enter

   `migrate_to_tomcat`

The migration utility creates a WEB-INF directory in the document root directory for each WebSphere Web application. In each of the WEB_INF directories is a WEB.XML file and two additional subdirectories named CLASSES and LIB.

The JAR files for the Web applications are copied into the LIB directory and the files that are referenced by the application's class-paths and the WebSphere system class-path are copied into the classes directory.

Classes and Jar files for your Web applications are then stored in these directories and the original locations are no longer used.

Additionally, the migration tool adds URL path references to the Enterprise Web Server OBJ.CONF file. The DBSWITCH.CONF, file found at *volume*:\NOVONYX\SUITSPOT\USERDB\DBSWITCH.CONF, is also edited.

For additional details, refer to the Migration Utility release notes found at the root of the NetWare 6 Operating System CD in the TOMCAT\33\BIN directory.

Undoing the Migration

If you change your mind, you can undo the migration by following a few simple steps.

1. Delete the file *volume*:\TOMCAT\33\CONF\APPS-WEBSPHERE.XML.
2. Remove the URL paths from the Enterprise Web Server's OBJ.CONF file.
3. Delete the WEB-INF directory in the document root of each WebSphere Web application that was migrated.
4. Remove the entries in DBSWITCH.CONF that point to the added document root directories.

10 Monitoring the Web Server

You can monitor your Web server's activity using one of several methods. You can view the server's status in real time—what is happening while you view it, compared to past performance—by using the Hypertext Transfer Protocol (HTTP) or the Simple Network Management Protocol (SNMP). You can also monitor your server by recording and viewing log files.

Working with Log Files

Server log files record your server's activity. You can use these logs to monitor your server and to help you when troubleshooting. Both the error log file and the access log file are located in /NOVONYX/SUITESPOT/HTTPS-*SERVERNAME*/LOGS. The error log file lists all the errors the server has encountered, and the access log file records information about requests to the server and the responses from the server. You can use the Server Status form to specify what to include in the access log file. Use the log analyzer to generate server statistics. You can back up server error and access log files by archiving them.

Viewing an Access Log File

You can view the server's active and archived access log files from the Server Status form.

1. From the Web Manager home page, click Enterprise Web Server *servername* > Server Status > View Access Log.
2. Select the access log file you want to see from the View This Log File drop-down list.

   Active log files for resources and archived log files appear in the list.

3. To limit how much of the access log you see, enter the number of lines you want to see in the Number of Entries field.

   The order of the log entries on the screen is the order in which they were recorded in the log.

4. If want to filter the access log entries for a particular word, enter the word in the Only Show Entries With field.

   Case is important; make sure the case for your entry matches the case of the word you're searching for.

5. Click OK.

Here is an example of an access log in the common logfile format:

```
wiley.a.com - - [16/Feb/1996:21:18:26 -0800] "GET / HTTP/1.0" 200 751
wiley.a.com - - [17/Feb/1996:1:04:38 -0800] "GET /docs/grafx/icon.gif HTTP/1.0"
204 342
wiley.a.com - - [20/Feb/1996:4:36:53 -0800] "GET /help HTTP/1.0" 401 571
arrow.a.com - john [29/Mar/1996:4:36:53 -0800] "GET /help HTTP/1.0" 401 571
```

The following table describes the last line of the sample access log.

Table 5 Fields in the Last Line of the Sample Access Log File

Access Log Field	Example
Hostname or IP address of client	arrow.a.com (In this case, the hostname is shown because the Web server's setting for DNS lookups is enabled; if DNS lookups were disabled, the client's IP address would appear.)
RFC 931 information	- (RFC 931 identity not implemented.)
Username	john (Username entered by the client for authentication.)
Date/time of request	29/Mar/1996:4:36:53 -0800
Request	GET/help
Protocol	HTTP/1.0
Status code	401
Bytes transferred	571

Here is an example of an access log using the flexible logging format:

```
wiley.a.com - - [25/Mar/1996:12:55:26 -0800] "GET /index.htm HTTP/1.0" "GET" "/
?-" "HTTP/   1.0" 304 0 - Mozilla/2.0 (WinNT; I)
wiley.a.com - - [25/Mar/1996:12:55:26 -0800] "GET / HTTP/1.0" "GET" "/?-" "HTTP/
1.0" 304 0    - Mozilla/2.0 (WinNT; I)
wiley.a.com - - [25/Mar/1996:12:55:26 -0800] "GET / HTTP/1.0" "GET" "/?-" "HTTP/
1.0" 304 0    - Mozilla/2.0 (X11; I; IRIX 5.3 IP22)
```

The access log in the flexible logging format looks similar to the access log using the common logfile format.

Viewing an Error Log File

The error log file contains errors the server has encountered after the log file was created; it also contains information about the server, such as when the server was started. Incorrect user authentication is also recorded in the error log. Use the error log to find broken URL paths or missing files.

1. From the Web Manager home page, click Enterprise Web Server *servername* > Server Status > View Error Log.
2. Enter the number of lines you'd like to see in the Number of Errors to View field if you want to see more or less than 25 lines of the error log at one time.

   The order of the log entries on the screen is the order in which they were recorded in the log.
3. If you'd like to filter the error messages for a particular word, enter the word in the Only Show Entries With field.

   Case is important; make sure the case for your entry matches the case of the word you're searching for.
4. Click OK.

Here is an example of an error log:

```
[13/Feb/1996:16:56:51] info: successful server startup
[20/Mar/1996 19:08:52] warning: for host wiley.a.com trying to GET /report.html,
append-trailer reports: error opening
[30/Mar/1996 15:05:43] security: for host arrow.a.com trying to GET /, basic-
ncsa reports: user jane password did not match database
```

In this example, the first line is an informational message—the server started successfully. The second log entry shows that the client wiley.a.com requested the file REPORT.HTML, but the file wasn't in the primary document directory on the server. The third log entry shows that the password entered for the user jane was incorrect.

Setting Log Preferences

You can customize access logging for any resource by specifying whether to log accesses, which format to use for logging, and whether the server should spend time looking up the domain names of clients when they access a resource.

Server access logs can be in common logfile format, flexible log format, or your own customized format. The Common Logfile Format is a commonly supported format that provides a fixed amount of information about the server. The flexible log format allows you to choose (from the Server Status form) what to log. A customized format uses parameter blocks that you specify to control what gets logged. Once an access log for a resource has been created, you can't change its format unless you archive it or create a new access log file for the resource.

To set access logging preferences, do the following:

1. From the Web Manager home page, click Enterprise Web Server *servername* > Server Status > Log Preferences.
2. Select the resource that you would like to apply custom logging to from the Editing drop-down list.
3. Select whether to log client accesses.
4. Enter the full path in the Log File field.

   By default, the log files are kept in the logs directory in the server root directory. If you specify a partial pathname, the server assumes the path is the logs directory in the server root.
5. Under Record, click Domain Names or IP Addresses.

6 In the Format list, select Common Logfile Format, flexible log format (Only Log option), or Custom Format.

7 (Conditional) If you selected Only Log, you can select any or all of the following flexible log format items in the checklist:

- Client Host Name: The hostname (or IP address if DNS is disabled) of the client requesting access.
- Authenticate Username: The authenticated username is listed in the access log if authentication is necessary.
- System Date: The date and time of the client request.
- Full Request: The exact request the client made.
- Status: The status code the server returned to the client.
- Content Length: The content length, in bytes, of the document sent to the client.
- HTTP Header, "Referer": The referer specifies the page from which the client accessed the current page. For example, if a user was looking at the results from a text search query, the referer would be the page that the user accessed the text search engine from. Referers allow the server to create a list of backtracked links.
- HTTP Header, "User-Agent": The user-agent information—which includes the type of browser the client is using, its version, and the operating system it's running on—comes from the User-Agent field in the HTTP header information the client sends to the server.
- Method: The request method used.
- URI: Universal Resource Identifier. The location of a resource on the server. For example, for http://www.a.com:8080/special/docs, the URI is special/docs.
- Query String of the URI: Anything after the question mark in a URI. For example, for http://www.a.com:8080/special/docs?*find_this*, the query string of the URI is *find_this*.
- Protocol: The transport protocol and version used.

8. (Conditional) If you selected a custom format, enter your custom format in the Custom format field.
9. If you don't want to log client access from certain hostnames or IP addresses, enter the hostname or IP address in the Hostnames and IP Addresses fields.

   Enter a wildcard pattern of hosts that the server should ignore when recording accesses. For example, use *.netscape.com if you don't want to log accesses from people whose domain is netscape.com; you can enter wildcard patterns for hostnames, IP addresses, or both.
10. Click OK.

Archiving Log Files

You can archive the access and error log files and have the server create new ones.

When you archive log files, the server renames the current log files and then creates new log files with the original names. You can back up or archive, or delete, the old log files, which are saved as the original filename followed by the date and time the file was rotated. For example, ACCESS might become ACCESS.24APR-04AM

You can archive log files immediately or have the server archive log files at a specific time on specific days. This information is stored in /NOVONYX/SUITESPOT/https-*servername*/LOGS.

Before running the log analyzer, you should archive the server logs.

1. From the Web Manager home page, click Enterprise Web Server *servername* > Server Status > Archive Log.

   The Archive Log Files form appears.
2. Do one of the following:
   - To rotate the log files immediately, click Archive.
   - To archive at specific times on specific days, click Rotate Log At, select a time from the drop-down menu, and check the days for archiving to occur.
3. Click OK.

Monitoring Current Web Server Activity

You can monitor your server's usage with the Monitor Current Activity page. You can see how many requests your server is handling and how it is handling these requests. If the interactive server monitor reports that the server is handling a large number of requests, you might need to adjust the server configuration or the system's network kernel to accommodate the requests.

1. From the Web Manager home page, click Enterprise Web Server *servername* > Server Status > Monitor Current Activity.
2. Click Monitor Server Activity on Port *port_number*.

The interactive server monitor reports the totals for the following server values on a new screen:

- Bytes Transferred: Number of bytes the server is transferring
- Total Requests: Number of requests the server is handling
- Bad Requests: Number of bad requests the server is handling
- 2*xx*: Number of status codes ranging from 200 to 299 that the server is handling
- 3*xx*: Number of status codes ranging from 300 to 399 that the server is handling
- 4*xx*: Number of status codes ranging from 400 to 499 that the server is handling
- 5*xx:* Number of status codes of 500 and higher that the server is handling
- *xxx*: Total number of 2*xx*, 3*xx*, 4*xx*, and 5*xx* status codes the server is handling minus time-outs and other errors that returned an HTTP status code
- 200: Number of successful transactions the server is processing
- 302: Number of relocated URL status codes the server is processing
- 304: Number of requests for which the server tells the client to use a local copy of a URL instead of retrieving a newer version from the server
- 401: Number of unauthorized requests the server is handling
- 403: Number of forbidden URL status codes the server is handling

Working with the Log Analyzer

Use the log analyzer to generate statistics about your server, such as a summary of activity, most commonly accessed URLs, times during the day when the server is accessed most frequently, etc. You can run the log analyzer from the Server Status form or the command line.

Before running the log analyzer, you should archive the server logs. For more information about archiving server logs, see "Monitoring the Server Using SNMP" on page 125.

Running the Log Analyzer from the Server Status Form

1. From the Web Manager home page, click Enterprise Web Server *servername* > Server Status > Generate Report.
2. Enter the name of your server in the Server Name field.

   This name appears in the generated report.
3. Select the output type—whether the report will appear in HTML or plain text format.
4. Select the log file you want to analyze.
5. If you want to save the results in a file, enter an output filename in the Output File field.

   If you leave the field blank, the analyzer prints results on the screen. For large log files, you should save the results to a file because printing the output to the screen might take a long time.
6. Select whether to generate the following server statistics:
   - Total Hits: Total number of hits the server received after access logging was enabled
   - 304 (Not Modified) Status Codes: Number of times the requesting client used a local copy of the requested document rather than retrieving it from the server
   - 302 (Redirects) Status Codes: Number of times the server redirected to a new URL because the original URL moved
   - 404 (Not Found) Status Codes: Number of times the server couldn't find the requested document or the server didn't serve the document because the client was not an authorized user

- 500 (Server Error) Status Codes: Number of times a server-related error occurred
- Total Unique URLs: Number of unique URLs accessed after access logging was enabled
- Total Unique Hosts: Number of unique client hosts who have accessed the server after access logging was enabled
- Total Kilobytes Transferred: Number of kilobytes the server transferred after access logging was enabled

7 Select whether to generate the following statistics:

- Top Number of One-Second Periods: Number of one-second periods during which requests were highest
- Top Number of One-Minute Periods: Number of one-minute periods during which requests were highest
- Top Number of One-Hour Periods: Number of one-hour periods during which requests were highest
- Top Number of Users: Number of users that accessed your server, provided that you included this as an item to log when you enabled access logging
- Top Number of Referers: Number of referers that appear in your log analysis, provided that you included this as an item to log when you enabled access logging
- Top Number of User Agents: Number of user agents that appear in your log analysis, provided that you included this as an item to log when you enabled access logging
- Top Number of Miscellaneous Logged Items: Number of miscellaneous logged items (request method, the URI, and the URI query) that appear in your log analysis (provided that you included this as an item to log when you enabled access logging)

To enable access logging, see "Setting Log Preferences" on page 118.

8. Select whether to generate a list of the following server access statistics:
   - Most Commonly Accessed URLs: The most commonly accessed URLs or URLs that were accessed more than a specified number of times
   - Hosts Most Often Accessing Your Server: The hosts most often accessing your server or hosts that have accessed your server more than a specified number of times
9. Enter the order in which you want to see the results in the Output order field.
10. Click OK.

Running the Log Analyzer from the Command Line

To analyze access log files from the command line, run the FLEXANLG tool, which is in EXTRAS/FLEXANLG in your server root directory.

To run FLEXANLG, enter the following command and options at the command prompt:

```
flexanlg [ -P ] [-n name] [-x] [-r] [-p order] [-i file]* [ -
m metafile ]* [ o file][ c opts] [-t opts] [-l opts]
```

The following describes the syntax. (You can get this information online by entering **`flexanlg -h`** at the command prompt.)

```
-P: proxy log format                                    Default: no
-n servername: The name of the server
-x : Output in HTML                                     Default: no
-r : Resolve IP addresses to hostnames                  Default: no
-p [c,t,l]: Output order (counts, time stats, lists)    Default: ctl
-i filename: Input log file(s)                          Default: none
-o filename: Output log file                            Default: stdout
-m filename: Meta file(s)                               Default: none
-c [h,n,r,f,e,u,o,k,c,z]: Count these item(s) -          Default: hnreuokc
    h: total hits
    n: 304 Not Modified status codes (Use Local Copy)
    r: 302 Found status codes (Redirects)
    f: 404 Not Found status codes (Document Not Found)
    e: 500 Server Error status codes (Misconfiguration)
    u: total unique URL's
    o: total unique hosts
    k: total kilobytes transferred
    c: total kilobytes saved by caches
    z: Do not count any items.
-t [sx,mx,hx, xx,z]: Find general stats - Default:s5m5h24x10
    s(number): Find top (number) seconds of log
```

```
    m(number): Find top (number) minutes of log
    h(number): Find top (number) hours of log
    u(number): Find top (number) users of log
    a(number): Find top (number) user agents of log
    r(number): Find top (number) referers of log
    x(number): Find top (number) for miscellaneous keywords
    z: Do not find any general stats.
-l [cx,hx]: Make a list of -                          Default: c+3h5
    c(x,+x): Most commonly accessed URLs
             (x: Only list x entries)
             (+x: Only list if accessed more than x times)
    h(x,+x): Hosts (or IP addresses) most often accessing your server
             (x: Only list x entries)
             (+x: Only list if accessed more than x times)
    z: Do not make any lists
```

Monitoring the Server Using SNMP

You can monitor your server in real time by using the Simple Network Management Protocol (SNMP). SNMP is a protocol used to exchange data about network activity. With SNMP, data travels between a managed device and a network management station (NMS) where users remotely manage the network.

A managed device is anything that runs SNMP (for example, hosts or routers). Your Novell® Enterprise Web Server is a managed device. An NMS is usually a powerful workstation with one or more network management applications installed. A network management application graphically shows information about managed devices (which device is up or down, which and how many error messages were received, etc.).

Every managed device contains an SNMP agent that gathers information regarding the network activity of the device. This agent is known as the *subagent*. Each Web server has a subagent.

Another SNMP agent exchanges information between the subagent and NMS. This agent is called the master agent. A *master agent* runs on the same host computer as the subagents to which it talks. You can have multiple subagents installed on a host machine. All of these subagents can communicate with the master agent.

Values for various variables that can be queried are kept on the managed device and reported to the NMS as necessary. Each variable is known as a *managed object*, which is anything the agent can access and send to the NMS. All managed objects are defined in a management information base (MIB), which is a database with a tree-like hierarchy.

How SNMP Works

SNMP exchanges network information in the form of protocol data units (PDUs). PDUs contain information about various variables stored on the managed device. These variables, also known as managed objects, have values and titles that are reported to the NMS as necessary. Communication between an NMS and managed device can take place in one of two forms: NMS-initiated and managed-device-initiated.

NMS-Initiated Communication

NMS-initiated communication is the most common type of communication between an NMS and a managed device. In this type of communication, the NMS either requests information from the managed device or changes the value of a variable stored on the managed device.

The following steps make up an NMS-initiated SNMP session:

1. The NMS searches the server's MIB to determine which managed devices and objects need to be monitored.
2. The NMS sends a PDU to the managed device's subagent through the master agent. This PDU either requests information from the managed device or tells the subagent to change the values for variables stored on the managed device.
3. The subagent for the managed device receives the PDU from the master agent.
4. If the PDU from the NMS is a request for information about variables, the subagent gives information to the master agent and the master agent sends it back to the NMS in the form of another PDU. The NMS then displays the information textually or graphically.

   If the PDU from the NMS requests that the subagent set variable values, the subagent sets these values.

Managed-Device-Initiated Communication

This type of communication occurs when the managed device needs to inform the NMS of an event that has occurred. A managed device such as a terminal would initiate communication with an NMS to inform the NMS of a shutdown or startup. Communication initiated by a managed device is also known as a *trap*.

The following steps make up a managed-device-initiated SNMP session:

1. An event occurs on the managed device.
2. The subagent informs the master agent of the event.
3. The master agent sends a PDU to the NMS to inform the NMS of the event.
4. The NMS displays the information textually or graphically.

The Enterprise Web Server MIB

Each Enterprise Web Server has its own management information base (MIB). The Enterprise Web Server's MIB is a file called HTTP.MIB, which contains the definitions for various variables pertaining to network management for the Enterprise Web Server. These variables are known as managed objects. Using the Enterprise Web Server MIB and network management software, such as HP* OpenView*, you can monitor your Web server like all other devices on your network.

The Enterprise Web Server MIB has an object identifier of netscape 1 (http OBJECT IDENTIFIER : := { netscape 1 }) and is located in the *server_root*\PLUGINS\SNMP\MIBFILES\NETWARE directory.

You can view administrative information about your Web server and monitor the server in real time using the Enterprise Web Server MIB. The following table lists and describes the managed objects stored in the HTTP.MIB.

Table 6 HTTP.MIB Managed Objects and Description

Managed Object	Description
httpEntityDescr	Description of the server (includes operating system information)
httpEntityId	Enterprise subtree for vendors (for example, the MIB has an object identifier of 1.3.6.1.4.1.1450)
httpEntityProtocol	HTTP version number
httpEntityVersion	Server software version number
httpEntityOrganization	Organization responsible for the server
httpEntityLocation	Full path for the server
httpEntityContact	People responsible for the server and contact information

Managed Object	Description
httpEntityAddress	IP address of the machine the server is running on
httpEntityPort	Port number that the server is listening on
httpEntityName	Server's identifier name (for example, server2.a.com)
httpEntityType	Type of server
httpEntityMethods	Methods supported by the server (for example, GET, POST, PUT)
httpEntityMaxProcess	Maximum number of active processes on the server
httpEntityMinProcess	Minimum number of active processes on the server
httpEntityMaxThread	Maximum number of active threads on the server
httpEntityMinThread	Minimum number of active threads on the server
httpStatisticsPort	Port number that this server is listening on
httpStatisticsAddress	IP address that this server is bound to
httpStatisticsStatus	Server status (up or down)
httpStatisticsNum ProcessIdle	Number of idle threads
httpStatisticsNum ProcessProc	Number of threads that are processing requests
httpStatisticsNum ProcessDns	Number of threads resolving hostnames
httpStatisticsRequests	Number of requests received and generated
httpStatisticsRequest Error	Number of request errors detected
httpStatisticsIn Unknowns	Number of unknown messages received/generated
httpStatisticsInBytes	Number of bytes received
httpStatisticsOutBytes	Number of bytes sent by the server
httpStatisticsTimeOut	Number of times the server timed out

Managed Object	Description
httpStatisticsProcess Num	Number of running processes
httpStatisticsThreadNum	Number of threads running
httpStatisticsNumBytes	Number of bytes sent by the server
httpStatisticsNum2xx	Number of 200-level status requests handled by the server
httpStatisticsNum3xx	Number of 300-level status requests handled by the server
httpStatisticsNum4xx	Number of 400-level status requests handled by the server
httpStatisticsNum5xx	Number of 500-level status requests handled by the server
httpStatisticsNum200	Number of 200 (Transfer OK) requests
httpStatisticsNum302	Number of 302 (Moved Temporarily) requests
httpStatisticsNum304	Number of 304 (Not Modified) requests
httpStatisticsNum401	Number of 401 (Unauthorized) requests
httpStatisticsNum403	Number of 403 (Forbidden) requests

For Additional Information

For additional information about the Enterprise Web Server, visit Netscape DevEdge* (http://developer.netscape.com/docs/manuals/doclist.html).

III Introducing NetWare Web Search Server

Make your Web or file data searchable in minutes! From simple search solutions to complex, revenue-generating search services, NetWare® Web Search bridges all types of networks—from file servers, to intranets, extranets, and the Internet—by bringing critical information to busy people. It is one of the industry's fastest and most accurate search engines available today.

Upon completing this section, you should know what Web Search can do for you and how to customize all of the search templates and plan and deploy enterprise-wide search solutions and services.

11 Introducing NetWare Web Search Server

NetWare® Web Search Server offers a powerful full-text search engine you can use to add search capabilities to your Internet or intranet Web sites. Compatible with the both NetWare Enterprise and Apache Web servers, you can create custom search forms and search result pages either from scratch or by using the templates provided with NetWare Web Search Server.

With NetWare Web Search Server, you can

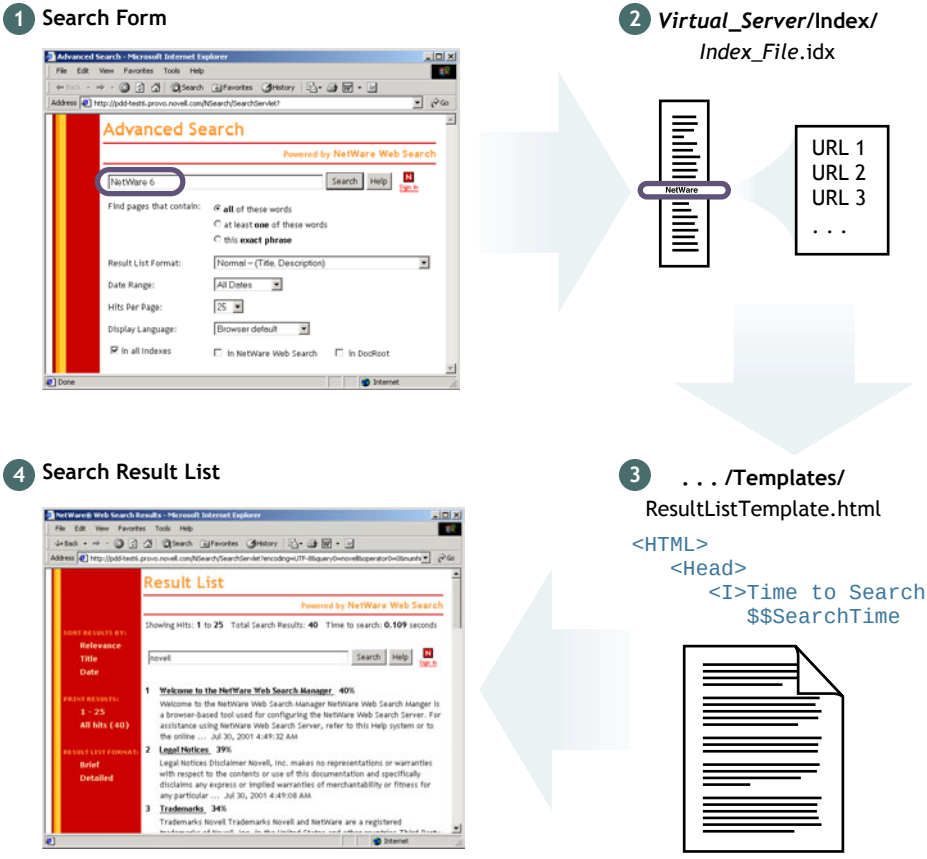
- Search across multiple Web sites, servers, and file formats in any language, all from a single interface.
- Host search services for one or more companies or organizations.
- Print large collections of dispersed but related files as a single, coherently organized document.
- Customize the look and feel of search and print results in all languages.
- Create themes, which are defined collections of search and print result templates that allow you to deploy custom look and feel virtual search servers, each for a specific company or department.

How NetWare Web Search Works

Understanding how Web Search handles searches can help explain the role of templates, template variables, and query parameters in Web Search. One of the great benefits of Web Search is the simplicity of customizing it to meet your needs or the needs of your clients.

The following figure shows what happens when users submit words through the search template and how Web Search then handles the words to generate and display search results.

Figure 4 How NetWare Web Search Handles a Search String



In this diagram, the user (1) enters a search string such as **`NetWare 6`**. The search string is then searched for in the index file on the Web Search Server (2). If the search string is located, the Uniform Resource Locators (URLs) and document titles and descriptions are passed on to the search results template (3) and displayed to the user (4). The information that is displayed is determined by the variables included in the search results template, which means that you can modify what information is actually returned to the user by adding or removing Web Search variables from the template.

Components of a Virtual Search Server

Similar in purpose to a software virtual server, a *virtual search server* is a way to group similar information together for a specific purpose and audience. For example, you might create one virtual search server for your company's support organization, another for it's public Web site, and yet another for your intranet. You might break these down even further by creating more focused virtual search servers for groups within these organizations.

Literally, a virtual search server is a collection of HTML templates and supporting data files and consists of the following components:

- A name and (optional) alias for accessing a virtual search server
- Index files containing key words and related URLs for use in search results
- Scheduled indexing events
- Search and print results templates
- Log files

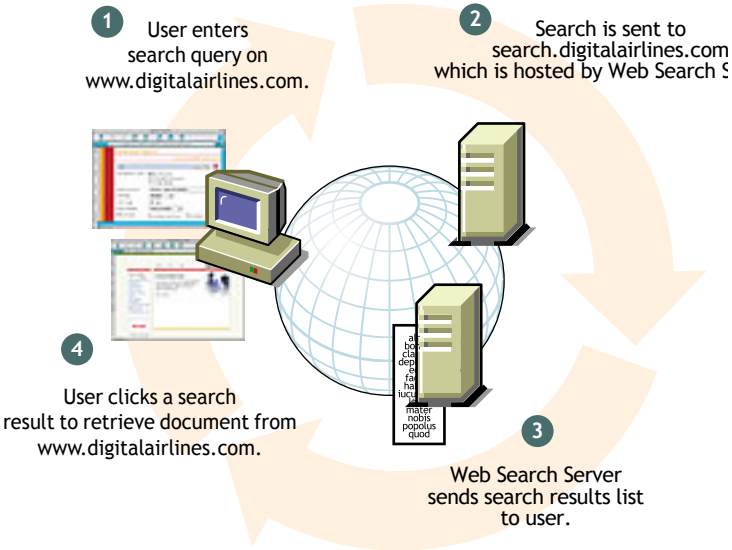
Each of these components are managed through the NetWare Web Search Manager, which is accessed using a Web browser (see Chapter 14, "Creating and Managing Virtual Search Servers," on page 149).

HINT: If you want two or more virtual search servers to share an index, create a duplicate index on each virtual search server that points to the same index directory. In this manner, all virtual search servers can search a shared index in addition to their own indexes.

General Architecture of a Web Search Service

The general architecture of a Web search service is depicted in the following diagram.

Figure 5 Search Service Architecture



In Figure 5, the user enters a search query in a search form found on www.digitalairlines.com (1 and 2). When the user clicks the Search button, the query is sent to search.digitalairlines.com, which is hosted by the NetWare Web Search Server (3), which then processes the query using index files that were created using the Web Search Manager.

Web Search then compiles the results of the search into an HTML template, which has been modified to match the look and feel of www.digitalairlines.com, and returns them to the user's Web browser (4).

When users click a search result link to view the content, they are taken to that content, whether it is hosted on www.digitalairlines.com or some other Web site.

Building a Virtual Search Server

Using NetWare Web Search Manager, you define a new virtual search server and then index data found on your NetWare server or content found on any Web site that can then be searched using the NetWare Web Search Server.

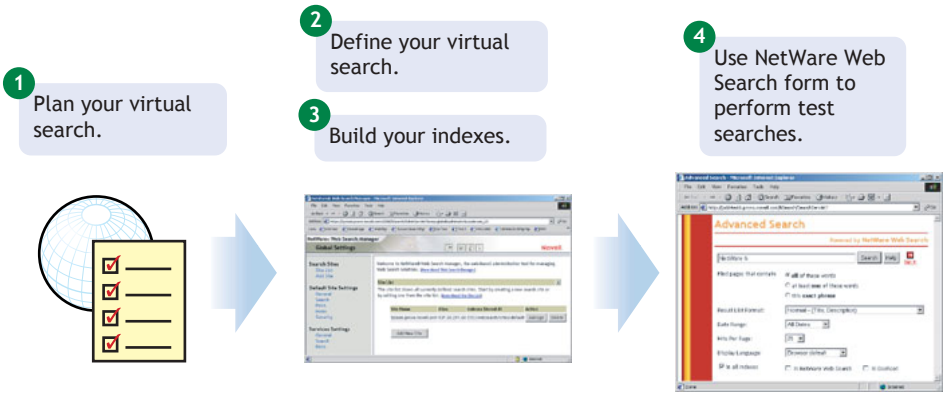
Building a virtual search server involves four fundamental tasks:

1. Plan the purpose of your virtual search server, which includes identifying what will be searched and how to optimize your search solution based on the content to be searched.

2. Defining a new virtual search server using Web Search Manager.
3. Building one or more indexes of your Web sites and file servers.
4. Testing your new virtual search server using the default search form and the search and print result templates.

After completing the first task of planning your virtual search server (see Chapter 13, "Designing Your Search Solution," on page 143), use the NetWare Web Search Manager to complete the remaining tasks. The following figure illustrates these tasks.

Figure 6 Steps to Creating a Virtual Search Server



Repeat this process for each new virtual search server you want to add. A virtual search server can include one or more indexes of files located on your file server and files located in one or more directories on one or more Web servers.

Indexing a Web server (or Web site) involves a process known as *crawling*. The Web Search Server begins indexing files on a Web server at the directory level you specify and continues to index along hypertext links until reaching a dead-end, which occurs when either a linked file cannot be found or when there are no more links found within the specified Web sites.

Accessing NetWare Web Search Manager

NetWare Web Search Manager is the tool you use to create and manage virtual search servers and their associated indexes.

To run NetWare Web Search Manager, do the following:

1. Type **`https://domainname:portnumber`** in your Web browser's address field, where *portnumber* is the port number of NetWare Web Manager and press Enter.

   HINT: You must use the HTTPS protocol because NetWare Web Search Manager uses Secure Sockets Layer (SSL). You can disable encryption from the Admin Preferences page of NetWare Web Manager (see "Securing Web Manager" on page 35).

2. Under NetWare Web Search Server, click the *servername* link, where *servername* is the name of your NetWare Web Search Server.

Taking a Test Run

When you install NetWare Web Search Server, some of your server's content is automatically indexed and appears on the default search form as the "NetWare Web Search" and "Doc Root" indexes.

Once you start the Enterprise Web Server, you can open the search page using your Web browser and perform a search against the content that has been automatically indexed.

To test NetWare Web Search using the default search page, do the following:

1. Type **`http://domainname/novellsearch`** in your Web browser's address field and press Enter.

   IMPORTANT: The URL is case sensitive. Use the exact case shown above.

2. Type NetWare in the Search field > press Enter.

HINT:

The Search form template, SearchTemplate.html, is stored on your server at /*searchroot*/TEMPLATES. See "Customizing the Look and Feel of Search Server Content" on page 139 for information about how to customize templates.

Customizing the Look and Feel of Search Server Content

Once you've created one or more virtual search servers and built indexes for them, you can customize them by modifying the default search form and result and print templates, or by creating a new templates from scratch using the variables and parameters described in Chapter 18, "Working with Template Variables and Search Parameters," on page 195.

For more information about modifying the default search form and the search and print templates to create your own custom search solution, see Chapter 17, "Customizing Your Search Solutions," on page 189.

For more information about building professional corporate and hosted search services, see Chapter 13, "Designing Your Search Solution," on page 143.

12 What's New with the NetWare Web Search Server

Since it's release in NetWare® 5.1, NetWare Web Search Server has been significantly enhanced with new features and improved usability.

Most significant is the ability to host multiple virtual search servers on your server, allowing you to create revenue-generating search services or enterprise search solutions for multiple, independent organizations, departments, or companies (see Chapter 13, "Designing Your Search Solution," on page 143).

Search speed, added search parameters and variables for fine-tuning searches, and embedded documentation in the administrator's interface are just a few of the enhancements and new features included in NetWare Web Search Server.

The following list highlights key new features and enhancements to NetWare Web Search Server included with NetWare 6:

- Takes advantage of NetWare 6 multi-processor (MP) capabilities
- Faster indexing and searching, including faster multiword searches and the ability to crawl several indexes simultaneously
- Support for hosting professional search services by letting you create and manage multiple virtual search servers simultaneously
- Search and print results are automatically returned in whatever language users specify in their Web browser's lanaguage preferences settings
- Support for the NetWare Cluster environment, including cluster-safe and active-passive failover
- Support for The U.S. Federal Mandate for Accessibility (Section 508) of the United States Rehabilitation Act
- More powerful and easier-to-use Web Search Manager

- Increased control over the indexing process by letting you control what is indexed, including the ability to index only portions of documents by skipping repetitive sections such as headers, footers, and navigation bars
- Improved support for the following file formats: QuattroPro*, PowerPoint*, RTF, XML, HTML, PDF, TXT, and Word*
- Support for XML, including full-text, field, and XML heirarchy searches
- Search and print templates redesigned for an enhanced look and feel
- New debugging feature that logs all errors related to information that is sent or received by a Web Search service
- Date-based sorting that searches for results according to user-specified dates
- Enhanced search functions, including the ability to search for a specific filename, path, URL, or file extension, which can be used when you know the name or path to a specific file—or used to restrict or filter your searches
- Improved search operators and query parameters for use in generating more accurate search results

13 Designing Your Search Solution

Web Search is a search service solution designed for adding powerful search capabilities to individual Web sites. It is not intended to index the entire Internet. But because many Web sites are comprised of multiple Web servers located across an enterprise, Web Search was designed to be able to index hundreds, even thousands, of Web sites as part of a single search solution.

In addition to enterprise search solutions, Web Search can also be set up to host multiple, independent virtual search servers, all from a single NetWare 6 server. (See "Becoming a Search Service Host" on page 146.)

This chapter provides suggestions for creating a search solution that fits your particular needs and circumstances. If you are interested in hosting professional search services, you might want to read the section "Becoming a Search Service Host" on page 146.

Components of a Virtual Search Server

A virtual search server is a fully functioning, self-contained search service created for a particular audience, such as a department, organization, or a specific group of customers.

A virtual search server typically contains its own indexes, log files, administration interface, search and print templates, scheduled events, virtual search server name, and an optional alias.

Providing search services involves creating one or more *virtual search servers*.

NOTE: Users cannot search more than one virtual search server at a time. However, a virtual search server can contain indexes created from content on multiple Web sites or file servers.

Taking the time to plan your search service strategy can save you time and money and improve the quality of your service.

When you create a new virtual search server, you create an independent search service, meaning that it is self-contained and doesn't depend on, or interact with, other virtual search servers.

Each virtual search server that you create typically contains one or more of the following components:

- *Indexes*: Files that hold key words and associated URLs of Web sites or file server content that have been indexed, or *crawled*.
- *Themes*: When applied, a theme instantly adds a common look and feel to your search page, search and print results pages, and response and error message pages.
- *Search and Print Results Templates*: Templates that become populated with the results of a search and then are displayed to the user. Depending on which templates are used, the level of detail displayed in search and print results varies.
- *Scheduled Events*: Index management, such as updating or regenerating, can be automated to occur at specific intervals using the Scheduling feature.

Deciding If You Need More Than One Virtual Search Server

To determine if you need more than one virtual search server, answer the following questions:

- ❑ Do you want to host search services for multiple, independent organizations?
- ❑ Do you want to consolidate multiple NetWare 5.1 Web Search servers onto a single machine?
- ❑ Do you need to prevent users from being able to search across multiple indexes at the same time?

If you answered *yes* to any of these, you will likely need to create more than one virtual search server. For information about creating virtual search servers, see Chapter 14, "Creating and Managing Virtual Search Servers," on page 149.

Using Web Search in a Clustered Environment

Installing and configuring NetWare Web Search Server with Novell Cluster Services™ requires three primary steps:

1. Setting up your clustered environment, including any supporting Web technologies.
2. Installing Web Search Server to the same shared volume from each of your clustered servers.
3. Creating virtual search servers using the DNS name and IP address of your cluster server.

Step One: Setting Up Your Clustered Environment

Before NetWare Web Search Server will run in a clustered environment, you must install and configure the following components on a server in your cluster.

- ❑ Novell Cluster Services

  See "Installation and Setup" in *Novell Cluster Services Overview and Installation*.

- ❑ NetWare Enterprise Web Server

  See "Configuring NetWare Enterprise Web Server with Novell Cluster Services" in *Novell Cluster Services Overview and Installation*.

- ❑ Apache Web Server

  See "Apache with Novell Cluster Services" in *Novell Cluster Services Overview and Installation*.

- ❑ Tomcat Application Server

  See "Tomcat and Novell Cluster Services" in *Novell Cluster Services Overview and Installation*.

Step Two: Installing Web Search Server to a Shared Volume

Install Web Search Server (using the NetWare 6 Installation CD) to the same shared volume in your clustered environment. For information about setting up a shared volume in a clustered environment, see "Cluster Enable Pools and Volumes" in *Novell Cluster Services Overview and Installation*.

Installing Web Search Server to each server in your cluster allows the installation program to correctly register Web Search with each of your server's Web and application servers.

Installing it to the same shared volume centralizes all of the required templates, indexes, and configuration settings for use by all servers in your cluster. Each time you create a new virtual search server, default settings and templates are drawn from the shared volume. Future upgrades or patches to the Web Search Server need to be installed only to the shared volume and can be done from any server in your cluster.

Step Three: Creating Virtual Search Servers

You can now begin creating virtual search servers from any of the Web Search Servers in your cluster.

When prompted to provide a name and alias for your new virtual search server, enter the DNS name and IP address (alias) of your cluster server. The Path field can be left blank because it is already known by Web Search Server.

If You Are Not Using a Shared Volume

We recommend that you use a shared volume. However, if for some reason your clustered environment is not utilizing a shared volume, you can install Web Search Server to each of your servers and then use mirroring software to syncronize the data between your servers.

If you do this, each time you upgrade or apply a patch to the Web Search Server, you will need to repeat the upgrade installation or patch on each of the servers in your cluster.

For more information about clustering, see *Novell Cluster Services Overview and Installation*.

Becoming a Search Service Host

Companies who want customers to find information about their products outsource this functionality to search service companies to make their information searchable. Figure 5, "Search Service Architecture," on page 136 shows how a search query on the Web site www.digitalairlines.com is sent to a Web Search server on the domain search.digitalairlines.com.

With NetWare Web Search Server, you can offer professional search services for other companies. Using a single installation of Web Search, you can host several virtual search servers simultaneously, which means that you could use the same NetWare server to host search services for several client or customer Web sites.

Getting Started

Once you have thought out your search service strategy, you can begin creating and defining your virtual search servers by referring to Chapter 14, “Creating and Managing Virtual Search Servers,” on page 149.

To learn more about customizing your search service, start by reading Chapter 16, “Understanding Templates,” on page 183. If you are already familiar with Web Search and its search and print templates, you might want to skip to Chapter 18, “Working with Template Variables and Search Parameters,” on page 195.

14 Creating and Managing Virtual Search Servers

This chapter provides detailed information about how to create and manage virtual search servers using NetWare® Web Search Manager.

About Virtual Search Servers

By definition, a virtual search server is a collection of one or more indexes and their related configuration files. Indexes are at the heart of a virtual search server. An index is an optimized binary file that contains keywords found in documents hosted on a Web or file server. Indexes are used by Web Search to return search results to users' Web browsers.

Before creating virtual search servers, particularly large or mission-critical ones, you should carefully plan how to configure your search services. A virtual search server that will be used by a small to medium department in a company requires different planning than a search server that will serve thousands of customers on an Internet site.

For information about how to plan an effective search service, see Chapter 13, "Designing Your Search Solution," on page 143.

Creating a Virtual Search Server

Once you have carefully planned your search service, you can start creating and configuring virtual search servers and begin adding indexes to them.

1 From the Web Search Manager Global Settings page, click Add New Virtual Search Server.

2. In the Name field, enter a new virtual search server name, which is typically the DNS or domain name of your server.

   For more information about virtual search server names, see "Naming a Virtual Search Server" on page 150.

3. In the Alias field, enter a virtual search server alias, which is typically the IP address of your server.

   See "Using the Virtual Search Server Alias" on page 151 for more information about aliases.

4. In the Location field, enter the path to where you want the index and configuration files to be stored.

   HINT: If this field is left blank, Web Search will store the virtual search server files in the /*searchroot*/sites/*sitename* directory. Also, you can store the files on any volume on the server where Web Search is installed, but not on other servers.

5. Click Create.

Naming a Virtual Search Server

When a user sends a search query to the Web Search Server, Web Search must determine which of all of your virtual search servers it should use to handle the incoming search request.

Web Search uses two methods for determining this:

1. Matching the domain name of the search query with the virtual search server names available in Web Search
2. Using the SITE=*searchsitename* query parameter to find matching virtual search server names

For example, in the following search request, Web Search uses the domain name search.domainname1.com as the name of the virtual search server:

```
http://search.domainname1.com/NSearch/SearchServlet?query=find+something
```

This approach requires that your server be set up to recognize the domain name search.domainname1.com. Most servers can be set up to recognize and service multiple domain names in both software and hardware virtual server configurations (see "Setting Up Multiple Web Servers" on page 73).

You could also use an IP address to designate the virtual search server. For multiple virtual search servers, this approach would work only in a hardware virtual server configuration where each virtual search server has its own unique IP address.

Hosting Multiple Virtual Search Servers Using One DNS Name

If you are hosting a search service for two or more customers, you can name each virtual search server according to the organization or company name of each customer and then use the &server query parameter when handling search queries. One of the advantages of using the &site query parameter is that it allows you to use a single DNS name.

For example, suppose your server's URL was searchit.novell.com. If you were setting up search services for a company called Digital Airlines and another company called DemoCity, you could host both services on your single server and then simply include the &server=digitalairlines and &server=democity query parameters within the search forms found on www.digitalairlines.com and www.democity.com.

Queries would be sent from the search forms on each Web site to the URLs corresponding to each virtual search server, as in the following:

http://searchit.novell.com/NSearch/SearchServlet?server=digitalairlines

and

http://searchit.novell.com/NSearch/SearchServlet?server=democity

Using the Virtual Search Server Alias

When defining a virtual search server, you are required to give it a name. But Web Search administrators can also define an alias that can be used when identifying a specific virtual search server during a search request.

An alias name typically follows one of two conventions:

1. *An IP address*: This could be used either in the domain name portion of a URL or be included in a search query using the &server query parameter. Using an IP address in place of a domain name to select a virtual search server only works in a hardware virtual server configuration where each search server has its own unique IP address.
2. Any other numeric or textual value that can be passed as the value of the &server query parameter.

For most virtual search servers, the best choice for a search server name and alias is the Web server's domain name and IP address.

For more information about creating software and hardware virtual servers on the NetWare Enterprise Web Server, see "Setting Up Multiple Web Servers" on page 73.

Storing Virtual Search Server Files

Search server files include a set of index and configuration files for each virtual search server. When you create a new virtual search server, you can specify where you want virtual search server files to be stored, or you can accept the default path which is determined by where you installed the NetWare Web Search Server.

Virtual search server files can be stored on any volume visible to the NetWare server that Web Search is installed on, regardless of which volume your Web Search Server is installed on. This includes SAN storage device volumes.

Creating Indexes

Web Search creates two types of indexes:

- Crawled: Created as Web Search follows (or crawls) hypertext links until it reaches a dead end. Web Search can crawl one or more Web sites, specific areas of a Web site, or specific URLs, even down to a specific filename.
- File System: Created as Web Search indexes content on a file server. Web Search can index one or more paths on multiple volumes, including Storage Area Network (SAN) storage devices.

There are two forms you can use to create an each type of index: the standard form and the advanced form.

For example, the Define Crawled Index is the standard form for creating a crawled index. But the Define Crawled Index (Advanced) form offers more options than the standard form, including options that override default virtual search server settings. Both methods are described in the following sections.

Searching across Multiple Indexes

Web Search can search across multiple indexes within a single virtual search server. However, searching a single index is generally faster than searching across multiple indexes.

HINT: While you can search across multiple indexes within a single virtual search server, you cannot search across multiple virtual search servers.

Restricting Search Results to Specific Areas

You can restrict search results to specific areas of your file or Web server in the following ways:

- Using multiple indexes and using the &collection=index_name query parameter.
- Using a single index, restrict results to certain URL paths using the &filefilter=path query parameter.
- Using a single index, restrict results to certain values in document fields by including /fieldname=value with either the query=value or filter=value search parameters.

HINT: Using the last option requires that indexed documents contain summary fields such as META tags. This option works for almost any file format that contains document summary fields, including HTML, XML, PDF, Word*, and WordPerfect*.

For information about preventing Web Search from indexing specific content, see "Excluding Documents from Being Indexed" on page 173.

Defining a New Crawled Index

Using the Define Crawled Index Page

1. From the Web Search Manager Global Settings page, click Manage in the row of the virtual search server that you want to work with.
2. Under Define a New Index, click New Crawled Index > Define Index.
3. In the Index Name field, enter a name for your index.

   HINT: A name can be a word, phrase, or a numeric value. If the virtual search server you are working on contains, or will contain, a large number of indexes, you might want to utilize a numbering scheme to help you manage multiple indexes more effectively. But keep in mind that the name you enter here appears on the default search page. So you might want to choose a name that can be understood by users of your search services.
4. Under Web Sites to Crawl, type the URL of the Web site that you want indexed.

   You can enter just the URL, such as www.mycompany.com, or you can also append a complete path, down to the file level, such as www.mycompany.com/*path*/index.html.
5. If desired, add another URL.

6. To add additional URLs, click Add More URLs.
7. Click Apply Settings.

Using the Define Crawled Index (Advanced) Page

The Define Crawled Index (Advanced) page offers some additional options beyond those available in the standard Define Crawled Index page. Changes made using this page will override default virtual search server settings.

1. From the Web Search Manager Global Settings page, click Manage in the row of the virtual search server that you want to work with.
2. Under Define a New Index, click New Crawled Index > Define Index.
3. On the Define Crawled Index page, click Advanced Index Definition.
4. In the Index Name field, enter a name for your new index.

   HINT: A name can be a word, phrase, or a numeric value. If the virtual search server you are working on contains, or will contain, a large number of indexes, you might want to utilize a numbering scheme to help you manage multiple indexes more effectively. But keep in mind that the name you enter here appears on the default search page. So you might want to choose a name that can be understood by users of your search services.
5. In the Index Description field, enter an optional description of the index to be created.
6. Under Web Sites to Crawl, enter the URL of the Web site to be indexed.

   HINT: If you enter a filename at the end of the URL, then just that file will be indexed.
7. In the Subdirectories to Exclude text box, type the directories that you want Web Search not to index.

   For example, /marketing or /sales/doc.
8. To direct Web Search to include or exclude specific file types, click Extensions to Include or Extensions to Exclude and then enter the extensions, separating each one with a single space, such as HTM PDF TXT.
9. To add additional URLs, click Define More Web Sites.
10. To delete a URL, click Remove Web Site.
11. In the Additional URLs text box, enter any other URLs that you want indexed.

For example, www.mycompany.com/marketing.

This allows you to specify additional pockets of information found on other Web sites, but not include all of the content of those sites to your searches.

HINT: When Web Search encounters links found in the pages of Additional URLs that point to pages specified in Web Sites to Crawl, Web Search follows those links. All other links that go outside of Web Sites to Crawl are not followed.

12 Under Additional Settings, enter the absolute path to where you want the index files stored in the Location of Index Files field.

For example, *volume*:*searchroot*\sites\mysites.

By default, index files are stored at *volume*:*searchroot*\sites\default\indexes\.

HINT: Changes made to Additional Settings override Default Settings.

13 From the Encoding (If Not in META Tags) drop-down list, select the encoding to be used by files being indexed that do not contain an encoding specification.

14 In the Maximum File Size to Index field, enter the maximum file size (in bytes) that Web Search should index.

Files exceeding this size will not be indexed and therefore, will not be included in search results.

15 In the Maximum Time to Download a URL field, enter a number (in seconds) before Web Search automatically skips the indexing of the specified URL.

16 To direct Web Search to pay attention to case of filenames and directory names, click Yes next to URLs that are Case Sensitive.

17 To direct Web Search to crawl dynamic content (URLs containing the question mark [?]), click Yes next to Crawl Dynamic URLs.

HINT: For more information about indexing dynamic content, see "About Indexing Dynamic Web Content" on page 160.

Once you define an index, you must generate it to make it searchable. See "Generating Indexes" on page 158.

Defining a New File System Index

Using the Define File System Index Page

1 From the Web Search Manager Global Settings page, click Manage in the row of the virtual search server that you want to work with.

2 Under Define a New Index, click New File System Index > Define Index.

3 In the Index Name field, enter a name for your index.

HINT: A name can be a word, phrase, or a numeric value. If the virtual search server you are working on contains, or will contain, a large number of indexes, you might want to utilize a numbering scheme to help you manage multiple indexes more effectively. But keep in mind that the name you enter here appears on the default search page. So you might want to choose a name that can be understood by users of your search services.

4 In the Server Path to be Indexed field, enter the absolute path to the folder containing the information that you want indexed.

For example, SYS:\SALES\REPORTS.

5 In the Corresponding URL Prefix field, enter the URL that should be used by the search results page to access the individual files.

For example, /SALES.

HINT: For information about defining a URL prefix in the NetWare Enterprise Web Server, see "Setting Additional Document Directories" on page 64.

6 To add additional paths, click Add More Paths.

7 Click Apply Settings.

Once you define an index, you must generate it to make it searchable. See "Generating Indexes" on page 158.

Using the Define File System Index (Advanced) Page

1. From the Web Search Manager Global Settings page, click Manage in the row of the virtual search server that you want to work with.
2. Under Define a New Index, click New Crawled Index > Define Index.
3. On the Define File System Index page, click Advanced Index Definition.
4. In the Index Name field, enter a name for your new index.

   HINT: A name can be a word, phrase, or a numeric value. If the virtual search server you are working on contains, or will contain, a large number of indexes, you might want to utilize a numbering scheme to help you manage multiple indexes more effectively. But keep in mind that the name you enter here appears on the default search page. So you might want to choose a name that can be understood by users of your search services.
5. In the Index Description field, enter an optional description of the index to be created.
6. In the Location of Index Files field, enter the absolute path to where you want the index files stored.

   For example, SYS:\NSearch\sites\mysites.

   By default, index files are stored at *volume*:*searchroot*\sites*site_name*\indexes\.
7. From the Encoding (If Not in META Tags) drop-down list, select the encoding to be used when indexeing files that do not contain an encoding specification.

   For example, HTML files can specify their encoding with a Content-Type META tag.
8. In the Maximum File Size to Index field, enter the maximum file size (in bytes) that Web Search should index.

   Files exceeding this size will not be indexed and therefore, will not be included in search results.
9. Under Path Information, type the absolute path to the folder containing the information that you want indexed in the Server Path field. For example, SYS:\SALES\REPORTS.
10. In the Corresponding URL Prefix field, enter the URL that should be used by the search results page to access the individual files.

    For example, /SALES.

HINT: For information about defining a URL prefix in the NetWare Enterprise Web Server, see "Setting Additional Document Directories" on page 64.

11 To exclude specific subdirectories from being indexed, enter their relative paths in the Subdirectories to Exclude field.

12 To direct Web Search to include or exclude specific file types, click Extensions to Include or Extensions to Exclude and then type the extensions, separating each one with a single space, such as HTM PDF TXT.

13 To add additional paths, click Define More Paths.

14 To delete a path, click Remove Path.

15 Click Apply Settings.

Once you define an index, you must generate it to make it searchable. See "Generating Indexes" on page 158.

Generating Indexes

Once you define an index, you must generate it before it can be used for searching. Generating an index is the actual process where Web Search Server examines file server or Web server content, gathers keywords, titles, and descriptions and then includes them in the index.

1 From the Web Search Manager Global Settings page, click Manage in the row of the virtual search server that you want to work with.

2 Click Generate in the Action column of the index that you want to work with.

The Active Jobs screen indicates the status of the current indexing jobs. When there is no current index job, the status page will read `No indexing jobs are currently running or defined`.

3 To cancel the current indexing jobs, click Cancel in the Status column.

You can direct Web Search to automatically update your indexes on specific dates and at specific times by scheduling events. For more information, see "Automating Index and Server Maintenance" on page 160.

Managing Existing Index Files

Once created, an index can then be edited or deleted. You can also view an index's log file. (See "Working with the Log File" on page 159)

Editing an Index

1. From the indexing Management page, click Edit in the Action column of the index you want to work with.
2. Make any of the changes you need to and then click Apply Settings.

   HINT: If you used the Advanced page to create the index, it will appear automatically. However, you can also click Advanced Index Definition to make advanced changes to an index you created using the standard Index Definition page.
3. If you added new paths or URLs or modified any of the existing ones, you should regenerate the index to include the new content.

Deleting an Index

1. From the indexing Management page, click Delete in the Action column of the index you want to delete.
2. In the Confirm Deletion of *indexname* page, click Delete Index to proceed, or click Cancel Deletion.

   WARNING: Once an index has been deleted, it cannot be restored. You must generate a new index.

Working with the Log File

The purpose of the log file is to help you identify any errors (and their possible causes) in performance during an indexing job.

In addition to reporting when the indexing job started and stopped, it also lists all files that were indexed, files that could not be found but were linked to, and even errors that might have occurred during the indexing process.

To view an index's log file, do the following:

1. Click View Log in the Action column of the index that you want to work with.
2. Review the contents of the log file and then either click your browser's Back button to return to the indexing Management page, or click Management in the left frame of the Web Search Manager.

About Indexing Dynamic Web Content

Much of the content on the World Wide Web is static HTML, which means that after a static Web page is created, it remains the same until someone updates it. By contrast, many newer Web pages are created by Web applications, including servlets, Java Server Pages (JSP), Common Gateway Interfaces (CGI), and Pearl Scripts, and are usually created in response to user input.

An example of dynamic Web content is an eCommerce Web page where items to be purchased are stored in a virtual shopping cart, the total cost is updated as users add or remove items from their shopping cart.

Because the content changes regularly, many search engines don't index dynamic content.

NetWare Web Search includes the ability to index dynamic content. The URL of dynamic Web content typically includes a question mark (?). You can direct Web Search to index these URLs by setting the Crawl Dynamic URLs option to Yes. You could then create a scheduled event that regenerates the specified indexes every few minutes.

Automating Index and Server Maintenance

You can eliminate a lot of manual work in keeping indexes up to date by using Web Search's index scheduling feature. Because the Web and file content you have indexed will eventually change, you can direct Web Search to update your indexes on specific dates and at specific times or intervals.

Adding a Scheduled Event

1. After selecting a search server from the Virtual Search Server List, click Scheduling in the left frame of Web Search Manager.
2. Click Add Event.

3. Specify the month, days, days of the week, or time (in hours and minutes) when you want Web Search to run the event.

   HINT: To select multiple dates and times, hold down the Ctrl key and click all of the items you want added. To select consecutive items, click the first item and then hold down the Shift key and click the last item.

4. Select the type of operation you want performed on your indexes.

   - Update: Web Search identifies new content on Web or file servers and updates the index.
   - Optimize: Web Search improves searching performance by removing unnecessary content and making the index file more compact.
   - Regenerate: Web Search replaces the existing index with a newly generated one.

5. In the Perform Operations On column, determine whether you want the chosen operation performed on all indexes or only on specified ones.

   HINT: If you have large indexes, you might consider creating multiple events that update your indexes at varied times. Doing so will minimize CPU utilization. By default, Web Search supports up to 5 simultaneous indexing sessions. All other indexes will wait until a previous index job has completed. You can control the number of simultaneous indexing jobs from Services Settings (see "General Services Settings" on page 168).

6. Click Apply Settings.

Editng or Deleting an Event

1. After selecting a virtual search server from the Virtual Search Server List, click Scheduling in the left frame of Web Search Manager.

   HINT: If no events have been scheduled, refer to the procedure above for adding a scheduled event.

2. To edit a scheduled event, click Edit in the row of the event you want to modify.
3. Make the desired changes and click Apply Settings.
4. To delete a scheduled event, click Delete in the row of the event you want to delete.
5. Click Delete Event to confirm the deletion, or click Cancel Deletion.

Modifying Default Virtual Search Server Settings

NetWare Web Search Manager's home page displays a list of all virtual search servers that exist on your Web Search server. This home page is called Global Settings because the changes you make from this page affect all new sites that you create, and they also affect the functionality of the search and print servlets that provide the Web Search services.

For example, if you changed the Default Query Encoding under Default Settings > General, any new virtual search servers you create would default to the new setting.

However, you can override default virtual search server settings by using the Advanced index definition pages when defining indexes for your sites.

General Settings

Changes you make to the query, response, and error log settings affect all newly created virtual search servers.

To modify general default virtual search server settings, do the following:

1. From the Web Search Manager home page, click General under Default Settings.
2. From the Default Query Encoding drop-down list, select an encoding that represents the character set encoding that most of your user queries will use.
3. In the Maximum Query Duration field, enter the maximum number of seconds before Web Search should end a query, regardless of whether a search has been completed.

   This option is one of several methods for letting you protect your server's resources from processing potential rogue searches, which are sometimes intended to harm your service by consuming server resources.
4. Under Response Settings, select an output encoding from the Default Encoding for Response Pages drop-down list.

   This setting specifies the encoding Web Search should use when responding to user queries using the search and print results templates, and the error and response messages templates.
5. Enter the maximum number of queries in the Refuse Queries if Potential Hits Exceed field to cancel the processing of search results that might take a long time to complete.

6 In the Maximum Log Size field, enter the maximum size (in bytes) that Web Search will allow the log file to grow to.

Depending on the number of visitors that your virtual search server hosts, log files can become large. This setting will protect your system's hard drive resources.

Default Search Settings

To modify default search settings for search features, do the following:

1 From the Web Search Manager home page, click Search under Default Settings.

2 Under Query Results Settings, enter the number of search results in the Default Number of Results to Display field that you want displayed on each search results page.

For example, if you set this to 25 (which is the default setting) and the number of hits in a return was 200, Web Search would only return 25 hits per search results page at a time.

3 Set a limit on the number of results allowed at one time on the results page by entering a number in the Maximum Number of Results to Display field.

4 Enter the highest number of search results that can be returned to a user query in the Highest Allowed Result Number field.

5 Under Template Settings, enter a path to where your Web Search templates are stored in the Templates Directory field.

HINT: The default path is *volume*:*searchroot*\Templates, but if you have created custom templates or for some reason want to keep your templates elsewhere, specify the path here so that Web Search knows where the templates are.

6 From the Default Encoding for Templates drop-down list, select the character set that your templates are written in.

This value will be used even with templates that do not specify an encoding. Encodings found in templates that do not match the encoding you specify here will override this encoding.

7 In the Default Search Page Template field, enter the filename of the search page template you want to use.

If you have created a custom template and want Web Search to use it as your search page, enter its name in this field.

8. In the Default Search Results Template field, enter the filename of the search results template you want to use.

   If you have created a custom search results template and want Web Search to use it as your default search results page, enter its name in this field.

9. In the Template to Use If No Results Returned field, enter the filename of the template that Web Search should return if no results are found.

10. In the Template to Use If Error Occurs field, enter the filename of the template that Web Search should return if there are errors while processing a user's query.

11. Click Apply Settings.

Default Print Settings

To modify default print settings, do the following:

1. From the Web Search Manager home page, click Print under Default Settings.

2. Under Print Results Settings, enter the number of print results in the Default Number of Results to Print field that you want displayed on each print results page.

   For example, if you set this to 25 (which is the default setting) and the number of hits in a return was 200, Web Search would only return 25 hits per print results page at a time.

3. Set a limit on the number of results allowed at one time on the results page by entering a number in the Maximum Number of Results to Print field.

4. Enter the highest number of search results that can be returned to a user query in the Highest Allowed Result Number field.

5. To limit the size of a print job, specify the largest print job size that Web Search will allow in the Maximum Print Job Size field.

   Any users requesting a print job larger than this value will receive a message informing them that their request was too large.

   HINT: This is a useful feature to administrators who want to keep down the size of print jobs in their own companies, departments, or organizations.

6. To be notified when a print job exceeds a certain size, enter the print job size in the Print Job Size Warning field.

By default, this message is sent using the ResponseMessageTemplate.html file and is intended as a warning to users that they are exceeding the allowed print job size. It then prompts the user to confirm the print job before continuing.

7. Under Template Settings, enter a path in the Templates Directory field to where your Web Search templates are stored.

   HINT: The default path is *volume*:*searchroot*\Templates, but if you have created custom templates or for some reason want to keep your templates elsewhere, specify the path here so that Web Search knows where the templates are.

8. From the Default Encoding for Templates drop-down list, select the character set that your templates are written in.

   This value will be used even with templates that do not specify an encoding. Encodings found in templates that do not match the encoding you specify here will override this encoding.

9. In the Default Print Results Template field, enter the filename of the print results template you want to use.

   If you have created a custom print results template and want Web Search to use it when returning print results, enter its name in this field.

10. In the Template to Use If No Results Returned field, enter the filename of the template that Web Search should return if no print results match a user's query.

11. In the Template to Use If More Information Is Needed field, enter the filename of the template to be sent back to users whose print jobs exceed the size you specify in the Print Job Size field. (See Step 6.)

12. In the Template to Use If Error Occurs field, enter the filename of the template that Web Search should return if there are errors while processing a user's print query.

13. Click Apply Settings.

Default Index Settings

These settings are intended to make the process of creating indexes even easier by letting you configure common settings as default settings. This saves you time by not making you make the same selections each time you create a new index.

To modify default index settings, do the following:

1. From the Web Search Manager home page, click Index under Default Settings.
2. Select the type of index that you want as the default index type on the Indexing Management page.
3. Check the URLs Are Case Sensitive check box if you want Web Search to recognize URLs that are different only in character case, but are otherwise identical. For example, www.digitalairlines.com verses www.DigitalAirlines.com.

   IMPORTANT: By setting this option to No can help Web Search to avoid indexing duplicate information, which can come from indexing URLs that are presented using different cases but actually point to the same information. However, if a Web server being indexed is configured to differntiate between cases, Web Search might leave out content that you want indexed.
4. Check the Crawl Dynamic URLs (URLs Containing '?') check box if you want dynamic content indexed, in addition to static content.

   See "About Indexing Dynamic Web Content" on page 160.
5. Enter a number (in bytes) in the Maximum File Size to Index field to keep Web Search from indexing files larger than the number you specify.
6. In the Maximum Time to Download a URL field, enter a number (in seconds) before Web Search automatically skips the indexing of the specified URL.
7. From the Encoding (If Not in META Tags) drop-down list, select the encoding to be used when indexeing files that do not contain an encoding specification.

   For example, HTML files can specify their encoding with a Content-Type META tag.
8. Click Apply Settings.

Default Security Settings

Security settings let you manage access to indexed content by requiring users to authenticate to a server before seeing rights-protected search results.

To modify default security settings, do the following:

1. From the Web Search Manager home page, click Security under Default Settings.
2. In the Basis for Authorization drop-down list, choose from the following options:
   - Allow All means that although the Login button appears on the default search page, no authentication will be required to view information. All results, whether contained in public or private directories, are returned. Web Search will not ask who the user is. This doesn't mean that if information is contained in an eDirectory protected folder that the user will be able to click the link in the results page and be given access.
   - Allow Public means that private content will not be returned in the search results.
   - User Login means that depending on what you select under Unauthorized Hits Filtered By, unauthorized search results will be filtered out either by a results template or by the NetWare Web Search engine.

   IMPORTANT: These settings apply only to file system indexes and to the server where you have Web Search Server installed.
3. Under Connection Settings, click Yes next to Require HTTPS if you want to protect usernames and passwords as they are sent across the network or Internet.
4. Enter a number (in minutes) in the Auto-logout Time field to direct Web Search to log users out who have been idle for the specified period of time.
5. If you need to change the authentication realm string, enter it in the Authentication Realm String field.

   HINT: Specifying the Enterprise Server's authentication realm string in this field makes it so that once users authenticate to the Enterprise Server, they won't have to authenticate again when using Web Search to search and access protected information.

Modifying Default Search Service Settings

Search Service Settings are meant for the administrator of the Web Search server and are intended to give him global control over all virtual search servers, including the ability to completely disable searching by turning it off.

They also allow the administrator to control the overall performance of the Web Search Server.

General Services Settings

General Services Settings affect error log and server settings for all virtual search servers.

To modify general services settings, do the following:

1. From the Web Search Manager home page, click General under Services Settings.
2. Select where you want log results displayed by choosing one of the following options from the Log Errors To drop-down list:
   - File: When this option is selected, you can click View next to the Log Errors To drop-down list and the log results are displayed in your browser.
   - Console: You can also view log results at the NetWare system console by selecting Console, pressing Ctrl+Esc on your server's keyboard, and then pressing the number corresponding to the Tomcat servlet engine.
   - Both: Displays results in both your browser and at the system console.

   HINT: You can access the log file directly by going to *volume*:*searchroot*\errors.log.
3. To start a new log file each time you restart the Web Search server, click Yes next to New Log When Services Load.

   HINT: You can also delete the log file at the path specified above. The log file will be recreated on the first instance of a new error, statistics, etc.
4. To limit the size of the log file, enter a file size (in bytes) in the Maximum Log Size field.
5. To limit the number of indexing jobs that can run at the same time, specify a number in the Maximum Number of Active Index Jobs field.

6. In the Default Location of Virtual Search Servers field, specify the path to where you want all virtual search server files to be stored, including index and configuration files.

   HINT: Changing this setting won't move existing sites to the new default location. But all new virtual search servers will be placed here.

7. To direct Web Search to reload configuration files modified manually, outside of Web Search Manager, click Yes next to the Detect Manual Search Server Changes field.

   If you make changes outside of Web Search Manager, such as modifying a configuration or properties file, Web Search will re-read those files as often as you indicate in the Seconds Between Checking for Changes field.

8. In the Seconds Between Checking for Changes field, specify how often Web Search should check for manual changes (changes made outside of Web Search Manager) to the configuration files.

9. To direct Web Search to reload Web Search templates that have been modified, click Yes next to the Detect Template Changes field.

   After making a change to a template from within your HTML editing tool and saving it on your server, Web Search will re-read the template as often as you specify in the Seconds Between Template Updates field. That way you can test your changes almost immediately.

10. In the Seconds Between Checking for Template Changes field, specify how often Web Search should reload search, print, results, and error templates.

11. Click Apply Settings.

Search Services Settings

Search Services Settings let you turn search capabilities on or off and manage debugging and statistics settings.

To modify search services settings, do the following:

1. From the Web Search Manager home page, click Search under Services Settings.

2. To enable search services for all virtual search servers on your Web Search server, click Yes next to Enable Search Service.

3. Under Debug Settings, click Yes next to Enable Search Debugging if you want to keep a log of all searches and query results going to all virtual search servers.

   IMPORTANT: We recommend that you use this features only while setting up or troubleshooting your search services because depending on search activity, the log file can grow in size very quickly. While you can limit it's size (see Step 6) it can become ineffective when the limit is reached and no more activities are added to the log file.

4. Select where you want log results displayed by choosing one of the following options from the Log Debug Messages To drop-down list:

   - File: When this option is selected, you can click View next to the Log Debug Messages To drop-down list and the log results are displayed in your browser.
   - Console: You can also view log results at the NetWare system console by selecting Console, pressing Ctrl+Esc on your server's keyboard, and then pressing the number corresponding to the Tomcat servlet engine.
   - Both: Displays results in both your browser and at the system console.

5. To start a new log file each time you restart the Web Search server, click Yes next to New Log When Services Load.
6. To limit the size of the log file, enter a file size (in bytes) in the Maximum Log Size field.
7. Under Statistics Settings, click Yes next to Enable Search Statistics Logging if you want an updated log file containing statistics about searches performed against all virtual search servers on your Web Search server.
8. In the Seconds Between Statistics Updates field, enter a number (in seconds) that should elapse between updates of the statistics log file.
9. For the next three fields, follow Step 4, Step 5, and Step 6 above.
10. In the Log Error If Search Time Exceeds field, enter a number (in seconds) before Web Search should record the current search as exceeding the specified time limit on the statistics display. This appears as the Limit portion of the statistics display.
11. Click Apply Settings.

Print Services Settings

To modify print services settings, do the following:

1. From the Web Search Manager home page, click Print under Services Settings.
2. To enable print services for all virtual search servers on your Web Search server, click Yes next to Enable Print Service.
3. Under Debug Settings, click Yes next to Enable Print Debugging if you want print debugging turned on.

   IMPORTANT: We recommend that you use this features only while setting up or troubleshooting your search services because depending on search activity, the log file can grow in size very quickly. While you can limit it's size (see Step 6) it can become ineffective when the limit is reached and no more activities are added to the log file.
4. Select where you want log results displayed by choosing one of the following options from the Log Debug Messages To drop-down list:
   - File: When this option is selected, you can click View next to the Log Debug Messages To drop-down list and the log results are displayed in your browser.
   - Console: You can also view log results at the NetWare system console by selecting Console, pressing Ctrl+Esc on your server's keyboard, and then pressing the number corresponding to the Tomcat servlet engine.
   - Both: Displays results in both your browser and at the system console.
5. To start a new log file each time you restart the Web Search server, click Yes next to New Log When Services Load.
6. To limit the size of the log file, enter a file size (in bytes) in the Maximum Log Size field.
7. Under Statistics Settings, click Yes next to Enable Print Statistics Logging if you want to an updated log file containing statistics about print requests performed on your Web Search server.
8. In the Seconds between Statistics Updates field, enter a number (in seconds) that should elapse between updates of the statistics log file.
9. For the next three fields, follow Step 4, Step 5, and Step 6 above.

10 In the Log Error If Print Time Exceeds field, enter a number (in seconds) before Web Search should record the current print job as exceeding the specified time limit on the statistics display. This appears as the Limit portion of the statistics display.

11 Click Apply Settings.

Backing Up Your Virtual Search Server Files

As with any valuable data, you should make sure that your virtual search server files are backed up. At minimum, you should back up your index files, which by default are stored at *volume*:*searchroot*\sites.

However, if you have customized templates, you might also want to back them up. By default, they are stored at *volume*:*searchroot*\templates.

15 Optimizing Search Results

There are a number of ways administrators can optimize the performance of their virtual search servers.

Improving Search Results through Intelligent Indexing

You can improve the accuracy of your search results by following these indexing guidelines:

- When defining and creating your indexes, start with the highest-level Web Site URLs and File System Paths possible.
- If content is showing up in your search results that you don't want included, try removing some paths or URLs from your defined indexes. Also, try excluding specific subdirectories that you know or suspect might contain content that you don't want searched.
- If you've indexed too many file types and cluttered your search results, try removing file types that you don't want indexed by using the Extensions to Exclude option on the Define Index page.
- Use the Robots META tag in your Web site's content.
- Exclude documents or specific sections of documents, including headers, footers, and navigation bars.

Excluding Documents from Being Indexed

One way to improve search results is to guard what content is actually indexed, thus clearing a path for relevant information.

Using the Extensions to Exclude Option

You can use the Extensions to Exclude option to direct Web Search to ignore specific file types. For example, if you don't want Word or PowerPoint documents to be included in search results, you would enter DOC and PPT in the Extensions to Exclude field. When these document types are encountered during and indexing job, Web Search skips over them.

Using the Extentions to Include Option

As mentioned above, you can use the Extensions to Exclude option to direct Web Search to ignore specific filetypes. However, if you can't specify all of the extensions to exclude, use the Extensions to Include option and specify all acceptable file extensions. A typical list would specify HTM, HTML, PDF, TXT, and DOC.

HINT: When entering extensions in the Extensions to Exclude box, separate each extension by a space or a hard return. Avoid using commas. For example:

```
htm html pdf txt doc
```

Using the Robots META Tag

Another effective way of controlling what Web Search indexes is using the Robots META tag, a tag inserted into the content that is being indexed by Web Search.

When a Web-based search engine encounters a document containing the Robots META tag, the search engine will do as the META tag instructs.

There are several values you can specify in the Robots META tag:

- NOINDEX: Indicates that the document is not to be indexed.
- NOFOLLOW: Indicates that hypertext links in the document are not to be crawled.
- FOLLOWINDEX: Indicates that hypertext links in the document should be crawled.
- ALL: Indicates that the document can be indexed and all links can be crawled.
- NONE: Indicates that the document is not to be indexed and that hypertext links are not to be crawled.

To include the Robots META tag, use this syntax:

```
<META name="Robots" content="value, optional_value">
```

Using the Robots Comment Tag

You can also use the Robots Comment tag to exclude specific sections of HTML documents from your search results. For example, you might not want such sections as repetitive headers, footers, navigation bars, and server-side includes to be indexed.

HINT: You can also place these tags at the top and bottom of all include files so these sections never get indexed when part of a larger document.

To direct Web Search where to begin skipping content while indexing, do the following:

1. At the point in your HTML document where you want Web Search to begin skipping content while indexing, enter the following tag:

   ```
   <!--*Robots NoIndex-->
   ```

2. Just after the content you want skipped, enter the following tag:

   ```
   <!--*Robots Index-->
   ```

3. Save your changes and index (or reindex) the content.

Modifying Document Descriptions Returned in a Search Results List

Web Search returns descriptions of each hit that is listed on the search results page. By default, the following information is returned for each result:

- Description field
- Summary field
- Abstract field
- The first 255 characters of the document (beginning with first heading and skipping links)

The first three fields are taken from the content of META tags in HTML documents or from document summary fields in other document types such as Word* or PDF files. If these tags or fields are not defined, Web Search will try to find the first heading and begin selecting words. If it can't find a heading, then it begins at the top of the document and selects the first 255 relevant display bytes as the description.

Improving the Relevance of Search Results

Web Search utilizes a sophisticated relevance-ranking algorithm. During a search, Web Search considers

- The number of times words appear in a document
- The proximity of words in a multiple word search (the closer the words appear, the more relevant the document will be)
- The order of words in a multiple word search (the exact order of words is more relevant)
- The location of words in a document (specifically words that appear in a META tag, title, body, header, footer, etc.)
- The formatting of words in a document (such as bold, font type and size, etc.)
- Query weighting in a multiple query scenario
- The number of times words occur within an entire index (for example, the word *the* has low relevance)

To illustrate how these criteria work, consider the following examples:

- Words in bold face are more relevant than regular words.
- Words contained in the <Title> tag are more relevant than words contained within the <body> tag.
- Words contained in the Keywords and Description META tags are more relevant than content words.
- Words contained within the <A HREF=> tag used for creating links are less relevant than words outside of this tag.
- A document containing a specified search term multiple times is more relevant than a document that contains the search term only once.
- A word within a 36-point body text is more relevant than within 4-point footer text.
- Documents returned from a query that is weighted at 100% is more relevant than a 50% weighted query. This is normally used in multiquery searches where each query has a specified weight. For example:

  ```
  query0=netware&weight0=100&query1=groupwise&weight1=100
  ```

Weighted Queries

A weighted query is used anytime you want to modify the order or relevance of certain hits in a user's normal search results list or when you want to add additional search results users might not have identified in their queries.

Web Search allows users to submit more than one query item as part of a single search request.

The following query parameters are combined to identify a single search query item:

&filter#=

&filteroperator#=

&operator#=

&query#=

&weight#=

One use of this feature could be to provide profile-enhanced search requests. For example, the following query returns French product downloads higher up in the search results list but does not eliminate results of any other language downloads:

```
&query0=product+downloads&weight0=100&query1=/language=french&weight1=90
```

This example directs Web Search to perform two completely separate searches. The search results from the two queries are then merged based on the relevance of the individual search results and the weighting of the respective query that produced them.

Another example might be to give the search results from one index more or less relevance than the search results of another index when performing a multiple-index search. For example, the search results from Novell might be more relevant than the search results from Novonyx.

To send multiple query items, these parameters must be grouped using a number (#) at the end of the parameter name so they will be interpreted properly. The numbering should begin at 0 or 1 and increment sequentially for each additional query item.

Ensuring Optimal Search Speed

Once a virtual search server has been accessed, all of its configuration files are read into memory. For speed reasons, the virtual search server remains cached in memory until a period of inactivity has elapsed. The virtual search server is then dynamically removed from memory until its next use. Because of this, the first time a virtual search server is accessed is usually the slowest.

However, there are other factors that can affect the performance of your Web Search services. As with any software, the amout of system resources (CPU, RAM, and hard drive) available affects Web Search Server performance. Web Search speed depends on the following factors:

- System processor speed
- Number of processors
- Amount of system memory (RAM)
- Number of hosted virtual search servers
- Number of indexes within each virtual search server
- Number of files included within each index
- Number of indexes included within each query
- Number of queries performed at one time
- Complexity of users' queries
- Number of search results returned with each results page
- Number of concurrent active indexing jobs
- Other functions being performed by your server

Adjusting any of these values can have a significant impact on the performance of your search services.

As a general guideline, use the fastest CPU possible and include as much RAM as possible. Although the duration of each user query is very short, while it is active it consumes an average of 500 KB of memory. Memory consumption varies widely while the indexer is calculating the final search results list, depending on the number of possible search results.

Also, try to schedule the regeneration of your indexes during off-peak hours. That way, they won't interfere with normal user searches. (See "Automating Index and Server Maintenance" on page 160.)

If you find that your users frequently enter search words such as *to*, *of*, *a*, *the*, *in*, etc., you might want to consider removing these words before they are submitted to the Web Search engine.

HINT: A stop-words processor is available in the Enterprise version of Web Search. Contact your local Novell resale representive for more information.

Making Good Use of Document Fields

A document field is any META tag or document summary field that helps to identify the document's contents. A document summary field might be a title, heading, or paragraph contained in a TITLE or META tag within an HTML document.

Web Search is designed to take advantage of document fields in order to improve the accuracy, relevance, display information, and speed of search results.

By design, Web Search always indexes all document fields in many document types, including HTML, PDF, Word, WordPerfect, XML, etc. Users can then constrain searches to the contents of any document field.

As a Web Search administrator, you can also use document fields to further restrict search results to certain products, categories, authors, titles, keywords, or any other content belonging to a document field.

To perform a feild-restricted search, use the /*fieldname=search_criteria* search operator.

HINT: You might consider sending this information as hidden data using the &filter= query parameter. For example:

```
&filter=/product=netware
```

Searching XML Documents

XML documents provide a tremendous advantage to narrowing search results because of their heirarchical structure and use of multiple document summary fields.

Web Search provides complete hierarchical searching using the fieldname=*search_criteria* operator. For example, you can find information anywhere in the XML document, within any of the title tags, or limit it to

within the title tag that is part of the <DOCUMENT><SUMMARY> hierarchy.

The following table shows example uses of the fieldname=*search_criteria* operator when performing a search in XML documents.

Example Values	Result
search_criteria	Finds *search_criteria* anywhere in the document.
/<Document*=*search_criteria*	Finds *search_criteria* anywhere within any tag that is part of the DOCUMENT hierarchy.
/<Document<Summary*=*search_criteria*	Finds *search_criteria* within any tag that is part of the <DOCUMENT> or <SUMMARY> hierarchy.
/<Document<Summary<Title=*search_criteria*	Finds *search_criteria* only within the <DOCUMENT><SUMMARY><TITLE> hierarchy.
/<Document*<Title=*search_criteria*	Finds *search_criteria* within any TITLE tag, located at any level within the DOCUMENT hierarchy.
/<*<Title=*search_criteria*	Finds the *search_criteria* within any TITLE tag in the document.

Using the &filter Query Parameter

The &filter query parameter allows Web Search administrators to enhance searches by adding hidden, additional query details when users submit a search query. This is an enhancement over previous versions of Web Search, which required that you use JavaScript to add additional details to search queries.

The &filter query parameter works just like the &query= parameter and can be used together using the optional number (#) value. For example, if the query parameter was &query0=*search_criteria*, the matching filter parameter would be &filter0=*additional_hidden_search_criteria*. This allows the multiple weighted queries feature to work as designed while allowing administrators to add additional query details to each query.

Unlike the &query parameter, the &filter parameter can be sent multiple times. For example, if users search for software patches, you could include the various products to be searched, which could then improve search time and accuracy:

```
query=software patches

filter=/Products=Product257

filter=/Products=Product16

filter=/Products=Product302
```

The resultant URL might appear as follows, but with the HTTP and domain name prefix:

```
&query=software+patches&filter=%2FProducts%3DProduct257&filter=%2FProducts%3D
Product16&filter=%2FProducts%3DProduct302
```

NOTE: All &filter operators are combined using default the &operator=*value*, AND. Also, the default Boolean conjunction joining the various filter operators is an OR search. You can change the default Boolean conjunction by using the &filteroperator=# query parameter. The pound sign (#) here acts just like the one used in the #operator=# query parameter.

16 Understanding Templates

NetWare® Web Search Server utilizes templates to generate search forms and search and print results as well as user feedback such as error or response messages.

A template is an HTML document containing one or more Web Search Server variables. Template variables are used to produce dynamic results when a user performs a search on the virtual search server you have defined.

Templates can be shared across virtual search servers or each virtual search server can point to its own set of templates.

This chapter describes how templates work and discusses the default NetWare Web Search Server templates that are included. To learn how to customize the default templates, see Chapter 17, "Customizing Your Search Solutions," on page 189, and Chapter 18, "Working with Template Variables and Search Parameters," on page 195.

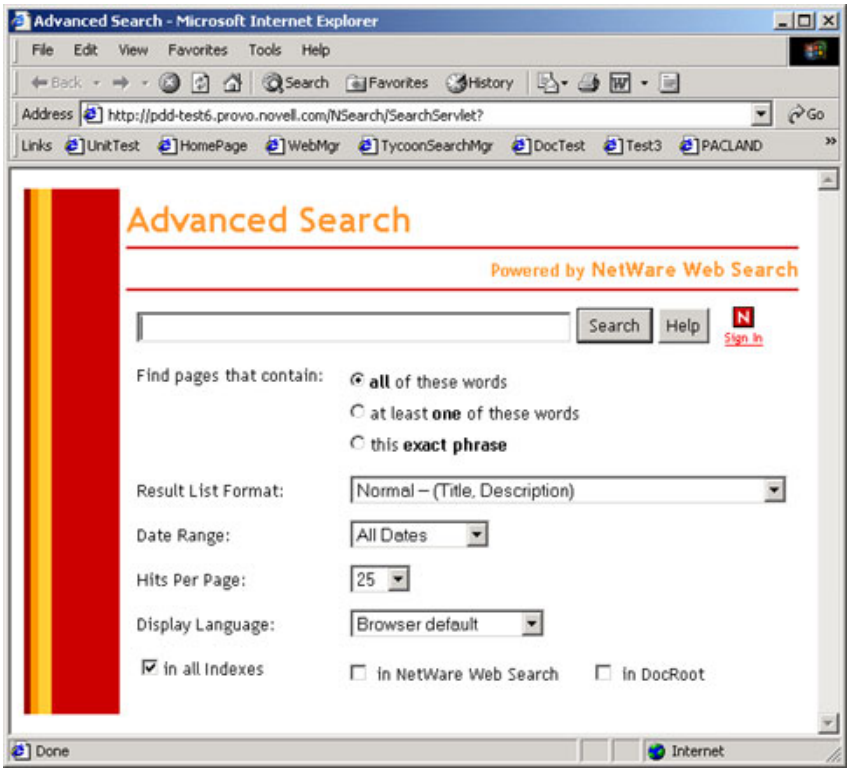
How Templates Work

As defined above, a template is normally an HTML document containing one or more Web Search Server variables. When users search your virtual search server, they use a Web browser to access the search form template. See Figure 7, "The NetWare Web Search Form As It Appears in a Web Browser," on page 184.

The Search form template, SearchTemplate.html, is stored (by default) on *volume*:*searchroot*\TEMPLATES. This path might be different if you chose to install Web Search in another directory.

For more information about customizing templates, see "Customizing the Look and Feel of Search Server Content" on page 139.

Figure 7 The NetWare Web Search Form As It Appears in a Web Browser



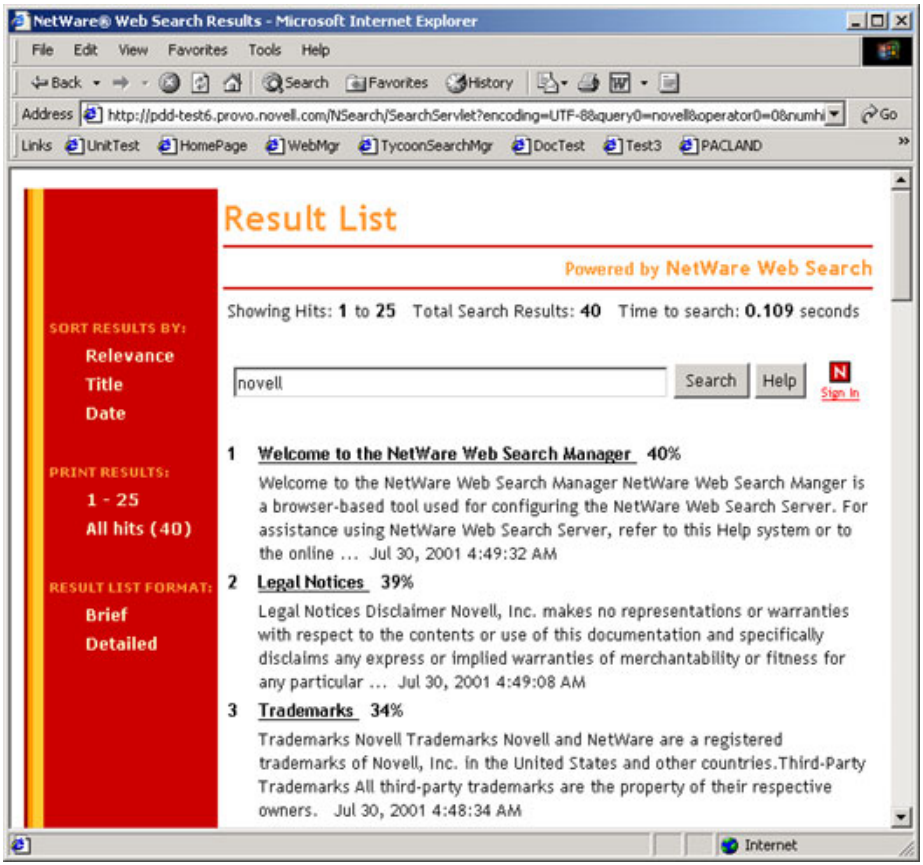
The Web Search form is used to capture user input, select available indexes, and then return the results in either a search or print results template, which appears to the user in a dynamically updated HTML document.

Search result templates display hits according to user selections on the search form. For more information about these search result templates, see Table 8 on page 186.

There are also search and print templates for several different languages. For a discussion about creating templates for international languages, see Chapter 19, “Internationalizing Your Search Solution,” on page 221.

After a query is submitted and results are found, Web Search populates a results template with all relevant information for each search result. (See Figure 8 on page 185.)

Figure 8 A Search Results Page Produced by the Search Results Template, ResultListTemplate.html.



You can also customize the search form to include additional parameters that allow you to offer more options to your users for more accurate searching.

Exploring the Default Search and Print Templates

NetWare Web Search Server includes several default templates used to create search forms and to format, display, and print search results for users.

You can use the templates as they are or you can modify them to look and feel how you want them to. You can also create as many additional templates as you need or replace the default templates with your own templates.

NetWare Web Search includes the five template categories:

- Search Page Templates
- Search Result Templates
- Print Result Templates
- Error Message Template
- Response Message Template

The templates are stored at *volume*:*searchroot*\TEMPLATES.

Search Page Templates

NetWare Web Search includes two search page templates that are used to generate a search page, as described in the following table.

Table 7 Default Search Page Templates

Template Name	Purpose
SearchTemplate.html	Lets users select a variety of options when performing searches and is the default search template used by NetWare Web Search Server.
SearchTemplate.Simple	Similar to SearchTemplate.html, except that this template contains no dynamic indexes.

Search Result Templates

NetWare Web Search includes several ready-made result templates, as described in the following table.

Table 8 Default Search Result Templates

Template Name	Purpose
ResultListTemplate.html	Formats and organizes search results and offers additional sorting functions to the user.

Template Name	Purpose
ResultListNoHitsTemplate.html	Indicates when no hits are found during a search and offers users a chance to refine their search.
ResultListTerseTemplate.html	Similar to ResultListTemplate but returns less information, such as dates and titles only.
ResultListVerboseTemplate.html	Similar to ResultListTemplate, but returns more information, such as file date, time, and language. Additional sort options are also provided.

Print Result Templates

From the search results page, users have the option of printing all files matching their search or only those files displayed on the current search results page. When one of these options is selected, the print result templates described in the following is displayed.

Table 9 Default Print Result Templates

Template Name	Purpose
PrintResultTemplate.html	Combines the full contents of each of the files in the print request into a single document which is then displayed in the user's Web browser. A dynamic table of contents is also created. Once the entire content is downloaded to the browser, the browser's print dialog appears.
PrintResultNoHitsTemplate.html	Indicates when no documents are found during a print request.

Error and Response Message Templates

In addition to the print, search, and search result templates, the error and response message templates are returned when an error occurs or when information is needed from the user.

The default response message template is returned to convey a specific message to the user such as "Print job exceeds recommended size limits," typically returned when a user attempts to print more content than the Web administrator has allowed.

The error and response message templates can be found at *volume*:*searchroot*\TEMPLATES\ErrorMessageTemplate.html and ResponseMessageTemplate.html.

How Templates Use System Memory

Templates are cached in memory for quick rendering speed. Each template consumes approximately 10 KB.

Similar to the virtual search server cache, templates remain cached in memory until a period of inactivity has elapsed. The template is then dynamically removed from memory until its next use. The first time a template is accessed, therefore, is normally the slowest.

HINT: Too many templates in the template cache can consume a great deal of memory. Try to share templates across sites to minimize the impact on system memory resources.

Working with Additional Languages

NetWare Web Search includes each of the templates described above in each of several languages. Using standard encoding practices, you can internationalize your templates.

Any changes made to the default templates should also be made to the language templates you will use. For a more complete discussion about creating a multilingual search solution, see Chapter 19, “Internationalizing Your Search Solution,” on page 221.

17 Customizing Your Search Solutions

You can quickly create a custom search solution by modifying the default NetWare® Web Search templates. Templates include some fundamental options for users, but you can add or remove options and modify the form layout and design to give the search form the look, feel, and function you need. If you are creating a hosted search service for another company's Web site, you can modify the templates to match the look and feel of their Web site.

If you are confident in coding with HTML, you can start with the default search page template to get a feel for the available parameters and then begin coding completely new search and print templates from scratch.

For more information about the necessary components for building a solution from scratch, see Chapter 16, "Understanding Templates," on page 183 and Chapter 18, "Working with Template Variables and Search Parameters," on page 195.

Customizing Templates

You can extend the capabilities of NetWare Web Search Server by customizing the templates.

The first step is to determine which components of Web Search you want to customize. For example, if you only want to add a few additional search features to the search page template and modify its background color and table size, you would modify the SearchTemplate.html or SearchTemplate.Simple files.

This section discusses how to customize the search, print, and result templates and how to use available parameters and variables to create a customized search solution.

Customizing the Search Templates

You can customize the design and functionality of the static or dynamic search templates by

- Modifying HTML code
- Adding or removing search parameters

If you are familiar with HTML, you can quickly modify the design of the default (dynamic) search template or the static search template. For example, you can change the colors of the search page or add new custom graphics.

To modify the functionality of the default search template, you can add or remove search parameters. Search parameters are used to communicate with NetWare Web Search. By embedding them in the correct places in your HTML source, you can extend or limit the functionality of the default search template.

For example, if you wanted your users to use a specific set of templates found in a themes directory, you would add the following HTML code, including the theme parameter, to the SearchTemplate.html file:

```
<INPUT TYPE="Hidden" NAME="theme" VALUE="$$QueryTemplateTheme">
```

This sample HTML code tells Web Search to look for templates only in the specified template directory. All themes are located within the templates directory specified in Web Search Manager.

For a complete list of available search parameters, see Table 16, “Search Parameters,” on page 211.

Customizing Search Result Templates

NetWare Web Search Server includes several default search result templates that are used to display hits, provide feedback to a user, or request information from a user after a search is performed. For more information about the default search result templates, see Chapter 16, “Understanding Templates,” on page 183.

You can customize the design and functionality of the default search result template, the template used when a user selects Normal from the Result List Format drop-down list in the NetWare Web Search form. For information about how to access the NetWare Web Search form, see “Taking a Test Run” on page 138.

Customizing the default search result template involves

- Modifying the HTML code
- Adding or removing search result variables

If you are familiar with HTML, you can quickly modify the design of the default search result template. For example, you can change the colors of the search page or add new graphics.

To modify the functionality of the default search result template, you can add or remove search result variables. Search result variables are placed in the template where you want search results to be displayed.

For example, if you wanted to display the total number of hits returned when a user performs a search and you wanted the information to appear in the upper-left corner of the search results page, you would add the following HTML code to the search result template file:

```
Total Search Results: $$TotalHits
```

After a user performs a search, the $$TotalHits variable would be replaced by the actual total number of hits found during the search.

The $$TotalHits variable is used to retrieve the total number of hits found during a search. You can place this variable anywhere in the results list template to organize the display of information in the way you want.

Default search result templates are located in *volume*:*searchroot*\TEMPLATES\. For a complete list of search result variables that you can use to customize default search result templates or to create new ones, see Table 12, “Search Result Variables,” on page 203.

Customizing Print Result Templates

NetWare Web Search Server includes two default print result templates: the default print result template and a "no hits" template. Print result templates are used to organize and format search results for printing and to provide feedback

to a user when no hits are found. For more information about the default print result templates, see Chapter 16, “Understanding Templates,” on page 183.

You can customize the design and functionality of the default print result template in the same way you customize the search result template by

- Modifying the HTML code
- Adding or removing print result variables

If you are familiar with HTML, you can quickly modify the design of the default print result template. For example, you can change the colors of the print results page or add new graphics to it.

To modify the functionality of the default print results template, you can add or remove print result variables. Variables are placed in the template where you want search results to be displayed.

For example, if you wanted to remove the table of contents from the default print results template, you would remove, or comment out, the following HTML code in the PRINTRESULTLIST.HTML template, which would include the $$BeginTOCList variable:

```
<CENTER><H2>Table of Contents</H2></CENTER>

<p>

<!--            TABLE OF CONTENTS                    -->

$$BeginTOCList[<BIG><B>$$Product</B></BIG>

<DL>]

<DT><A HREF="#$$Bookmark"><BIG>$$Title</BIG></A>

<SPACER TYPE=HORIZONTAL SIZE=20>

<I><SMALL>[$$URL]</SMALL></I>

$$EndTOCList[</DL>]
```

You could either save your changes in the default print result list template or you could save it using a new name, thereby creating an alternative template for users who don’t want a table of contents in the print results. To be effective, you would then have to add a hypertext link in the search result template that would include the &template=*new_template_name* query parameter.

Default print result templates can be found at *volume*:*searchroot*\TEMPLATES. For a complete list of print result variables that you can use to customize default search result templates or to create new ones, see Table 13, “Print Result Variables,” on page 207.

Customizing Error and Response Message Templates

Error and response messages are used to either provide feedback to the user or to request information from the user.

Error and response message templates are used to display the content of error and response messages sent by the Web Search Server in response to search or print errors. Similar to search and print templates, error and response templates can be customized. However, because the contents of error and response messages are built into NetWare Web Search Server, you cannot modify the contents of the messages or the button objects that might appear, depending on the type of response being generated.

Customizing Error Messages

There are several error messages that can be returned to a user. For example, when users incorrectly use a search operator in a search form, they might get the message, "Search Error: Incorrect use of Boolean operator." An error number might also appear.

While you can utilize HTML tags to format an error message, add or remove variables to determine what information is shown to the user, or even reorganize where the messages will appear in the template, you cannot modify the message itself.

Customizing Response Messages

The same concepts apply to response messages, but response messages return buttons that a user can click. Which buttons appear are determined by the NetWare Web Search Server. While you can modify the labels of these buttons, you cannot determine which buttons will appear, or when.

Testing Your Search and Print Solution

Once you've completed customizing the templates and the search form, you can test them in your Web browser by pointing to the search form URL and entering a search string. See "Taking a Test Run" on page 138 for information about how to access the NetWare Web Search form.

HINT: Remember that a search cannot be performed until you have defined at least one index and generated it using NetWare Web Search Manager. Refer to Web Search Manger's online Help for the steps required in defining and generating an index. Also, see Chapter 11, "Introducing NetWare Web Search Server," on page 133 for an overview of indexes.

18 Working with Template Variables and Search Parameters

If you are a developer or are comfortable programming in HTML and working with variables and parameters, you can create an advanced search solution that your users can use to perform complex searches.

Building an advanced search solution involves the use of search and print template variables and search parameters to create or customize search and print templates, and to create or customize one or more search forms.

You must also have used NetWare® Web Search Manager to define and generate one or more indexes.

The Web Search Manager is accessed from NetWare Web Manager. For more information about using NetWare Web Manager, see Chapter 2, "Introducing NetWare Web Manager," on page 25.

Guidelines for Using Variables

Please note the following guidelines when using variables to either customize the default templates, or to create new templates from scratch:

- Case Sensitivity: All variables are case sensitive. Changing case in a variable will cause Web Search to ignore the variable.
- Variable Formatting: All variables must be used exactly as they appear in the tables below. Variables always begin with two dollar signs ($$) next to each other.
- Success of a Variable: The inclusion of a variable does not guarantee that information will be returned after a search is performed. For example, using the $$Author variable might not return the name of a document's

author if that information is not included in the META tag of the document.

- Internationalizing Templates: If you want to internationalize your templates, you must create a template for each language you want to support in your search solution. For more information about languages, see Chapter 19, “Internationalizing Your Search Solution,” on page 221.

IMPORTANT: In prior versions of NetWare Web Search Server, the term *search site* was defined as a collection of one or more indexes and related configuration files. To avoid confusion with the term Web site, the term was changed wherever it appeared in the documentation and in the variables and parameters. *Search site* is now referred to as *virtual search server*.

New variables and parameters that parallell the term *virtual search server* have been added. Note that they function identically to the prior variable and parameters, and that the old variables and parameters can still be used.

Similarly, the term *collection* has been changed to *index* .

We recommend that you start using the newer variables and parameters so as to avoid confusion.

For more information about how to implement variables in a search or print template, or how to implement search parameters in an HTML document to create a search form, see Chapter 17, “Customizing Your Search Solutions,” on page 189.

Global Template Variables

Global template variables are ones that can be used in any of the Web Search templates.

Table 10 Global Template Variables

Variable Name	Description
$$Authenticated	Indicates whether or not the user is authenticated by returning either a 1 or 0.
$$BeginAuthenticated	Begins a section for a valid, logged in user. Used in conjunction with $$EndAuthenticated. If a user is authenticated, the text between these two tags will be processed and appear in the output. If a user is not authenticated, the text will be removed from the search result. To control the appearance of unauthenticated search results, see $$BeginUnAuthenticated.

Variable Name	Description
$$BeginLoop	End of the header section. Beginning of the repeating body section. This section is repeatedly parsed until there are no further result items to process. See also “$$EndLoop” on page 198.
$$BeginIndexesLoop	Begins a repetitive section that will be processed for each index the user specified in the search query. See also “$$EndIndexesLoop” on page 198 and “$$QueryIndex[number]” on page 200.
$$BeginFiltersLoop	Begins a repetitive section that will be processed for each filter parameter associated with the current query item. Note that multiple query items can be sent as part of a single query. See also “$$EndFiltersLoop” on page 198.
$$BeginQueryLoop	Begins a repetitive section that will be processed for each query item associated with the current search query. See “$$NumQueryItems” on page 199 for more information. See also “$$EndQueryLoop” on page 198.
$$BeginReturnFieldsLoop	The beginning of a repetitive section that will be reprocessed for each return field the user specified in the search query. See also “$$QueryReturnField[number]” on page 200.
$$BeginSortKeysLoop	Begins a repetitive section that will be processed for each sort key the user specified in the search query. See also “$$EndSortKeysLoop” on page 198 and “$$SortKeysCurrent” on page 202.
$$BeginUnAuthenticated	Begins a section for an unrecognized or logged out user. Used in conjunction with $$EndUnAuthenticated. If a user is not recognized, the text between these two tags will be processed and appear in the output. If the user is recognized as a valid, logged-in user, this text will not appear in the output. To control the appearance of authenticated search results, see $$BeginAuthenticated.

Variable Name	Description
$$Counter[*variable_number*, *increment_number*]	Inserts the value of the specified var# counter into the search result page. All counters initialize to zero. The optional second parameter specifies the amount to increment or decrement the current value. A maximum of 10 counters is supported. For example: $$Counter[1] = insert value of counter #1 $$Counter[1,1] = increment counter #1 by 1 and display the new value $$Counter[5,-3] = decrement counter #5 by 3 and display the new value
$$DefaultQueryEncoding	Default encoding of user query if not specified using the &encoding= query parameter.
$$EndAuthenticated	Ends a section for a valid, logged in user. For more information, see "$$BeginAuthenticated" on page 196.
$$EndFiltersLoop	Ends a repetitive section that will be processed for each filter parameter associated with the current query item. See also "$$BeginFiltersLoop" on page 197.
$$EndIndexesLoop	Ends a repetitive section that will be processed for each index the user specified in the search query. See also "$$BeginIndexesLoop" on page 197 and "$$QueryIndex[number]" on page 200.
$$EndLoop	End of the repeating body section. Beginning of the footer section.
$$EndQueryLoop	Ends a repetitive section that will be processed for each query item associated with the current search query. See "$$NumQueryItems" on page 199 for more information. See also "$$BeginQueryLoop" on page 197.
$$EndReturnFieldsLoop	The end of a repetitive section that will be reprocessed for each return field the user specified in the search query. See also "$$QueryReturnField[number]" on page 200.
$$EndSortKeysLoop	Ends a repetitive section that will be processed for each sort key the user specified in the search query. See also "$$BeginSortKeysLoop" on page 197 and "$$SortKeysCurrent" on page 202.
$$EndUnAuthenticated	Ends a section for an unrecognized or logged out user. For more information, see the description for the $$EndUnAuthenticated variable.

Variable Name	Description
$$FilterCount	Identifies the number of filters associated with the current query item. Note that multiple query items can be associated with a single query.
$$FilterCurrent	Identifies the number of the current filter associated with the current query item.
$$FilterOperator	Identifies the boolean operator used to join the filters associated with the current query item. The full set of filters is always joined to the current query item using the Boolean AND.
$$FilterValue[*number*]	Pulls the value of the specified filter associated with the current query item. If the optional # parameter is not provided, the current filter loop value ($$FilterCurrent) is used.
$$IncludeFile[*template_name*]	Automatically pulls in the designated template at the location of this variable. The included template can contain other template variables, which will be processed as though they were a part of the original template. The template name parameter can either be a full FILE:// URL based on the file system of the server or a relative path based on the location of the parent template. The template name parameter can be located within quotation marks. See the SearchResultTemplate.html file for an example use of this variable.
$$IndexesCount	The number of indexes associated with the user query. See also “$$BeginIndexesLoop” on page 197.
$$IndexesCurrent	The number of the current index. See also “$$BeginIndexesLoop” on page 197.
$$NumQueryItems	The number of query items contained within the current query. While most queries use only 1 query item, it is possible to construct a query with multiple search criteria, each weighted with a value between 1 and 100. While the resultant search contains hits from each of the queries, the search results are organized with the most relevant hits first (from any of the individual queries).
$$Query[number]	Query specified by the client into the search field. The optional number identifies the corresponding query item. The value of $$QueryCurrent is used if the optional number is not provided. See also “$$NumQueryItems” on page 199 for more information.
$$QueryCount	The number of query items associated with the search query. See also “$$NumQueryItems” on page 199 for more information.

Variable Name	Description
$$QueryCountry	The country requested by the client. Note that this must be an uppercase, two-character value as specified in ISO 3166-1.
$$QueryCurrent	The number of the current query item. See “$$NumQueryItems” on page 199 for more information. See also “$$BeginQueryLoop” on page 197.
$$QueryDate	The begin date requested by the client. Only those documents dated on or after the specified date will be returned in the search results. See the query parameter “date” on page 214 for more information.
$$QueryEncoding	Actual encoding used to interpret the query (this could be either the same as the $$DefaultQueryEncoding, the value of the &encoding= query parameter, or UTF8).
$$QueryFileFilter	Returns the filename filter associated with the user query.
$$QueryIndex[*number*]	The names of the indexes the user specified in the search query. If the optional number is not provided, the current value of $$IndexesCurrent is used. See also “$$BeginIndexesLoop” on page 197.
$$QueryLanguage	The language requested by the client. Note that this must be a lowercase, two-character value as specified in ISO 639.
$$QueryNumHits	The number of search results requested by the client.
$$QueryOperator	The type of the current search: 0 = Boolean AND search 1 = Boolean OR search 2 = phrase search
$$QueryReturnField[*number*]	The name of the return fields the user specified in the search query. If the optional number is not provided, the current value of $$ReturnFieldsCurrent is used. See also “$$BeginReturnFieldsLoop” on page 197 and “$$EndReturnFieldsLoop” on page 198.

Variable Name	Description
$$QueryServerName[text]	Identifies the name of the Virtual Search Server provided with the &server= query parameter. The optional text parameter can be provided in the following formats: $$QueryServerName = NameOfServer $$QueryServerName[text] = text NameOfServer $$QueryServerName[%text] = text URLEncodedNameOfServer $$QueryServerName[text $$QueryServerName text] = text NameOfServer text $$QueryServerName[%text $$QueryServerName text] = text URLEncodedNameOfServer text See also “$$ServerName” on page 202.
$$QueryTemplate	The template name requested by the client. See also “$$TemplateName” on page 202.
$$QueryTemplateTheme	The template theme requested by the client. This is not necessarily the theme of the search result since the specified theme may not exist. See also “$$TemplateTheme” on page 202.
$$QueryVersion	The version number of the current query format.
$$QueryWeight[*number*]	The weighting of the current query item 1-100. See “$$NumQueryItems” on page 199 for more information.
$$ResultEncoding	Identifies the encoding used to return the current search results page. This is either the value of the valid &retencoding= query parameter or the default specified by the search administrator in the NetWare Web Search Manager.
$$ReturnField[*number*]	The name of the return fields the user specified in the search query. If the optional number is not provided, the value of $$ReturnFieldsCurrent will be used. See also “$$BeginReturnFieldsLoop” on page 197 and “$$EndReturnFieldsLoop” on page 198.
$$ReturnFieldsCount	The number return fields specified in the search query. See also “$$BeginReturnFieldsLoop” on page 197.
$$ReturnFieldsCurrent	The number of the current iteration of the $$BeginReturnFieldsLoop.

Variable Name	Description
$$SearchFor[*number*]	Query entered by the client into the search field. If the optional number is not provided, the value of $$QueryCurrent will be used. See “$$Query[number]” on page 199 for more information.
$$ServerName	The name of the virtual search server that produced the current output. See also “$$QueryServerName[text]” on page 201.
$$ServerLocation	Path on the network server to the virtual search server configuration files and indexes.
$$SortField[*number*]	The name of the field to sort on. If the optional number is not provided, the value of $$SortKeysCurrent will be used. See “$$SortByURL[sortfield.sortorder ...]” on page 206 and the query parameter “sortfieldnumber” on page 220 for more information.
$$SortKeysCount	The number of sort keys associated with the current query.
$$SortKeysCurrent	The current sort keys number. See “$$BeginSortKeysLoop” on page 197 for more information.
$$SortOrder[*number*]	The order to sort the field (ascending, descending, and default). If the optional number is not provided, the value of $$SortKeysCurrent will be used. See “$$SortOrder[number]” on page 202 for more Information.
$$TemplateLocale	Identifies the locale of the template, such as zh_TW. The locale information is taken from the template filename.
$$TemplateName	Identifies the filename of the template currently displayed in the browser. See also “$$QueryTemplate” on page 201.
$$TemplateTheme	Identifies the theme (or theme directory) that the current template belongs to. See also “$$QueryTemplateTheme” on page 201.

Search Page Variables

In addition to the global template variables, the following table lists all available search page variables that can be used to extend the functionality of the default search templates (SearchTemplate.html or SearchTemplate.Simple) or to create new templates from scratch.

Table 11 Search Page Variables

Variable Name	Description
$$BeginServerIndexesLoop	Begins a repeating section in the search template where information for each of the defined indexes will be written.
$$EndServerIndexesLoop	Ends a repeating section in the search template where information for each of the defined indexes will be written.
$$ServerIndexDescription	Inserts the description of the virtual search server defined in the Web Search Manager.
$$ServerIndexName	Inserts the name of the virtual search server defined in the Web Search Manager.

Search Result Variables

In addition to the Global Template Variables, the following table lists all available search result variables that can be used to extend the functionality of the default search result templates or to create new templates from scratch.

Table 12 Search Result Variables

Variable Name	Description
$$Author	The name of the original author of a document returned in a hit.
$$BeginAuthorized	Begins a section for a search result that the user has rights to see. Used in conjunction with $$EndAuthorized. If a search result is authorized, this section of text and template variables will be processed. If unauthorized, this section is removed from the output. See also "$$BeginUnAuthorized" on page 204.

Variable Name	Description
$$BeginUnAuthorized	Begins a section for a search result that the user does not have rights to see. Used in conjunction with $$EndUnAuthorized. If a search result is not authorized, this section of text and template variables will be processed. If the search result is authorized, this section is removed from the output. See also “$$BeginAuthorized” on page 203.
$$EndAuthorized	Ends a section for a search result that the user has rights to see. For more information, see the description for the $$BeginAuthorized variable.
$$EndUnAuthorized	Ends a section for a search result item that the user does not have rights to see. For more information, see the description for the $$EndUnAuthorized variable.
$$Index	The name of the index in which a particular search result item was found.
$$DateTime[*date_format*]	The date and time of a hit. This is automatically written in Java’s medium format using the client’s locale (all calendars, translations, date and time formats are observed). $$DateTime[ ] can use an optional date and time format provided within the brackets [ ]. The text should conform to the Java *DateFormat* syntax.
$$Description	The abstract, description, or first 255 display bytes of the result item.
$$FileFormat	Indicates a specific document type. For example, .DOC or .PDF.
$$FirstHit	The hit number of the first item in the current result page. Is displayed using the client’s locale.
$$Language	The language of the result item. $$Language is displayed in the language of the client’s locale.
$$LastHit	The hit number of the last item in the current result page. Is displayed using the client’s locale.

Variable Name	Description
$$MoreHits[*page#*, *text*]	A conditional text section to be included only if there are additional hits in the search results that can be retrieved. If the first section of the conditional text contains a number followed by a comma (for example, $$MoreHits[3, text to be included]), then the server will first determine if the designated search results page exists. If the page# is missing, 1 (the next page) is assumed. If the designated page is available, the remaining text after the comma and up to the closing bracket is written to the result page. Note that the initial number is relative to the current page. That is, -1 references the page immediately before the current page and 1 references the page immediately after. Zero refers to either the previous page or the next page.
$$MoreHitsURL[*page_number*]	The URL needed to display another page of search results. The optional parameter identifies the desired search result page number. If not provided, 1 is assumed. Note that the page number is relative to the current page. That is, -1 refers to the page immediately before the current page and 1 references the page immediately after. Zero (0) refers to the current page. The URL is inserted only if the designated page exists.
$$Number	The hit number of the current result item. Possible numbers begin with 1 and end with $$TotalHits. Is displayed using the client's locale.
$$PageNum[*page#*]	Inserts a user-specified search result page number. The optional page# parameter identifies the relative page from the current result page. That is, minus one (-1) refers to the page immediately before the current page and one (1) references the page immediately after. Zero (0) refers to the current page. The page number is inserted only if the designated page exists.
$$PrintURL[*first_hit_number*, *number_of_hits*]	The URL used to print the hits listed on the current search result page. The optional parameters can be specified to define the beginning search result number and the number of search results to include in the print job. The *number_of_hits* parameter can use the $$TotalHits template variable.
$$Relevance	How closley the result matches the user's query, indicated by percentages (1% to 100%).

Variable Name	Description
$$SearchFor[*number*]	Query entered by the client into the search field. See "$$Query[number]" on page 199 for more information.
$$SearchTime	Inserts the amount of time used to process the current search request. $$SearchTime is displayed using the client's locale.
$$Size	The size of the data pointed to by the result item's URL. Is displayed using the client's locale.
$$SortByURL[*sortfield.sortorder ...*]	The URL used to show the current result page sorted by one or more search result fields. Sort field names include title, author, changedate, filelength, language, summary, relevance, url, index, format, and document_number. Optional sort orders include ascending and descending. Sort field and sort order names are separated by a period. Multiple sort fields are separated by a space.
$$Title	If a title is not available in documents being searched, $$URL is used instead; if the URL is unavailable, < *title unavailable* > continues to be used.
$$TotalHits	The total number of hits that match the search query. This is not the same as the number of hits displayed in any particular result page. Is displayed using the client's locale.
$$URL	URL of the result item.

Print Result Variables

In addition to the Global Template Variables, the following table lists all available print result variables that can be used to extend the functionality of the default print result templates or to create new templates from scratch. For more information about how to implement variables in a template (HTML) page, see Chapter 17, "Customizing Your Search Solutions," on page 189.

Table 13 Print Result Variables

Variable Name	Description
$$BeginAuthorized	Begins a section for a search result that the user has rights to see. Used in conjunction with $$EndAuthorized. If a search result is authorized, this section of text and template variables will be processed. If unauthorized, this section is removed from the output. See also "$$BeginUnAuthorized" on page 204.
$$BeginUnAuthorized	Begins a section for a search result that the user does not have rights to see. Used in conjunction with $$EndUnAuthorized. If a search result is not authorized, this section of text and template variables will be processed. If the search result is authorized, this section is removed from the output. See also "$$BeginAuthorized" on page 203.
$$EndAuthorized	Ends a section for a search result that the user has rights to see. For more information, see "$$BeginAuthorized" on page 203.
$$EndUnAuthorized	Ends a section for a search result item that the user does not have rights to see. For more information, see "$$BeginUnAuthorized" on page 204.
$$BeginTOCList[*text*]	Beginning of the table of contents repeating section. This section is repeatedly parsed until there are no further TOC result items to process. This is a conditional text section. The items within the brackets ([ ]) are processed only if the current item represents a change in the depth of the hierarchy. If $$Product appears within the conditional text, it will be replaced only if the current item also represents a new product.
$$Bookmark	The HTML anchor name of the current result item. This can be used to jump from a TOC entry to the corresponding section within the print job. All bookmark entries begin with "novell_print_toc_" and are followed by the number of the current result item, as in novell_print_toc_1.
$$Description	The abstract, description, or first 255 display bytes of the result item.

Variable Name	Description
$$EndTOCList[*text*]	End of the table of contents section. This is a conditional text section. The items within the brackets ([ ]) are written out each time a result item occurs that decreases the depth of the hierarchy. If the depth of the current item is several levels less than the previous item, the text within the conditional text block is written out that many times.
$$Number	The hit number of the current result item. Possible numbers begin with 1 and end with $$TotalHits. Is displayed using the client's locale.
$$NumIndents	The number of indentations required for the current Table of Contents entry.
$$Product	The name of the product associated with the current item in the table of contents. This displays only if this is the first result item within that product. See also "$$BeginTOCList[text]" on page 207.
$$Title	Title of the result item. For empty titles, <title unavailable> is displayed. Is localized using the client's locale.
$$TotalHits	The total number of hits that match the search query. This is not the same as the number of hits displayed in any particular result page. Is displayed using the client's locale.
$$URL	URL of the result item.
$$URLContent	The entire contents of the URL are placed into the template at this location. The URL contents are not parsed to validate their data type, formatting, or functionality. Only text/plain and text/html files are printed. All other files are inserted into the print job as an error message.

Error Message Variables

In addition to the Global Template Variables, the following table lists all available error message variables that can be used to enhance the organization of the default error message template, or to create new templates from scratch. For more information about how to implement variables in a template (HTML), see Chapter 17, "Customizing Your Search Solutions," on page 189.

Table 14 Error Message Variables

Variable Name	Description
$$ErrorNumber	A numeric version of the error.
$$ErrorMessage	A text version of the error. Generally quite terse.
$$ErrorDescription	A longer version of the message. This might include additional error details or problem resolution information.

Response Message Variables

In addition to the Global Template Variables, the following table lists all available response message variables that can be used to enhance the organization of the default response message templates or to create new templates from scratch. For more information about how to implement variables in a template (HTML), see Chapter 17, "Customizing Your Search Solutions," on page 189.

HINT: The repeating variables $$BeginLoop and $$EndLoop should not be used in a response message and will be ignored if used.

Table 15 Response Message Variables

Variable Name	Description
$$Cancel[*text*]	If the Cancel button is specified by Server logic, parses and inserts the conditional text into the response page. Currently used by PrintServlet when a print job exceeds the print job size warning limit.

Variable Name	Description
$$Continue[*text*]	If the Continue button is specified by Server logic, parses and inserts the conditional text into the response page. Currently used by PrintServlet when a print job exceeds the print job size warning limit.
$$Help[*text*]	If the Next button is specified by Server logic, parses and inserts the conditional text into the response page.
$$Ignore[*text*]	If the Ignore button is specified by Server logic, parses and inserts the conditional text into the response page.
$$Next[*text*]	Compare to $$Prev.
$$No[*text*]	If the No button is specified by Server logic, parses and inserts the conditional text into the response page.
$$OK[*text*]	If the OK button is specified by Server logic, parses and inserts the conditional text into the response page. Currently used by PrintServlet when a print job exceeds the maximum print job size.
$$Prev[*text*]	If the Previous button is specified by Server logic, parses and inserts the conditional text into the response page.
$$ResponseNumber	A numeric version of the response required of the user.
$$ResponseMessage	A text version of the response required of the user. Generally quite terse. Can often be used as a title.
$$ResponseDescription	A longer version of the message. This might include additional details or *see also* type information.
$$Retry[*text*]	If the Retry button is specified by Server logic, parses and inserts the conditional text into the response page.
$$URL	Inserts the URL to use when the parent button is clicked. This must appear within the brackets of a button's conditional text section. The URL logic is generated by the server.
$$Yes[*text*]	If the Yes button is specified by Server logic, parses and inserts the conditional text into the response page.

Search Parameters

The following table lists all available search parameters, including required syntax, a description of their default values, and examples. Each of these parameters can be used to extend or enhance the functionality of the search page templates or to create new search page templates from scratch. For more information about how to implement parameters in an HTML document, see Chapter 17, “Customizing Your Search Solutions,” on page 189.

HINT: If you use a parameter but leave its value blank, the default value for that parameter will be used.

Table 16 Search Parameters

Parameter Name	Value	Description
server	String	Specifies the name of the virtual search server which is to receive this request. This query parameter is optional if the domain name of the request matches the name or alias of a registered virtual search server. **Syntax**: server=*virtual_search_server_name* **Example**: server=digitalairlines **Default**: domain name portion of search request
query*number*	String	The actual search criteria that is passed to the Web Search Server. The next six parameters below are combined with this parameter and are identified by adding the unique number to them. **Syntax**: query*number*=searchcriteria **Example**: query0=novell+AND+groupwise

Parameter Name	Value	Description
filter*number*	String	The &filter#= query parameter is used to send additional query details not specified by the user to help limit the scope of a search. Normally, these would be included as hidden fields on an HTML form. This parameter supports all of the same features and functionality as the &query= parameter. However, unlike the &query= parameter, this parameter can be sent multiple times for a single query item. The individual filters associated with a single query item are joined using the value of the filteroperator parameter. The set of filters is logically joined with the rest of the query item using the boolean AND operator. **Syntax**: filter*number*=searchcriteria **Example**: filter0=/product=Groupwise See also “filteroperatornumber” on page 212.
filteroperatornumber	Number	This parameter identifies the boolean conjunction to be used between multiple filters (several filters can be associated with a single query item). The complete set of filters is always associated with the corresponding query item using the boolean AND operator. **0** = AND **1** = OR **2** = PHRASE **Syntax**: filteroperator*number*=*number* **Example**: filteroperator0=1 **Default**: None See also the query parameters “operatornumber” on page 213 and “filternumber” on page 212.

Parameter Name	Value	Description
id*number*	String	A document ID that is used to narrow a search. You can specify more than one ID by using the same field name more than once. **Syntax**: id*number=documentID* **Example**: &id0=z1.0010.&id0=z1.0020 **Default**: None
operator*number*	Integer	Indicates which operator to use between two or more words in a search. **0** = AND **1** = OR **2** = PHRASE **Syntax**: operator*number=number* **Example**: operator0=1
weight*number*	Integer	Lets you assign a level of importance to the current query item as far as it relates to the other query items that are part of the search query. Web Search Server uses this number along with the relevance number to determine a search results' final relevance and then orders the results accordingly. **Range**: 0 to 100 **Syntax**: weight*number=number* **Example**: weight0=75

Parameter Name	Value	Description
type*number*	Integer	Indicates the type of search. Options include: **0** = Normal search; 0 is the default. **1** = Searches only the given document numbers. **2** = Root search used by the search tree control to get the top tree nodes. **3** = Used to get the children of the given document number. **4** = Searches the descendants of the given document numbers and is used to narrow a search or a print request, including all of its children. **Syntax**: type*number*=*number* **Example**: type0=2 **Default**: 0 (zero)
date	Integer	Lets you specify a date range to be searched in milliseconds. The example shows the number of milliseconds spanning a three-month time frame. The minus sign (-) before the number indicates three months back in time. If you pass a positive number such as 940457147873, then Web Search creates a date and time based on the number of milliseconds elapsed since January 1, 1970; 12:00 a.m. The example number 940457147873 produces the search start date of October 20, 1999, at 4:05:47 p.m. **Syntax**: date=*number* **Example**: date=-7905600000

Parameter Name	Value	Description
filefilter	String	Use this query parameter to filter search results based on their path, domain, filename, or extension. This parameter uses the same query syntax as the &query= parameter. **Syntax**: filefilter=*value* **Example**: filefilter=www.novell.com **Example**: filefilter=pdf **Default**: None
index	String	Lets you restrict a search to one or more specified indexes. The index name you specify using this parameter must exactly match the name of an index defined at the server. You can specify more than one index by either sending this parameter more than once or by separating the list of indexes with a semi-colon. **Syntax**: index=*index_name1[;index_name2]* **Example**: index=GroupWise&index=NetWare **Example**: index=GroupWise;NetWare
numhits	Integer	Indicates the number of hits you want returned at one time in the search results page. **Syntax**: numhits=*number* **Example**: numhits=25 **Default**: 25
starthit	Integer	Indicates the hit number you want Web Search to begin searching from. If you entered 35 as the STARTHIT parameter value, Web Search would return hits beginning with hit number 35. **Syntax**: starthit=*number* **Example**: starthit=35 **Default**: 1

Parameter Name	Value	Description
lang	String	Lets you specify a language using the two-character, lowercase language value derived from ISO6391. **Syntax**: lang=*language_code* **Example**: lang=ja **Default**: browser language preference
country	String	Lets you specify your country using the two-character, uppercase country value derived from ISO3166. **Syntax**: country=*country code* **Example**: country=TW **Default**: browser language preference
template	String	Lets you specify the specific results template you want your search results returned in. The following list of templates are the default templates included with the Web Search Server. However, your Web Search Server administrator might have created custom templates using different names. Check with your administrator if these templates do not work for you. You must type the names of these templates *exactly* as they appear in this list: ◆ ResultListTemplate.html ◆ ResultListTerseTemplate.html ◆ ResultListVerbose.html ◆ PrintResultTemplate.html Localized versions for multiple languages can also be used. See “Working with Multiple Languages” on page 221. **Syntax**: template=*filename* **Example**: template=ResultList.html

Parameter Name	Value	Description
theme	String	The name of the theme, or directory within the templates directory, where a complete set of search and print templates are stored. **Syntax**: theme=*theme_name* **Example**: theme=Intranet
showfirsthit	Boolean	If true, rather than displaying the search results page, this parameter automatically goes to the URL of the first hit on the current page. **Syntax**: showfirsthit=*value* **Example**: showfirsthit=True **Default**: False
retfield	String	Lets you determine the level of detail given about each result item. The fewer the details, the faster a search is returned to a user. Field names include title, author, URL, changedate, language, summary, relevance, index, format, and filelength. **NOTE:** Type these fields exactly as they appear. To specify more than one field, use the RETFIELD parameter and value, separated by an ampersand (&) as in the following: retfield=title&retfield=author **Syntax**: retfield=*field_name* **Example**: retfield=title

Parameter Name	Value	Description
buttonpressed	String	A button pressed by the user. If this value is part of the query, then a response message should not be sent to the client. Options include Yes, No, OK, Cancel, Continue, Ignore, Retry, Prev, Next, and Help. **Syntax**: buttonpressed=*button_name* **Example**: buttonpressed=Cancel
gettotalhits	Boolean	Lets you enable or disable the total number of hits calculation. For example, if you set the GETTOTALHITS parameter to FALSE, the Total Number of Hits label on the results page will display 0 (zero). Setting this parameter to TRUE will show the total number of hits found during the search. In some complex situations this can save valuable processing time. **Syntax**: gettotalhits=*value* **Example**: gettotalhits=false **Default**: True
encoding	String	Specifies the character set encoding used to encode the search request itself. **Syntax**: encoding=*value* **Example**: encoding=Shift_JIS **Default**: UTF-8
retencoding	String	Lets you specify the character set encoding to be used by the next results page returned to the user. **Syntax**: retencoding=*character_set_encoding* **Example**: retencoding=iso-8859-1 **Default**: UTF8

Parameter Name	Value	Description
sortbydate	String	Sorts the Total Search Results list by date, ignoring the normal relevance ordering. **Syntax**: sortbydate=*value* **Example**: sortbydate=true **Default**: True
relevance	String	Tells Web Search whether or not to sort the search results by relevance. Turning this feature off is a potential speed gain since the sort algorithm will then not run. **Syntax**: relevance=*value* **Example**: relevance=false **Default**: true
sortkeys	Integer	Lets you specify the number of sort fields that should be used to sort the search results. **Syntax**: sortkeys=*number* **Example**: sortkeys=1

Parameter Name	Value	Description
sortfield*number*	String	Allows you to specify the fields on which to sort the search results returned in a results page. Grouped together with the sortorder query parameter by adding a *number* to the end of the parameter name. Field names include title, author, URL, changedate, language, summary, relevance, index, format, and filelength. **IMPORTANT:** Type these fields *exactly* as they appear above. **Syntax**: sortfield*number=field_name* **Example**: sortfield1=title
sortorder*number*	Integer	Lets you specify the alphanumeric ordering of search result items (hits). Grouped together with the sortfield query parameter by adding a *number* to the end of the parameter name. Options include the following: **0**= Ascending **1**= Descending **2**= Default for each field. **Syntax**: sortorder*number=number* **Example**: sortorder1=0

19 Internationalizing Your Search Solution

NetWare® Web Search Server is capable of handling search queries, search results, templates, and Web content in many languages and character sets. Web Search can auto-detect languages and character sets, but to ensure a complete international search solution, you must identify language, country, and character information throughout your Web Search implementation.

This chapter discusses all of the issues related to supporting multiple languages from a single search solution.

Working with Multiple Languages

Customizing your search solution is important only if you want to let your users conduct language-specific searches. You specify the language of a template by inserting a language identifier in the META tag of your templates or HTML files. The language identifier can also be used in Search Results pages to let users quickly recognize the search results that interest them.

NetWare Web Search Server also lets Web clients specify their locale at the time the search query is entered. The default Search page illustrates this feature by auto-detecting a user's locale and selecting the appropriate language from the Display Language drop-down list. This selection sends two parameters to the Web Search Server: language and country. The country parameter is almost always blank. The search engine uses this information to find locale-specific versions of the templates used to return search results.

To specify the language of a template or of any HTML content that gets indexed as part of your virtual search server, you must enter a language identifier within an HTML file’s header section. For example, if you wanted to identify a Russian template, you would add the following META tag:

```
<meta http-equiv="Content-Language" content="ru">
```

In some cases, such as Traditional and Simplified Chinese, you will need to use the two-character, uppercase country codes. For example:

```
<meta http-equiv="Content-Language" content="zh-TW">
```

```
<meta http-equiv="Content-Language" content="zh-CN">
```

The first line of the example indicates the Chinese language (ZH) and the geographic location as Taiwan. The second line of the example indicates the Chinese language (ZH) but China as the geographic location.

This combination of language and country codes is called a *locale*. For more information about locales, refer to Table 17 on page 230.

Specifying Locales within Template Filenames

NetWare Web Search Server consists of three primary servlets: SearchServlet, PrintServlet, and AdminServlet. Each servlet returns information to the Web client using server-side templates. Templates are stored on at *volume*:*searchroot*\TEMPLATES. For more information about templates, see Chapter 16, "Understanding Templates," on page 183.

After determining a Web client's locale, Web Search attempts to locate a matching search result template. That is, each of the Web Search services automatically attempts to locate a version of the requested template that most closely matches the Web client's locale.

IMPORTANT: NetWare Web Search cannot find locale-specific templates without the two-character language code and the optional two-character country code. See Table 17 on page 230 for more information about language code syntax.

For example, if a Web client requests to see search results using the ResultListTemplate.html file and the client is a Chinese language user from Taiwan and the server is Russian, then Web Search will try to find a Chinese-Taiwan version of the template first (ResultListTemplate_zh_TW.html) because that exactly matches the client's language and country. The following table lists the template names the system would look up in this example in order of priority.

Template Name	What Web Search Concludes
1. ResultListTemplate_zh_TW.html	Specific client locale
2, ResultListTemplate_zh.html	Simplified client locale
3. ResultListTemplate.html	Client requested name
4. ResultListTemplate_ru.html	Specific server locale (no simplified versions)
5. ResultListTemplate_en.html	English language version

Template Name	What Web Search Concludes
6. ResultListTemplate.html	Up to the first underscore (_)

If this scenario were reversed so that the search client was Russian and the server was Chinese (Taiwan), and the client requested the ResultListTemplate_ja.html template, then the lookup order would follow the order shown in the following table.

Template Name	What Web Search Concludes
1. ResultListTemplate_ja_ru.html	Specific client locale (no simplified versions)
2. ResultListTemplate_ja.html	Client requested name
3. ResultListTemplate_ja_zh_TW.html	Specific server locale
4. ResultListTemplate_ja_zh.html	Simplified server locale
5. ResultListTemplate_ja_en.html	English language version
6. ResultListTemplate.html	Up to the first underscore (_)

All templates undergo this rigorous lookup system. Once a template is located, its name is stored and associated with the original client locale so that all subsequent requests for that template from the same locale automatically find the template without performing the same rigorous lookup.

No further lookups are attempted for that combination of client locale and template name until the NetWare Web Search Server is restarted. If all template lookups fail, then an error message is returned to the client performing the search.

Understanding Character Set Encodings

A character set is a grouping of alphabetic, numeric, and other characters that have some relationship in common. For example, the standard ASCII character set includes letters, numbers, symbols, and control codes that make up the ASCII coding scheme.

A *character set encoding* is the mapping of a character set to a value that can be understood and processed by a computer.

NetWare Web Search relies on character set encodings to identify the characters used when performing a search, reading a template, posting results to a Web browser, or indexing Web-based content. If the encoding information is missing in any of these areas, NetWare Web Search uses the default encodings identified in the SearchServlet and PrintServlet properties files. You can modify these settings using NetWare Web Search Manager.

Because most languages have several encodings that their character sets are identified by, NetWare Web Search Server supports a wide variety of character set encodings and encoding aliases.

Some examples of character set encodings include iso-8859-1, shift_jis, big5, and latin2. The official list of registered encodings is available from the Internet Assigned Numbers Authority (see Table 17 on page 230). These are the official names for character sets that can be used in the Internet and can be referred to in Internet documentation. However, not all IANA-registered character set encodings are supported by NetWare Web Search Server. Refer to Table 17 on page 230 for a list of encodings and encoding aliases that are supported by NetWare Web Search Server.

Unicode and UTF8

Unicode is a 16-bit character encoding standard developed by the Unicode Consortium. By using two bytes to represent each character, Unicode enables almost all of the written languages of the world to be represented using a single character set. Unicode does not require any special processing to access any character in any language.

This makes Unicode very easy to use when processing text from multiple languages and scripts. This is the reason NetWare Web Search converts all external files into Unicode for processing.

As already mentioned, Unicode is two bytes wide for all characters. Although this is ideal for computer processing, it doubles the size of all single-byte languages. This has a significant impact on Internet performance. For this reason, NetWare Web Search also supports an alternate representation of Unicode known as UTF-8. UTF-8 is a Unicode Transformation Format that uses sequences of 1 to 6 bytes to represent all the characters in the Unicode standard. Most notably, ASCII characters are transmitted without any conversion at all. This means that most Internet content is already in the UTF-8 representation. Many Asian languages, however, require three bytes per character in the UTF-8 format. Other languages can require up to six bytes to represent each of their characters.

You will have to decide if Unicode or UTF-8 best meets your needs when creating HTML content, Web Search templates, or search pages.

Search Encodings

The only encodings NetWare Web Search currently supports when performing a search are Unicode and UTF-8. Therefore, any page that allows Web users to enter a search must ensure that the results are passed to the server in one of these two formats. See "Template Encodings" on page 227 for more information.

To pass Unicode characters to NetWare Web Search, use the syntax %uHHHH, where

- Percent sign (%) is used as the CGI escape character
- Lowercase letter U (u) indicate that the subsequent 4 characters represent a Unicode value.
- Four uppercase H letters (HHHH) indicate four hexadecimal characters (0-9, A-F)

To pass UTF-8 characters to NetWare Web Search, just use normal ASCII characters or the syntax %HH... for all other characters, where

- % is the CGI escape character
- HH indicates two hexidecimal characters (0-9, A-F)
- . . . indicates additional %HH groupings that might be required to properly transmit a character

HINT: If the encoding of the page containing a search form is already set to UTF-8 or Unicode, most browsers automatically transmit the entered search text correctly using the designated encoding.

By default, NetWare Web Search uses UTF-8 in its sample search pages.

Response Encodings

One of the many parameters that can be sent when conducting a search is the encoding that should be used when returning the results back to the browser. All NetWare Web Search encodings listed in Appendix B, "Combined Character Sets for Use with NetWare Web Search," on page 235 can be used.

If the search result page contains the ability to refine or redo the search, then the response encoding can significantly impact the possible characters that can

be entered when conducting the next search from this page. For example, if the user requests results in the iso-8859-1 encoding (HTML's default), then only iso-8859-1 characters can be entered in the subsequent search from that page. Other characters can still be sent to the Web Search services using the %uHHHH and %HH formats, but the browser will not allow users to enter normal text characters other than that supported by iso-8859-1.

Although Web Search can return search results from many languages, some characters found in titles and descriptions might be returned as question marks (?) indicating that these characters are not available in the current response encoding. If a character can be represented in the current encoding but a font is not available, many browsers will substitute an alternate character such as an empty box character. Once the appropriate fonts have been installed, these characters will then display properly.

By default, NetWare Web Search returns all search, print, and administration pages in UTF-8.

HTML Encodings

Since HTML content can contain text written in many character sets, all HTML files need to include a tag that identifies the character set encoding. To identify the encoding of an HTML file (or search template), use the following META tag at the top of the file's header section:

```
<meta http-equiv="Content-Type" content="text/html;
charset=Shift_JIS">
```

In this example, you would replace Shift_JIS with the appropriate Internet Assigned Numbers Authority (IANA)-assigned encoding value.

It is very important that the CHARSET value accurately represent the character set encoding that was actually used when the HTML Web content or Web Search template was created. A correct entry allows Web Search to accurately interpret and convert the characters in the document. An incorrect entry prevents Web Search from being able to read the characters as valid data in the authored language.

IMPORTANT: Improperly identified characters result in garbled text. In some cases, the Web-based content cannot be properly indexed or printed. In the most severe cases, the document being read might produce a server-side exception, which will ultimately discontinue processing the document and perhaps the entire current operation.

Because Web Search is Unicode-based, when reading templates or when indexing or printing HTML content, all character encodings are converted from their source encoding to Unicode for internal processing.

During indexing, if a document contains characters not supported by the designated encoding, if the document doesn't have an encoding designation, or if the designation is inaccurate, the indexer will do its best to recover. But if it cannot, it might index the information incorrectly or quit indexing that page entirely.

When reading a template file, Web Search might automatically cease processing the file if it contains any characters not supported by the current encoding. It will try to ignore the invalid text and continue, but this might not be possible.

When displaying search results or when printing HTML content, any character that does not match the specified response encoding will receive a question mark (?) in its place when rendered at the browser. Although some characters are properly supported by the current encoding, the browser might not have the required fonts to display the characters. In this case, users might see square boxes representing these characters. This is an indication that the valid character reached the browser, but the operating system could not provide a font to properly render the character. The user would than have to either change fonts or install the correct fonts in order to properly display the characters.

HINT: If a document does not contain a CHARSET encoding value, the default encoding for HTML documents is ISO-8859-1, also known as Latin1. The default encoding for plain text documents is US-ASCII.

Web Search also allows administrators to define the default encodings for templates, HTML content when printing, and search and print responses. Refer to the NetWare Web Manager Help for information about changing the default encodings.

Template Encodings

All HTML documents should include a Content-Type META tag identifying their character set encodings. The character set encoding allows HTML Web clients (or browsers) to understand the contents of the file. This tag is also used by browsers to automatically switch their display system and fonts to correctly show the Web page's contents. This lets users surf the World Wide Web without having to constantly change their display system as they encounter content from various languages and characters sets.

However, because NetWare Web Search lets administrator specify both template encodings and response encodings, browsers might get confused when presented with the valid response encoding in the HTTP header and one or more alternate encodings from the Content-Type META tags within the file that was part of the original Web Search template.

NOTE: $$IncludeFile[ ] templates can also contain their own Content-Type meta tags.

To solve this problem, NetWare Web Search allows placing the Content-Type META tag specifying the template's encoding within an HTML comment. This effectively obscures the original template encoding from the browser, but still allows Web Search to read the encoding when the template file is processed.

A sample Web Search template is illustrated below. The Content-Type META tag has been hidden inside of an HTML comment. This template can be embedded within other templates using the $$IncludeFile[ ] template variable without affecting Web Search's ability to distinguish between the various encodings. This file can also be processed and then sent to a user's Web browser without conflicting with the response encoding provided by Web Search in the HTTP response headers.

```
<html>
<head>

<!-- Note that the HTML encoding command (meta tag) is hidden
within a comment so that it does not affect a user's browser
display. -->

<!-- The actual encoding used when sending this file to the
user is controlled by the response encoding -->

<!-- <META HTTP-EQUIV="Content-Type" CONTENT="text/html;
charset=iso-8859-1"> -->

</head>
<body>

Template data here.

</body>
</html>
```

Encoding Issues When Printing

When NetWare Web Search processes a print request, it gathers the entire contents of each file and builds an appended print job page, one file after another. Each file can contain its own Content-Type META tag identifying its encoding. Each file's encoding will be used by Web Search to convert that file into Unicode before being sent out using the response encoding.

Unfortunately, all of these encoding META tags might confuse the browser's display system. While Web Search has already properly converted the files into a single response encoding, the browser sees the Content-Type META tags which direct it to do something else, and gets confused.

The way to solve this problem is to create a print results template that contains a Content-Type META tag encoding at both the top and bottom of the file, before and after the various documents get printed. All current browsers take either the first Content-Type META tag that they encounter or the last. Constructing a print template with both satisfies all browsers.

Languages Included in the Default Templates

There are additional search and print templates for each of the following languages:

- Chinese (Traditional and Simplified)
- English
- French
- German
- Italian
- Japanese
- Korean
- Portuguese
- Russian
- Spanish

Templates are stored at *volume*:*searchroot*\TEMPLATES.

Where to Go from Here

The following table lists additional resources for learning more about locales, country and language codes, and encodings.

Table 17 Additional Information Resource

Component	Resource Location
Language and country codes (locale)	RFC1766 (http://www.ietf.org/rfc/rfc1766.txt) **NOTE:** While RFC1766 uses the hyphen character (-) to separate language and country information, Web Search uses the underscore character (_) in order to conform to the Java convention. ISO639 (http://www.ics.edu/pub/ietf/http/related/iso639.txt) ISO3166 (http://www.chemie.fu-berlin.de/diverse/doc/ISO_3166.html)
Character sets	Internet Assigned Numbers Authority (IANA) Character Set registry (http://www.isi.edu/in-notes/iana/assignments/character-sets)
Unicode	Unicode Consortium home page (http://www.unicode.org/)
UTF-8	"UTF-8: A Transformation Format of ISO10646" (ftp://nis.nsf.net/internet/documents/rfc/rfc2279.txt)

IV Appendixes

This section contains additional information and reference materials related to several Web service components:

- Appendix B, “Combined Character Sets for Use with NetWare Web Search,” on page 235
- Appendix C, “HTTP Methods and eDirectory Trustee Requirements,” on page 253
- Appendix D, “Managing Users and Groups Using Local Database or LDAP Modes,” on page 255
- Appendix E, “Controlling Access to Your Server Using Local Database or LDAP Modes,” on page 273
- Appendix F, “Port Number Assignments,” on page 295

A Troubleshooting NetWare Web Search

This appendix provides some troubleshooting topics that can help you overcome search and print performance issues.

Troubleshooting

Characters of descriptions or titles appear as intelligible characters

Possible Cause: You've probably indexed documents written in multiple languages and encodings. Web Search can index most of the world's languages and encodings. However, Web Search needs to know the encoding of each document.

Possible Cause: Some of your documents were probably not tagged with an encoding or were incorrectly tagged.

Action: Make sure all of your documents contain the correct Content-Type META tag. If your international documents do not contain a Content-Type META tag, either add it or use the Encoding (If Not in META Tags) index definition option to specify the default encoding.

Several titles or descriptions contain the same text

Possible Cause: If search results include duplicate titles or descriptions, your description fields (description, summary, or abstract) might include boilerplate information.

Action: The more accurate your META tag description fields are, the better your search results will be. Where possible, consider adding descriptions to your document's META tags.

Possible Cause: It could also be that you have indexed the same document more than once, or several links throughout your Web site might point to the same document but do so using different character cases each time.

Action: To solve the latter problem, try using the URLs Are Case Sensitive option to direct Web Search to turn off case-sensitive crawling. Also, remove any duplicate backup files you might have and exclude any backup directories from your index definition.

Some titles are returned as the URL of the document instead

Possible Cause: Web Search pulls document titles from within each document that it indexes. If your document doesn't have a title, Web Search uses the URL or path of the document instead. If the URL is unavailable, a Title Unavailable message is returned.

Action: Make sure all of the documents you index have specifically defined titles.

Additional Assistance

If the problem you are working with doesn’t appear in this appendix, visit the Novell® Support Connection Web site (http://support.novell.com).

B Combined Character Sets for Use with NetWare Web Search

The following tables list the character set encoding names and aliases that Web Search recognizes when indexing, searching, displaying, or printing files. This information is a subset of the character names registered by the Internet Assigned Numbers Authority (IANA).

Whenever possible, the items listed in the first column of each table are the preferred MIME names listed in the Internet Assigned Numbers Authority (IANA) Character Sets registry. If a preferred MIME name is not available, items in the first column represent the primary registered names.

Items in the second column of each table are aliases which are also at times used to identify that encoding.

Note that not all aliases exactly represent the parent encoding under which they are listed. In these cases, they overlap significantly enough that they will be handled identically by the various NetWare® Web Search engines.

HINT: Character encodings appear in the exact case specified in the Internet Assigned Numbers Authority (IANA) Character Sets registry. Some uses of these encodings are case sensitive. However, NetWare Web Search ignores the case of these encodings.

ASCII Character Set

Preferred MIME Name or Primary Registered Name	Encoding Names
US-ASCII (MIBenum: 3)*	ANSI_X3.4-1968
	ANSI_X3.4-1986
	ASCII
	ascii7
	iso_646-us
	ISO646-US
	ISO_646.irv:1991
	iso-ir-6
	646
	us
	IBM367
	cp367
	csASCII
IBM437 (MIBenum: 2011)	ibm-437
	cp437
	437
	csPC8CodePage437

* A MIBenum is a record number corresponding to an entry in IANA's Management Information Base.

Arabic Character Set

Preferred MIME Name or Primary Registered Name	Encoding Aliases
ISO-8859-6 (MIBenum: 9)	ISO_8859-6:1987
	ISO_8859-6
	iso8859-6
	iso8859_6
	8859_6
	IBM1089
	ibm-1089
	cp1089
	1089
	iso-ir-127
	ECMA-114
	ASMO-708
	arabic
	csISOLatinArabic
Windows-1256 (MIBenum: 2256)	cp1256
	win1256
	ms1256

Chinese (Simplified) Character Set

Preferred MIME Name or Primary Registered Name	Encoding Aliases
gb2312 (MIBenum: 2025)	csGB2312
gb_2312-80 (MIBenum: 57)	iso-ir-58
	chinese
	csISO58GB231280
	gb2312-80
	gb2312-1980
	gb-2312-80
gbk	GBK
	windows-936
	ms936
	cp936
	cp-936
euc-cn	EUC_CN
	euccn
	euc-gb

Chinese (Traditional) Character Set

Preferred MIME Name or Primary Registered Name	Encoding Aliases
big5 (MIBenum: 2026)	Big5
	windows-950
	win950
	ms950
	csBig5
IBM950 (MIBenum: ????)	ibm-950
	cp950
	cp-950
	950

Cyrillic Character Set

Preferred MIME Name or Primary Registered Name	Encoding Aliases
ISO-8859-5 (MIBenum: 8)	ISO_8859-5:1988
	ISO_8859-5
	iso8859-5
	iso8859_5
	8859-5
	iso-ir-144
	IBM915
	ibm-915
	cp915
	915
	cyrillic
	csISOLatinCyrillic
KOI8-R (MIBenum: 2084)	koi8_r
	koi8
	cp878
	cp-878
	csKOI8R
Windows-1251 (MIBenum: 2251)	win1251
	cp1251
	ms1251

European Character Set

Preferred MIME Name or Primary Registered Name	Encoding Aliases
Windows-1252 (MIBenum: 2252)	cp1252
	ms1252
	win1252
	ansi
	ansi-1252
ISO-8859-1 (MIBenum: 4)	ISO_8859-1:1987
	ISO_8859-1
	iso8859-1
	iso8859_1
	8859_1
	iso-ir-100
	IBM819
	ibm-819
	CP819
	819
	l1
	latin1
	csISOLatin1

Preferred MIME Name or Primary Registered Name	Encoding Aliases
ISO-8859-2 (MIBenum: 5)	ISO_8859-2:1987
	ISO_8859-2
	iso8859-2
	iso8859_2
	8859_2
	iso-ir-101
	IBM912
	ibm-912
	cp912
	912
	l2
	latin2
	csISOLatin2
ISO-8859-3 (MIBenum: 6)	ISO_8859-3:1988
	ISO_8859-3
	iso8859-3
	iso8859_3
	8859-3
	iso-ir-109
	IBM913
	ibm-913
	cp913
	913
	l3
	latin3
	csISOLatin3

Preferred MIME Name or Primary Registered Name	Encoding Aliases
ISO-8859-4 (MIBenum: 7)	ISO_8859-4:1988
	ISO_8859-4
	iso8859-4
	iso8859_4
	8859-4
	iso-ir-110
	IBM914
	ibm-914
	cp914
	914
	l4
	latin4
	csISOLatin4
Windows-1250 (MIBenum: 2250)	cp1250
	ms1250
	win1250
IBM850 (MIBenum: 2009) (UNICODE)	ibm-850
	cp850
	850
	csPC850Multilingual
IBM852 (MIBenum: 2010)	ibm-852
	cp852
	852
	csPCp852

Preferred MIME Name or Primary Registered Name	Encoding Aliases
IBM860 (MIBenum: 2048)	ibm-860
	cp860
	860
	csIBM860
IBM863 (MIBenum: 2050)	ibm-863
	cp863
	863
	csIBM863
IBM865 (MIBenum: 2052)	ibm-865
	cp865
	865
	csIBM865

Greek Character Set

Preferred MIME Name or Primary Registered Name	Encoding Aliases
ISO-8859-7 (MIBenum: 10)	ISO_8859-7:1987
	ISO_8859-7
	iso8859-7
	8859_7
	IBM813
	ibm-813
	cp813
	813
	iso-ir-126
	ELOT_928
	ECMA-118
	greek
	greek8
	csISOLatinGreek
Windows-1253 (MIBenum: 2253)	cp1253
	ms1253
	win1253

Hebrew Character Set

Preferred MIME Name or Primary Registered Name	Encoding Aliases
ISO-8859-8 (MIBenum: 11)	ISO_8859-8:1988
	ISO_8859-8
	iso8859-8
	8859_8
	ibm916
	ibm-916
	cp916
	916
	iso-ir-138
	hebrew
	csISOLatinHebrew
Windows-1255 (MIBenum: 2255)	win1255
	cp1255
	ms1255

Japanese Character Set

Preferred MIME Name or Primary Registered Name	Encoding Aliases
ISO-2022-JP (MIBenum: 39)	iso2022-jp
	iso-2022-jis
	junet
	jis
	jis_encoding
	csJISEncoding
	csISO2022JP
ISO-2022-JP-2 (MIBenum: 40)	iso-2022-jp2
	csISO2022JP2
Shift_JIS (MIBenum: 17/2024)	sjis
	shift-jis
	ShiftJis
	x-sjis
	x-shift-jis
	windows-31j
	csWindows31J
	ms932
	cp932
	win932
	windows-932
	MS_Kanji
	csShiftJIS
	pck
	\u30b7\u30d5\u30c8\u7b26\u53f7\u5316\u8868\u73fe

Preferred MIME Name or Primary Registered Name	Encoding Aliases
EUC-JP (MIBenum: 18)	Extended_UNIX_Code_Packed_Format_for_Japanese eucjp x-euc-jp euc_jpnew 10/18/99 x-eucjp eucjis csEUCPkdFmtJapanese

Korean Character Set

Preferred MIME Name or Primary Registered Name	Encoding Aliases
euc-kr (MIBenum: 38)	euc_kr
	euckr
	csEUCKR
ks_c_5601-1987 (MIBenum: 36)	ks_c_5601-1989
	ksc5601-1987
	ksc5601_1987
	ksc_5601
	ksc5601
	5601
	korean
	csKSC56011987
IBM949 (MIBenum: ????)	ibm-949
	cp949
	cp-949
	949
Windows-949 (MIDenum: ????)	win949
	ms949

Thai Character Set

Preferred MIME Name or Primary Registered Name	Ecoding Aliases
IBM874 (MIBEnum: ????)	ibm-874
	cp874
	874
Windows-874	win874
	ms874

Turkish Character Set

Preferred MIME Name or Primary Registered Name	Encoding Aliases
ISO-8859-9 (MIBenum: 12)	ISO_8859-9:1989
	ISO_8859-9
	iso8859-9
	8859_9
	ibm920
	ibm-920
	cp920
	920
	iso-ir-148
	l5
	latin5
	csISOLatin5
Windows-1254 (MIBenum: 2254)	win1254
	cp1254
	ms1254

Vietnamese Character Set

Preferred MIME Name or Primary Registered Name	Ecoding Aliases
Windows-1258 (MIBenum: 2258)	win1258
	ms1258
	cp1258
	cp-1258

C HTTP Methods and eDirectory Trustee Requirements

HTTP access to a file or resource in Novell® eDirectory™ mode is evaluated using NetWare® file system trustee assignments, depending on the HTTP method used. The table below defines the NetWare file system trustee assignments required to grant access to Web resources using specified HTTP methods.

This table applies only while running the Enterprise Web Server in eDirectory mode.

HTTP Method	NetWare Trustee Assignment Required for Access
COPY	Read on source, Create on destination
DELETE	Erase
EDIT	Write
GET	Read
HEAD	File Scan
INDEX	File Scan
MKDIR	Create
MOVE	Erase and Read on source, Create on destination
POST	Read on the CGI executable file
PUT	Create on parent directory if file is being created, or WRITE if file is being replaced

D Managing Users and Groups Using Local Database or LDAP Modes

This appendix covers creating and managing User and Group objects while running in local database or LDAP modes. If you are running the Enterprise Server in Novell® eDirectory™ mode, refer to Chapter 7, "Using a Directory Service to Control User Access to Network Resources," on page 85.

HINT: If you are using eDirectory mode, you can use ConsoleOne™ to manage users and groups. If you need to access eDirectory User and Group objects from a remote location, use NetWare® Web Manager. Web Manager lets you add and remove users and groups and manage access rights.

The procedures outlined in this section refer exclusively to Web Manager as the directory managment tool for managing User and Group objects while in local database or LDAP modes. For more information about using eDirectory, refer to the *Novell eDirectory 8.6 Administration Guide*.

Creating Users

1. From the NetWare Web Manager home page, click Users and Groups > *organization* > New User.
2. In the appropriate fields, enter the requested information. At a minimum, you must specify the user's
   - Surname
   - User ID

   After entering a user's first and surnames, the ID is automatically generated in the User ID field. You can replace this user ID with an ID of your own choice if you want.

   IMPORTANT: The user ID must be unique. NetWare Web Manager ensures that it is unique by searching the entire directory begining at the search base (base DN)

to see if the user ID is in use. However, if you use the ldapmodify command line utility to create a user, be aware that the utility does not ensure unique user IDs. If duplicate user IDs exist in your directory, the effected users will not be able to authenticate to the directory.

3 Click Create User to add a user.

For information on editing users, see "Managing Users" on page 257.

Additional Information about User Entries

The following information might be of interest to the network administrator concerning creating user entries:

- User entries use the inetOrgPerson, organizationalPerson, and person object classes. For more information on how these are used, search the Novell Support Connection Web site (http://support.novell.com).
- By default, the distinguished name for users is as follows:

  ```
  cn=full name, ou=organization, ...,o=base organization,
    c=country
  ```

  For example, if a user entry for Sam Warden is created within the Organizational Unit Engineering, and the directory's suffix is O=Ace Industry, C=US, then the person's DN is

  ```
  CN=Sam Warden, OU=Engineering, O=Ace Industry, C=US
  ```

  However, you can change this format to a UID-based distinguised name.
- Suffixes are optional if you are using the local directory. If you did not configure a suffix for your local directory, then you literally use the string " " (quote quote) to represent the search base on calls to ldapsearch.
- The values on the user form fields are stored as the following LDAP attributes:

User Field	Corresponding LDAP Attribute
Given Name	givenName
Surname	sn
Full Name	cn
User ID	uid
Password	userPassword
E-Mail Address	mail

The following fields are also available when editing the user entry:

User Field	Corresponding LDAP Attribute
Title	title
Telephone	telephoneNumber

Managing Users

From the Manage Users form you can

- Find user entries
- Change user attribute values
- Change the user's password
- Manage the user's licenses
- Rename the user's entry
- Delete the user's entry
- Change some, but not all, product-specific information. Additional forms are added to this area that allow you to manage product-specific information. For example, when a Web server is installed under NetWare Web Manager, then an additional form is added that allows you to edit settings specific to that server.

The following sections describe these activities in detail.

For more information regarding user entries when using a directory server, see "Additional Information about User Entries" on page 256.

Finding User Objects

If you are running in LDAP or local directory modes and you need to edit a User object, you can quickly search for and retrieve a User object.

1. From the Web Manager home page, click NetWare Enterprise Web Server *servername* > Users and Groups.
2. In the Find User field, type some descriptive value for the entry that you want to edit. You can enter any of the following in the search field:
   - A name: Type a full name or a partial name

- A user ID
- A telephone number
- An e-mail address: Any search string containing an at (@) symbol is assumed to be an e-mail address
- An asterisk (*): Type an asterisk to see all of the entries currently in your directory (or achieve the same effect by simply leaving the field blank)
- Any LDAP search filter: Type a search filter to see any string that contains an equal sign (=) that is considered a search filter

3. In the Format field, select either On-Screen or Printer.
4. Click Find.

The Find All Users Whose Field

This field allows you to build a custom search filter. Use this field to narrow down the search results returned by Find User.

Find All Users Whose provides the following search criteria:

1. The left drop-down list lets you specify the attribute that the search will be based on.

   The options include the following:

   - Full Name
   - Last Name
   - User ID
   - Phone Number
   - E-Mail Address

2. The center drop-down list lets you select the type of search you want to perform.

   The options include the following:

   - Contains: Entries with attribute values containing the specified search string are returned.
   - Is: Use this option when you know the exact value of an user's attribute.

- Isn't: Returns all the entries whose attribute value does not exactly match the search string. For example, if you want to find all the users in the directory whose names are not *Sam Warren*, use this option.

  NOTE: Using this option can cause an extremely large number of entries to be returned to you.

- Sounds Like: Causes an approximate, or phonetic, search to be performed. Use this option if you know an attribute's value, but you are unsure of the spelling. For example, if you are not sure if a user's name is spelled Sarret, Sarette, or Sarett, use this option.
- Starts With: Returns all the entries whose attribute value starts with the specified search string.
- Ends With: Returns all the entries whose attribute value ends with the specified search string.

3. In the right-most text field, type your search string.

Editing User Information

1. From the Web Manager home page, click NetWare Enterprise Web Server *servername* > Users and Groups > Manage Users.
2. Find the user entry you want.

   See "Finding User Objects" on page 257 for more information.
3. Edit the field corresponding to the attribute that you want to change.
4. Click Save Changes.

HINT: You might want to change an attribute value that is not displayed by the edit user form. In this situation, use the ldapmodify command line utility.

You can change the user's first, last, and full name field from this form; but to fully rename the entry (including the entry's distinguished name), you need to use the Rename User form. For more information on how to rename an entry, see "Renaming Users" on page 260.

Managing User Passwords

The password you set for user entries is used by the various Web technologies for user authentication. You can create, change, or disable a password.

1. From the Web Manager home page, click NetWare Enterprise Web Server *servername* > Users and Groups > Manage Users.
2. Find the user entry you want.

   See "Finding User Objects" on page 257 for more information.
3. At the top of the User Edit form, click Password.
4. To create or change a password, type the new password and the confirmation password, and then click Set Password.
5. To disable the password, click Disable Password.

   This prevents the user from logging in to a Web server, for example, without deleting the user's directory entry. You can reinstate the password by using the Password Management form to enter a new password.
6. Click General to return to general user information.

Managing User Licenses

To track which Web technology or service your users are licensed to use, do the following:

1. From the Web Manager home page, click NetWare Enterprise Web Server *servername* > Users and Groups > Manage Users.
2. At the top of the User Edit form, click Licenses.
3. Select the Web technology that you want the user to be able to use.
4. Click Save Changes.
5. Click General to return to general user information.

Renaming Users

1. From the Web Manager home page, click NetWare Enterprise Web Server *servername* > Users and Groups > Manage Users.
2. Select the user entry you want.

   See "Finding User Objects" on page 257 for more information.

3 Click Rename User.

4 Type the new name.

If you are using common name-based DNs, specify the user's full name. If you are using UID-based distinguished names, enter the new UID value that you want to use for the entry.

5 Type the modified given name, surname, full name, or UID in the applicable fields as is appropriate to match the new distinguished name for the entry.

If you are using common name-based distinguished names, and you change the distinguished name to use a new common name, then you should make sure that this new common name is listed as the first choice in the list of full names. This ensures that the appropriate name is displayed when a list showing this entry is generated.

HINT: The rename feature changes only the user's name; all other fields are left intact. In addition, the user's old name is still preserved so searches against the old name will still find the new entry.

When you rename a user entry, you only change the user's name; you cannot use the rename feature to move the entry from one Organizational Unit to another. For example, suppose you have

- Organizational Units for Marketing and Accounting
- An entry named Sam Warren under the Marketing Organizational Unit

You can rename the entry from Samuel Warren to Sam Warren, but you cannot rename the entry such that Samuel Warren under the Marketing Organizational Unit becomes Samuel Warren under the Accounting Organizational Unit.

6 To return to the general information form, click General.

Removing Users

1 From the Web Manager home page, click NetWare Enterprise Web Server *servername* > Users and Groups > Manage Users.

2 Find the user entry you want.

See "Finding User Objects" on page 257 for more information.

3 Click Delete User > OK.

Creating Groups

1. From the Web Manager home page, click NetWare Enterprise Web Server *servername* > Users and Groups > New Group.
2. In the Group Name field, type the group's name.

   You can optionally add a description for the group in the Description field.
3. Click Create Group to add the group and immediately return to the New Group form.
4. Click Create and Edit Group to add the group and then proceed to the Edit Group form for the group you have just added.

   For information on editing groups, see "Editing Group Attributes" on page 264.

Managing Groups

From the Group Edit form you can

- Find groups
- Change group attributes
- Add and delete owners of the group
- Add and delete See Also information
- Add and delete members of the group
- Rename the group
- Delete the group
- Change the group's description

The following sections describe these activities in detail.

Finding Group Entries

1. From the Web Manager home page, click NetWare Enterprise Web Server *servername* > Users and Groups > Manage Groups.
2. In the Find Group field, type the name of the group that you want to find.

   You can enter any of the following in the search field:

   - A name: Type a full name or a partial name
   - An asterisk (*): Type to see all of the groups currently residing in your directory
   - Any LDAP search filter: Type to see any string that contains an equal sign (=) that is considered to be a search filter
3. In the Format field, select either On-Screen or Printer.
4. Click Find.
5. In the resulting table, click the name of the entry you want to edit.

The Find All Groups Whose Field

This field lets you build a custom search filter. Use this field to narrow down the search results.

Find All Groups Whose provides the following search criteria:

1. The left drop-down list lets you specify the attribute that the search is based on.

   The options are

   - Full Name
   - Description
2. In the middle drop-down list, select the type of search you want to perform.

   The options include the following:

   - Contains: Entries with attribute values containing the specified search string are returned.
   - Is: Use this option when you know the exact value of a group's attribute.

- Isn't: Returns all the entries whose attribute value does not exactly match the search string. If you want to find all the groups in the directory whose names do not contain *administrator*, use this option.
- Sounds Like: Causes an approximate, or phonetic, search to be performed. Use this option if you know an attribute's value, but you are unsure of the spelling. For example, if you are not sure if a group's name is spelled Sarret's list, Sarette's list, or Sarett's list, use this option.
- Starts With: Returns all the entries whose attribute values start with the specified search string.
- Ends With: Returns all the entries whose attribute values end with the specified search string.

3. In the right-most text field, type your search string.

Editing Group Attributes

1. From the Web Manager home page, click NetWare Enterprise Web Server *servername* > Users and Groups > Manage Groups.
2. Find the group you want to edit.

   See "Finding Group Entries" on page 263 for more information.
3. In the Group Edit form, change the displayed fields as desired.
4. Click Save Changes.

   HINT: To change an attribute value that is not displayed by the group edit form, use the ldapmodify command line utility.

Adding Group Members

1. From the Web Manager home page, click NetWare Enterprise Web Server *servername* > Users and Groups > Manage Groups.
2. Find the group you want to add members to.

   See "Finding Group Entries" on page 263 for more information.
3. Click Edit under Group Members.

   A new form is displayed that allows you to search for entries. If you want to add user entries to the list, make sure Users is shown in the Find drop-

down list. If you want to add group entries to the group, make sure Group is shown.

4. In the right-most text field, type a search string.

   Type any of the following:

   - A name: Type a full name or a partial name
   - A user ID: Use if you are searching for user entries
   - A telephone number
   - An e-mail address: Any search string containing an at (@) symbol is assumed to be an e-mail address
   - An asterisk (*): Type an asterisk to see all of the entries or groups currently residing in your directory
   - Any LDAP search filter: Type a search filter to see any string that contains an equal sign (=) is considered to be a search filter

5. Click Find and Add to find all the matching entries and add them to the group.

   If the search returns any entries that you do not want add to the group, check the box in the Remove from List column. You can also construct a search filter to match the entries you want removed and then click Find and Remove.

6. When the list of group members is complete, click Save Changes.

Adding Groups to the Group Members List

You can add groups (instead of individual members) to the group's members list. Doing so causes any users belonging to the included group to become a member of the receiving group. For example, if Sam Warren is a member of the Marketing Managers group, and you make the Marketing Managers group a member of the Marketing Personnel group, then Sam Warren is also a member of the Marketing Personnel group.

To add a group to the members list of another group, add the group as if it were a user entry. See "Adding Group Members" on page 264 for more information.

Removing Entries from the Group Members List

To delete an entry from the group members list:

1. From the Web Manager home page, click NetWare Enterprise Web Server *servername* > Users and Groups > Manage Groups.
2. Find the group you want to edit.

   See "Finding Group Entries" on page 263 for more information.
3. Click Edit under Group Members.
4. For each member that you want to remove from the list, check the corresponding box under the Remove from List column.
5. Click Save Changes.

Managing Owners

You manage a group's owners list the same way as you manage the group members list. The following table shows you which section to read for more information.

If you want to	Use the steps in
Add owners to the group	"Adding Group Members" on page 264
Add groups to the owners list	"Adding Groups to the Group Members List" on page 265
Remove entries from the owners list	"Removing Entries from the Group Members List" on page 266

Managing *See Also*s

*See also*s are references to other directory entries that might be relevant to the current group. They allow users to easily find entries for people and other groups that are related to the current group.

You manage see alsos the same way as you manage the group members list. The following table shows you which section to read for more information.

If you want to	Use the steps in
Add users to *See also*s	"Adding Group Members" on page 264
Add groups to *See also*s	"Adding Groups to the Group Members List" on page 265
Remove entries from *See also*s	"Removing Entries from the Group Members List" on page 266

Removing Groups

1. From the Web Manager home page, click NetWare Enterprise Web Server *servername* > Users and Groups > Manage Groups.
2. Find the group you want to delete.

   See "Finding Group Entries" on page 263 for more information.
3. Click Delete Group > OK.

Renaming Groups

1. From the Web Manager home page, click NetWare Enterprise Web Server *servername* > Users and Groups > Manage Groups.
2. Find the group you want to edit.

   See "Finding Group Entries" on page 263 for more information.
3. Click Rename Group.
4. Type the new group name.

When you rename a group entry, you only change the group's name; you cannot use the Rename feature to move the entry from one Organizational Unit to another. For example, suppose you have

- Organizational units for Marketing and Engineering
- A group named Research and Development under the Engineering Organizational Unit.

You can rename the group from Research and Development to Development and Research, but you cannot rename the entry such that Research and Development under the Engineering Organizational Unit becomes Research and Development under the Marketing Organizational Unit.

Creating Organizational Units

1. From the Web Manager home page, click NetWare Enterprise Web Server *servername* > Users and Groups > New Organizational Unit.
2. In the Unit Name field, type the name of the Organizational Unit.
3. In the optional Description field, you can type a description of the unit.
4. Click Create Organizational Unit.

Additional Information about Organizational Units

The following information might be of interest to the directory administrator:

- New Organizational Units are created using the OrganizationalUnit object class.
- The distinguished name for new Organizational Units is of the form:

  ```
  ou=new organization, ou=parent organization, ...,o=base
    organization, c=country
  ```

  For example, if you create a new Organization called Accounting within the Organizational Unit West Coast, and your Base DN is o=Ace Industry, c=US, then the new Organization Unit's DN is

  ```
  ou=Accounting, ou=West Coast, o=Ace Industry, c=US
  ```

Managing Organizational Units

- Find Organizational Units
- Remove Organizational Units
- Edit Organizational Unit attributes
- Rename Organizational Units
- Delete Organizational Units

Finding Organizational Units

1. From the Web Manager home page, click NetWare Enterprise Web Server *servername* > Users and Groups > Manage Organizational Unit.
2. In the Find Organizational Unit field, type the name of the unit you want to find. You can enter any of the following in the search field:
   - A name: Type a full name or a partial name.
   - An asterisk (*): Type to see all of the groups currently residing in your directory.
   - Any LDAP search filter: Type to see any string that contains an equal sign (=) is considered to be a search filter.
3. In the Format field, select either On-Screen or Printer.
4. Click Find.
5. Click the name of the Organizational Unit that you want to find.

The Find All Units Whose Field

This field allows you to build a custom search filter. Use this field to narrow down the search results that are otherwise returned by Find Organizational Unit.

Find All Units Whose provides the following search criteria:

1. The left drop-down list allows you to specify the attribute on which the search will be based.

   The options include the following:

   - Unit name
   - Description
2. In the center drop-down list, select the type of search you want to perform.

   The options include the following:

   - Contains: Entries with attribute values containing the specified search string are returned.
   - Is: Returns the exact value of an Organizational Unit's attribute.
   - Isn't: Returns all the entries whose attribute value does not exactly match the search string. If you want to find all the Organizational

Units in the directory whose name does not contain "Marketing," use this option.

- Sounds Like: Causes an approximate, or phonetic, search to be performed. Use this option if you know an attribute's value, but you are unsure of the spelling.
- Starts With: Returns all the entries whose attribute value starts with the specified search string.
- Ends With: Returns all the entries whose attribute value ends with the specified search string.

3. In the right text field, type your search string.

   For more information on how to find an Organizational Unit entry, see "Finding Organizational Units" on page 269.

Editing Organizational Unit Attributes

1. From the Web Manager home page, click NetWare Enterprise Web Server *servername* > Users and Groups > Manage Organizational Unit.
2. Find the Organizational Unit you want to edit.

   See "Finding Organizational Units" on page 269 for more information.
3. In the Organizational Unit edit form, change the displayed fields as desired.
4. Click Save Changes.

HINT: It is possible that you will want to change an attribute value that is not displayed by the Organizational Unit edit form. In this situation, use the ldapmodify command-line utility.

Renaming Organizational Units

1. Make sure no other entries exist in the directory under the Organizational Unit that you want to rename.
2. From the Web Manager home page, click NetWare Enterprise Web Server *servername* > Users and Groups > Manage Organizational Unit.
3. Find the Organizational Unit you want to edit.

   See "Finding Organizational Units" on page 269 for more information.
4. Click Rename.
5. Type the new Organizational Unit name.

When you rename an Organizational Unit entry, you can only change the Organizational Unit's name; you cannot use the Rename feature to move the entry from one Organizational Unit to another. For example, suppose you have

- Organizational units for Marketing and Engineering
- An Organizational Unit called User Research under the Marketing Organizational Unit

You can rename the entry from User Research to User Validation, but you cannot rename the entry such that User Research under the Marketing Organizational Unit becomes User Research under the Engineering Organizational Unit.

Deleting Organizational Units

1. Make sure no other entries exist in the directory under the Organizational Unit that you want to rename.
2. From the Web Manager home page, click NetWare Enterprise Web Server *servername* > Users and Groups > Manage Organizational Unit.
3. Find the Organizational Unit you want to delete.

   See "Finding Organizational Units" on page 269 for more information.
4. Click Delete > OK.

E Controlling Access to Your Server Using Local Database or LDAP Modes

You can control who accesses the files on your Web site. This appendix discusses the various methods you can use to determine who has access to specific files or directories on your Web site. If you want to control who can configure the Web server itself, see "Securing Web Manager" on page 35.

The NetWare® Enterprise Web server can be secured using either Novell® eDirectory™ or local database modes. While in eDirectory mode, you manage access control through NetWare file system trustees.

Controlling Access Using Native eDirectory Mode

Novell eDirectory offers unparalleled directory services and is the best choice for use with NetWare Web technologies and services. eDirectory is one of the easiest and most powerful directory services available today and is included with NetWare 6. eDirectory is the default directory service mode for use by the NetWare Enterprise Web Server. We recommend that you use eDirectory mode.

For a comparison of eDirectory, local database, and LDAP modes for use with the Enterprise Web Server, see Chapter 7, "Using a Directory Service to Control User Access to Network Resources," on page 85.

Controlling Access with NetWare Web Access Controls

What Is Access Control?

Access control lets you determine who can access the server. There are two options for controlling access:

- User-Group: Requires users to enter a username and password before accessing the server. Or the server can use client authentication by checking an LDAP directory for a security certificate before giving access to a file or set of files on your Web site.
- Host-IP: Requires the user to view your Web site from a specific computer, where the server recognizes the computer by either its hostname or its IP address.

User-Group Authentication

You can require users to authenticate themselves before getting access to your Web site. Authentication means that users verify their identity either by entering a username and password or by using a client certificate installed in their Web browser. The first method of requiring the username and password is the traditional method, which can be done with or without encryption. The second method of using client certificates is the SSL method, which must be done with encryption on. Refer to the Novell Documentation Web site (http://www.novell.com/documentation) for more information on encryption.

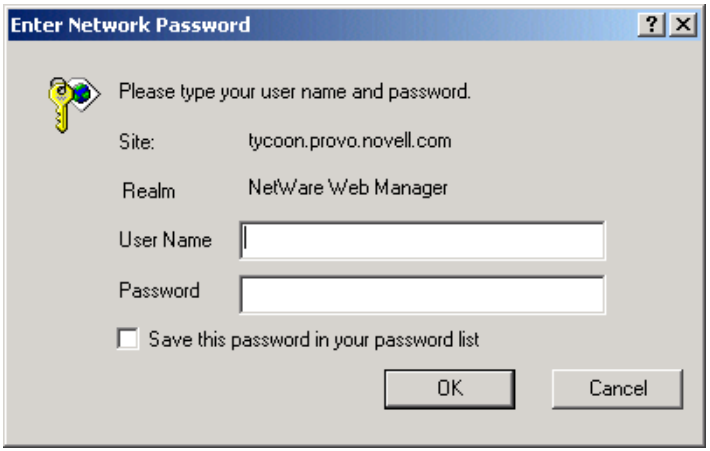
Username and Password Authentication

If you require users to enter a username and password to get access to your Web site, you store the list of users and groups in an LDAP database, which can be either a file stored on the Web server computer or an LDAP server on a remote computer, for example, Novell Directory Services (NDS) using LDAP or by using NDS directly.

When users attempt to access a file or directory that has User-Group authentication, the Web browser displays a dialog box asking the user to enter a username and password. The server can get this information encrypted or not, depending on whether encryption is turned on for your server.

After entering the username and password, users either see the requested file or directory listing, or a message denying them access. This following figure shows the authentication window.

Figure 9 Authentication Window



IMPORTANT: If your server doesn't use SSL encryption, the username and password that the end user types are sent unencrypted across the network. Someone could intercept the network packets and read the username and password being sent to the Web server. For this reason, User-Group authentication is most effective when combined with SSL encryption, or Host-IP authentication, or both.

Client Certificate Authentication

You can confirm users' identities with security certificates before giving the users access to your Web site. You can do this in the following two ways:

- The server can use the information in the certificate as proof of identity.
- The server can verify the certificate itself, provided the certificates are published in an LDAP directory.

When a request comes in and you have client authentication on, the server performs these actions in the following order:

- When the browser sends the certificate, the server checks if the certificate is from a trusted certificate authority (CA). If not, the server ends the transaction.
- If the certificate is from a trusted CA, the server maps the certificate to a user's entry using the CERTMAP.CONF file.
- If the certificate maps correctly, then the Web server follows the ACL rule, or command, specified for that user. The rule can deny or allow the request.

Host-IP Authentication

You can limit access to files and directories on your Web site by making them available only to people using specific computers. You specify hostnames or IP addresses for the computers that you want to allow or deny. You can use wildcard patterns to specify multiple computers or entire networks. If you want to use Host-IP authentication, you must have DNS running in your network and your computer must be configured to use it.

Users can access the files and directories immediately without entering a username or password. If the computer doesn't have access, the user will get a message denying access. You can also customize this message.

HINT: It is possible for more than one person to have access to a computer. For this reason, Host-IP authentication is most effective when combined with User-Group authentication. If both methods of authentication are used, the end user will have to enter a username and password before getting access.

Access Control Files

When you use access control on your Web server, the settings are stored in a file with the extension .ACL. Access control files are stored in the directory *server_root*/*server_type*ACL, where *server-type* is the name of the server.

The main ACL filename is GENERATED-HTTPS-*server-id*.ACL. The temporary working file is called GENWORK-HTTPS-*server-id*.ACL. If you use the Server Manager forms to restrict access, you'll have these two files. However, if you want more complex restrictions, you can create multiple files and reference them from the MAGNUS.CONF file. There are also a few features available only by editing the files. For example, you can restrict access to the server depending on the time of day or day of the week.

You also manually create and edit .ACL files if you want to customize access control. For example, you might want to use an Oracle* or Informix* database of users instead of an LDAP database. To do this type of customizing, you need to use the access control API to program a hook into the server's access control structure. This API is written in C. For more information on the API, see the Netscape DevEdge Online site (http://developer.netscape.com).

How Does Access Control Work?

You can control access to the entire server or to parts of the server (directories, files, file types). When the server evaluates an incoming request, it determines access based on a hierarchy of rules called access control entries (ACEs), and then it uses the matching entries to determine if the request is allowed or denied. Each ACE specifies whether or not the server should continue to the next ACE in the hierarchy. The collection of ACEs is called an access control list (ACL).

When a request comes in to the server, the server looks in OBJ.CONF for a reference to an ACL, which is then used to determine access. By default, the server has one ACL file that contains multiple ACLs.

For example, suppose someone requests the following URL:

http://www.novell.com/my_stuff/web/presentation.html

The server would first check access control for the entire server. If the ACL for the entire server was set to continue, the server checks to see if there is an ACL for the file type .HTML. Then it checks for an ACL for the directory MY_STUFF. If one exists, it checks the ACE and then moves on to the next directory. The server continues traversing the path either until it reaches an ACL that says not to continue or until it reaches the final ACL for the requested URL (in this case, the file PRESENTATION.HTML).

To set up access control for this example using the Server Manager forms, you could create an ACL for the file only or for each resource leading to the file, for example, one for the entire server, one for the MY_STUFF directory, one for the MY_STUFF/WEB directory, and one for the file.

The following sample ACL file illustrates one way to control access to this resource.

```
# File automatically written

## You may edit this file by hand#

version 3.0;

# This ACL allows everyone in the local database or LDAP
  directory
  acl "agents";
  authenticate (user,group) {
  prompt = "<Enterprise or News> Server";
  };
  deny (all)
```

```
    user = "anyone";
allow absolute (all)
    user = "all";# This ACL denies all access to the
my_stuff directory
acl "path=C:\Novonyx\SuiteSpot\docs\my_stuff";
deny (all)
    user = "anyone";# This ACL allows access to anyone in
the user database
acl "path=C:\Novonyx\SuiteSpot\docs\my_stuff\web";
allow (all)
    user = "anyone";# This ACL allows access to the file to
anyone in the "my_group" group
acl
"path=C:\Novonyx\SuiteSpot\docs\my_stuff\web\presentation
.html";
allow (all)
    user = "anyone";
    group = "my_group"# This is the default ACL and denies
access to anyone
acl "default";
deny (all)
```

Restricting Access

This section takes you through the process of restricting user access to documents on your Web site. The sections following this one describe in detail each option available when using access control. Keep in mind that most access control rules use only a subset of the available options.

There is also a section of examples on restricting different resources. You can review these examples in "Restricting Access to the Entire Server" on page 287.

To create an access-control rule:

1. From the Web Manager home page, click Enterprise Web Server *servername* > Server Preferences > Restrict Access.

   A form appears on which you select and edit an existing access control rule or specify a new rule by either choosing the resource you want to apply to the rule (the file, directory, or wildcard pattern you want to control) or typing a name to assign to the ACL. There are three sections to this main form:

   - Pick a Resource: Allows you to specify a wildcard pattern for files or directories to restrict access to (such as *.HTML) or to specify a directory or a filename to restrict.

- Pick an Existing ACL: Allows you to select an ACL that you've created.
- Type in the ACL Name: Allows you to create named ACLs. Use this option only if you're familiar with ACL files and the OBJ.CONF configuration file. You'll need to manually edit OBJ.CONF if you want to apply named ACLs to resources.

2. In the section you want to modify, from the Editing field select the part of your Web site (the resource) that you want to control.

   For example, you can select Entire Server to set up access control for your entire server.

   HINT: Refer to Table 18 on page 280 at the end of this procedure for an example list of resources that are typically given limited access control.

3. Click Edit Access Control.
4. Click New Line.
5. Click Deny to select the action you want to apply to the rule.

   The bottom frame displays a form where you can select whether you want to allow or deny access to the users, groups, or hosts you'll specify in the following steps. Select the option you want > click Update.

6. Click Anyone to specify User-Group authentication listed under the Users/Groups column.
7. Select the options you want > click Update.

   See "Specifying Users and Groups" on page 282 for more detailed information about each option.

8. Click Anyplace to specify the computers you want to include in the rule.
9. Select the options you want > click Update.

   See "Specifying Hostnames and IP Addresses" on page 284 for more information about each option.

10. Click All to specify the access rights you want to include in the rule.

    Check the access rights in the bottom frame > click Update.

11. Click X under the Extra column to enter a customized ACL entry if you are familiar with ACL files.

    This area is useful if you use the access control API to customize ACLs.

12. Click Update.

13 Check the appropriate box in the Continue column if you want the access control rule to continue in a chain.

This means the next line is evaluated before the server determines if the user is allowed access.

14 Check Access Control Is On.

See "When Access Control Is On" on page 285 for more information.

15 Check Response When Denied if you want the user to be redirected to another URL if their request is denied.

16 Select Respond with the Following URL > type the URL in the field.

17 Click Update.

See "Responding When Access Is Denied" on page 286 for more information.

18 Repeat steps 8 through 17 for each rule you need.

19 Click Submit to store the new access control rules in the ACL file.

If you click Revert, the server removes any changes you made to the rules from the time you first opened the two-frame window

WARNING: Be cautious when using Revert because you cannot restore your edits. In most cases, it is probably better to delete the rule lines individually.

20 Click Save and Apply.

Table 18 Example List of Resources That Are Typically Given Limited Access Control

Resource Wildcard	What It Means
default	A named ACL created during installation that restricts write access, so only users in the local database or LDAP directory can publish documents, for example, by using the Web Publisher.
Entire Server	One set of rules determines the access to your entire Web site, including any virtual servers you have running. To restrict access to a virtual server, specify the path of its document root.

Resource Wildcard	What It Means
*.html	Controls access to all files with an .HTML extension.
*.cgi	Controls access to all files with a .CGI extension.
usr/ns-home/cgi-bin/*	Controls access to all files and directories in the CGI-BIN directory. Note that the path is absolute. On NT, the path must include the drive letter.
agents	A named ACL that restricts access to all agents. The Web server contains this ACL by default.
uri="/sales"	Controls access to the SALES directory in the document root. To specify URIs, create a named ACL.

The following sections describe the options that appear in the bottom frame of the access control window.

Setting Access Control Actions

You can specify the action the server takes when a request matches the access control rule.

- Allow: The users or computers can access the requested resource.
- Deny: The users or computers cannot access the requested resource.

The server goes through the list of ACEs to determine the access. For example, the first ACE is usually to deny everyone. If the first ACE is set to continue, the server checks the second ACE in the list. (If Continue is not checked, everyone would be denied access to the resource.) If the second entry matches, then the next ACE is used. The server continues down the list until it reaches either an ACE that doesn't match or that matches, but is set to not continue. The last ACE that matches is used to determine if access is allowed or denied. For example, any user in the database can view a file (read access), but they must be in the Pubs group if they want to publish a file to the server.

Specifying Users and Groups

You can restrict access to your Web site based on the user who requests a resource. With user and group authentication, users are prompted to enter a username and password before they can access the resource specified in the access control rule.

The Web server uses a list of users, who might be sorted into groups, to determine access rights for the user requesting a resource. The list of users (and the groups they are included in) are stored either in a database on the Web server computer or in an LDAP server, such as the Netscape Directory Server. You should make sure the database has users and groups in it before you set access control.

You can allow or deny access to everyone in the database, or you can allow or deny specific people by using wildcard patterns or lists of users or groups.

To configure access control with users and groups, follow the general directions for restricting access. When you click the Users/Groups column, a form appears in the bottom frame. The following list describes the options in the form:

- Anyone (No Authentication): Anyone can access the resource without having to enter a username or password. However, the user might be denied access based on other settings, such as hostname or IP address.
- Authenticated People Only: All users requesting the resource will have to type a username and password before getting access.

  If the username they enter isn't in the database, the access control rule won't apply to them. However, if the rule says Deny and then a group is listed, that group is denied, but everyone else in the database could be allowed depending on if there is another ACL that matches their request.
- All in the Authentication Database: Matches any user who has an entry in the database. To use this option, you must also select Authenticated People Only.
- Only the Following People: Allows you to specify certain users and groups to match.
  - User: Matches the individual users you specify.
  - Group: Matches all users in the groups you specify.

You can list the users and groups of users individually by separating the entries with commas. Or you can enter a wildcard pattern. To use this option, you must also select Authenticated People Only.

- Prompt for Authentication: Allows you to specify message text that appears in the authentication window. You can use this text to describe what the user needs to enter. Depending on the operating system, the user will see about the first 40 characters of the prompt. Netscape Navigator and Netscape Communicator cache the username and password and associate them with the prompt text. This means that if the user accesses areas (files and directories) of the server that have the same prompt, the user won't have to retype usernames and passwords. Conversely, if you want to force users to reauthenticate for various areas, you simply need to change the prompt for the ACL on that resource.
- Authentication Methods: Specifies the method the server uses when getting authentication information from the client.
  - Default: Uses the default method you specify in the OBJ.CONF file, or Basic if there is no setting in OBJ.CONF. If you select Default in this form, the ACL rule doesn't specify a method in the ACL file. Default is the best choice because you can easily change the methods for all ACLs by editing one line in the OBJ.CONF file.
  - Basic: Uses the HTTP method to get authentication information from the client. The username and password are only encrypted if encryption is turned on for the server.
  - SSL: Uses the client certificate to authenticate the user. If you use this method, SSL must be turned on for the server. If you have encryption on, you can combine basic and SSL methods.
  - Other: Uses a custom method you create using the access control API.
- Authentication Database: Allows you to select a database that the server uses to authenticate users. The default setting means the server looks for users and groups in either the local database or an LDAP directory, depending on the setting specified in the Administration Server. However, you can configure individual ACLs to use different databases. You can specify different databases and LDAP directories in the file *server_root*/USERDB/DBSWITCH.CONF and then choose the database you want to use in the ACL by selecting it in the drop-down list. If you use the access control API to use a custom database (for example, to use an Oracle or Informix database), you can type the name of the database in the Other field in the Users & Groups form.

Specifying Hostnames and IP Addresses

You can restrict access to your Web site based on which computer the request comes from. You specify this restriction by using wildcard patterns that match the computers' hostnames or IP addresses.

To specify users from hostnames or IP addresses, follow the general directions for restricting access. Restricting by hostname is more flexible than by IP address; if a user's IP address changes, you won't have to update this list. Restricting by IP address, however, is more reliable; if a DNS lookup fails for a connected client, hostname restriction cannot be used.

The hostname and IP addresses should be specified with a wildcard pattern or a comma-separated list. The wildcard notations you can use are specialized; you can only use an asterisk (*). Also for the IP address, the asterisk must replace an entire byte in the address. For example, `198.95.251.*` is acceptable, but `198.95.251.3*` is not. When the asterisk appears within an IP address, it must be the right-most character. For example, `198.*` is acceptable, but `198.*.251.30` is not.

For hostnames, the asterisk must also replace an entire component of the name. For example, `*.novell.com` is acceptable, but `*sers.novell.com` is not. When the asterisk appears in a hostname, it must be the left-most character. For example, `*.novell.com` is acceptable, but `users.*.com` is not.

Setting Access Rights

You can set access rights to files and directories on your Web site. In addition to allowing or denying all access rights, you can specify a rule that allows or denies partial access rights. For example, you can give people read-only access rights to your files, so they can view the information, but not change the files. This is particularly useful when you use the Web publishing feature to publish documents.

When you create an access control rule, the default access rights are set to all access rights. To change access rights, click the appropriate link in the Rights column in the top frame, then check or uncheck the access rights you want to set for a particular rule. The following list describes each access right you can check.

- Read Access: Lets a user view a file. This access right includes the HTTP methods GET, HEAD, POST, and INDEX.

- Write Access: Lets a user change or delete a file. This access right includes the HTTP methods PUT, DELETE, MKDIR, RMDIR, and MOVE.
- Execute Access: Applies to server-side applications, such as CGI programs, Java applets, and agents.
- Delete Access: Lets users delete a file or directory.
- List Access: Lets a user get directory information. That is, they can get a list of the files in that directory. This applies to Web Publisher and to directories that don't contain an INDEX.HTML file.
- Info Access: Lets a user get headers (HTTP_HEAD method). This is mainly used by the Web Publisher.

Writing Customized Expressions

You can enter custom expressions for an ACL. You can use this feature if you are familiar with the syntax and structure of ACL files. There are a few features available only by editing the ACL file or creating custom expressions. For example, you can restrict access to your server depending on the time of day, day of the week, or both.

The following customized expression shows how you could restrict access by time of day and day of the week. This example assumes you have two groups in your LDAP directory: the Regular group gets access Monday through Friday, 8:00 a.m. to 5:00 p.m. The Critical group gets access all the time.

```
allow (read)
  {
  (group=regular and dayofweek=”mon,tue,wed,thu,fri”);
  (group=regular and (timeofday>=0800 and timeofday<=1700));
  (group=critical)
  }
```

For more information on valid syntax and ACL files, see the Help.

When Access Control Is On

You can turn off access control for any part of the server that a user accesses. For example, you could create an ACL that restricts access to the resource .HTML. You could then have an ACL for the entire server that is turned off. In this case, the only time access-control is used is when a user requests any file or directory in the *.HTML extension.

When you uncheck the option, you'll get a prompt asking if you want to erase records in the ACL. When you click OK, the server deletes the ACL entry for that resource from the ACL file.

If you want to deactivate an ACL, you can comment out the ACL lines in the file GENERATED-HTTPS-SERVER-ID.ACL by putting pound signs (#) at the beginning of each line.

Responding When Access Is Denied

You can choose the response a user sees when denied access. You can vary the message for each access control object. By default, the user is sent a message that says the file wasn't found. The HTTP error code "404 Not Found" is also sent.

To change what message is sent for a particular ACL:

1. In the ACL form, click Response When Denied.
2. In the lower frame, select Respond with the Following URL.
3. In the text field, type a URL or URI to a text or HTML file in your server's document root that you want to send to users when they are denied access.

   Make sure the file doesn't contain references to other files, such as style sheets or images, because they won't be sent.
4. Click Update.

   IMPORTANT: Make sure any users who get the response file have access to that file. If you have access control on the response file and the user is denied access to both the original resource and the response file, the server will send the default denied response.
5. Click Submit in the top frame.

Examples of Restricting Access

This section describes some common examples for restricting access to a Web server and its contents. Some of these examples assume you set up the default ACL to deny anyone access to the server. You can also add a **`deny all`** line as the first rule to each of these examples, as done in the example for the entire server.

Restricting Access to the Entire Server

This example allows access to users in a group called Employees, who access the server from computers in a sub-domain. There are no access control rules for other resources on the server. You might use this example if you have a server for a department and you only want users to access the server from computers in a specific subdomain of your network.

1. From the Web Manager home page, click Enterprise Web Server *servername* > Server Preferences > Restrict Access.
2. In the section called Pick a Resource, select the entire server from the Editing drop-down list.

   The resource must be selected.
3. Click Edit Access Control.
4. Click New Line.
5. Click New Line again to create a second rule.
6. Click Deny in the second rule.
7. In the bottom form that appears, select Allow > click Update.
8. Click Anyone in the second rule.
9. In the bottom form, type the group that you want to have access to the server.

   For this example, type **`Employees`** in the Group field.

   HINT: Note that the two options, Authenticated People Only and Only the Following People, are checked automatically.
10. Click Update.
11. Click Anyplace in the second rule.
12. In the bottom form, type a wildcard pattern for the hostnames of the computers you want to allow.

    For example, type *.emp.mozilla.com in the Host Names field.
13. Click Update.
14. Uncheck the Continue box in the second rule of the top frame > click Submit.
15. Click Save and Apply.

Be sure to restart the server for the changes to take effect. The following text is the ACL file for this example:

```
# File automatically written## You may edit this file by
  hand#version 3.0;acl "default";deny (all)     user =
  "anyone";allow absolute (all)    user = "employees" and
  dns = "*.emp.mozilla.com";
```

Restricting Access to a Directory (Path)

This example lets users in a group called Executives have read access to a directory and its subdirectories and files on the server. The user called CEO has full permissions to the directory.

You might use this example if you have a directory on your server that one person owns (he or she publishes to this directory) and you want one group of users to read the files. For example, you might have a project owner who publishes status information for the project team to review.

1. From the Web Manager home page, click Enterprise Web Server *servername* > Server Preferences > Restrict Access.
2. In the section called Pick a Resource, click Browse.
3. In the form that appears, click the link for the directory you want to restrict.

   HINT: If you want to view all files in your server root, click Options on the Choose a Part of Your Server form > check the List Files As Well As Directories checkbox > click OK.
4. Click Edit Access Control.
5. Click New Line twice to create two rules.

   IMPORTANT: Don't edit the default values for the first rule. These values deny all access to the directory. You'll edit the second rule to allow read access to the Executives group.
6. Click Deny in the second rule.
7. In the bottom form that appears, select Allow > click Update.
8. Click Anyone in the second rule.
9. In the bottom form, type the group you want to have access to the server.

   For this example, type **`Executives`** in the Group field.
10. Click Update.

11 Click All in the top frame.

12 Uncheck the Write and Delete access rights.

13 Click Update.

14 Click New Line to create a rule for the CEO user.

15 Select Allow.

16 Click Anyone in the third rule.

17 In the bottom form, type **CEO** in the User field > click Update.

18 Uncheck Continue for both the second and the third rules.

This means that the server ignores any ACLs for directories or files under the directory you specified above.

19 Click Submit > Save and Apply.

The entry in the GENERATED.HTTPS-*serverid*.ACL file for this example looks like this:

```
acl "path=d:/novonyx/suitespot/docs/senior-staff/";
  deny (all)
      user = "anyone";
  allow absolute (read,execute,list,info)
      group = "executives";
  allow absolute (all)
      user = "ceo";
```

Restricting Access to a URI (Path)

This example uses a URI to control access to a single user's content on the Web server. URIs are paths and files relative to the server's document root directory. Using URIs is an easy way to manage your server's content if you frequently rename or move all or part of it, for example, for disk space. It's also a good way to handle access control if you have additional document roots.

This example gives anyone read access to files and directories in the path specified by the URI /MY_DIRECTORY. Only one user (yourself in this example) has full access to the directories and files.

You might use this example if you have several users who publish their content on your server. The users want to have write access to their content, and they want anyone to have read/execute access.

1. From the Web Manager home page, click Enterprise Web Server *servername* > Server Preferences > Restrict Access.
2. In the section called Type In the ACL Name, type the URI you want to control.

   For example, type **URI=/my_directory**. Click Edit Access Control. The two-frame forms appear.
3. Click New Line to create the first rule that allows all users read access.
4. Click Deny.
5. In the bottom form that appears, select Allow > click Update.
6. Click All.
7. Uncheck the Write and Delete access rights.

   This means users can't add or remove files, but they can view them and run any applications in the directories.
8. Click Update.
9. Click New Line to create a rule for the owner of the directory > select Allow for the second rule.
10. Click Anyone.
11. In the bottom form, type **ME** in the User field > click Update.
12. Uncheck Continue for both the first and second rules.

    This means that the server ignores any ACLs for other URIs, directories, or files under the URI you specified above.
13. Click Submit > Save and Apply.

The entry in the GENERATED.HTTPS-*serverid*.ACL file for this example looks like this:

```
acl "uri=/my_directory";
  allow absolute (read,execute,list,info)
      user = "anyone";
  allow absolute (all)
      user = "me";
```

Restricting Access to a File Type

This example controls write and delete access to all files with the extension .CGI. You might use this example if you only want specific users to create programs that run on your server. In this example, anyone can run the programs, but only users in the Programmers group can create or delete them.

1. From the Web Manager home page, click Enterprise Web Server *servername* > Server Preferences > Restrict Access.
2. In the section called Pick a Resource, click Wildcard.
3. In the prompt that appears, type **`*.CGI`** > click OK.

   This wildcard pattern matches any request that contains a file or directory with the .CGI extension.
4. Click Edit Access Control.
5. Click New Line to create the first rule that will allow all users read access.
6. Click Deny.
7. In the bottom form that appears, select Allow > click Update.
8. Click All > uncheck the Write and Delete access rights.

   This means users can't add or remove files or directories with the .CGI extension.
9. Click Update.
10. Click New Line to create a rule that allows write and delete access to the Programmers group.
11. Select Allow for the second rule.
12. Click Anyone.
13. In the bottom form, type **`Programmers`** in the Group field.
14. Click Update > Submit > Save and Apply.

In this example, both Continue boxes are checked. This means that if a file is requested, the server will first look at the ACL for the file type, and then it will continue to look for another ACL that matches, for example, an ACL on the URI or the path. The server checks ACLs in the following order:

1. Pathcheck Functions in OBJ.CONF: For example, these could be wildcard patterns for files or directories. The entry in the ACL file would appear as follows: **`acl "*.cgi";`**

2. URIs: For example, a path relative to the document root. The entry in the ACL file would appear as follows: **`acl ""uri=/my_directory";`**
3. Pathnames: For example, an absolute path to a file or directory. The entry in the ACL file would appear as follows: **`acl "path=d:\novonyx\suitespot\docroot1\sales/";`**

The entry in the GENERATED.HTTPS-*serverid.*ACL file for this example looks like this:

```
acl "*.cgi";
  allow (read,execute,list,info)
      user = "anyone";
  allow (all)
      group = "programmers";
```

Restricting Access Based on Time of Day

This example restricts write and delete access to the server during working hours. You might use this example if you don't want people publishing documents at times when people might be accessing the files. This example allows users to publish during the evening hours of the week (between 6:00 p.m. and 6:00a.m., Monday through Friday) and all times during the weekend.

1. From the Web Manager home page, click Enterprise Web Server *servername* > Server Preferences > Restrict Access.
2. In the section called Pick a Resource, select the entire server from the Editing drop-down list > click Edit Access Control.
3. Click New Line.
4. Click Deny.
5. In the bottom form that appears, select Allow > click Update.
6. Click All > uncheck the Write and Delete access rights.

   This means that if a user wants to add, update, or delete a file or directory, this rule won't apply and the server will search for another rule that matches.
7. Click Update.
8. Click New Line to create a rule that restricts the write and delete methods.
9. Select Allow for the second rule.
10. Click the X link to create a customized expression.

11 In the bottom field, edit the existing lines to include the following:

```
user = "anyone" anddayofweek = "sat,sun" or(timeofday >=
  1800 andtimeofday <= 600)
```

12 You might want to select the entire text element and copy it to memory; if there are errors, you'll have to re-enter the text.

13 Click Update.

The top form will display unrecognized expressions in the Users/Groups and From Host columns, because you created a custom expression.

14 Click Submit.

If you made any errors in the custom expression, you'll get a JavaScript alert. Correct any changes > click Submit again.

15 Click Save and Apply.

16 Restart your server for the changes to take effect.

F Port Number Assignments

Port numbers enable IP packets to be sent to a particular process on a computer that is connected to the Internet. Some port numbers are permanently assigned; for example, e-mail data under SMTP goes to port number 25. A process such as a Telnet session receives a temporary port number when it starts. The data for that Telnet session goes to newly assigned port number, and the port number goes out of use when the telnet session ends.

A total of 65,535 port numbers are available for use.

Some port numbers in NetWare® 6 can be reasssigned from one net service to another. Others cannot be reassigned. When adding or removing products or services from your NetWare 6 installation, or when making new port number assignments, refer to the following table, which indicates default port assignments and notes which ports can be reassigned and which cannot.

The symbols used in the table indicate the following:

✓ = The port is configurable

✘ = The port is not configurable

♦ = Dependent on a subsystem

? = Availability and dependency cannot be determined

Table 19 Port Assignments and Availability Status, Listed by Product

Product or Service	Assigned Ports and Availability Status
Apache	✓ 80
	✓ 443

Product or Service	Assigned Ports and Availability Status
Apple* Filing Protocol (AFP)	✘ 548
BorderManager™	✘ 21 ✘ 119 ✓ 443 ✘ 1040 ✘ 1045 ✘ 1959 ✘ 7070 ✘ 8080 ✘ 9090
Common Internet File System (CIFS)	? 139
CsAudit	✓ 2000
DirXML™ NDS-to-NDS®	✓ 8090
DirXML Remote Loader	✓ 8000
Domain Name Service (DNS)	✘ 53
eGuide	♦ 389 ♦ 636
File Transfer Protocol (FTP)	✘ 20 ✘ 21
GroupWise® Monitor	✓ 1099

Product or Service	Assigned Ports and Availability Status
GroupWise Internet Agent (GWIA)	? 25
	? 110
	? 143
	✘ 389
	✘ 636
	✓ 9850
GroupWise Web Access	♦ 80
	♦ 443
	✓ 7205
iFolder™	♦ 80
	♦ 389
	♦ 443
	♦ 636
iMonitor	✓ 80
iPrint	♦ 443
	? 631
Lightweight Directory Access Protocol (LDAP)	✓ 389
	✓ 636
Line Printer Requester (LPR)	? 515
Media Server	✘ 554
Message Transfer Agent (MTA)	✓ 3800
	✓ 7100
	✓ 7180

Product or Service	Assigned Ports and Availability Status
NetWare Core Protocol™ (NCP™)	✘ 524
NetWare Enterprise Web Server	✓ 80
	✓ 443
NetWare File System	✘ 20
	✘ 111
	✓ 2049
NetWare Graphical User Interface	✓ 9000
	✓ 9001
NetWare/IP (NWIP)	✘ 396
NetWare Remote Manager (NRM)	? 80
	? 81
	✓ 8008
	✓ 8009
NetWare Web Access	♦ 80
Network Time Protocol (NTP)	✘ 123
NLSLRUP.NLM	✓ 21571
	✓ 21572

Product or Service	Assigned Ports and Availability Status
Novell Internet Messaging System (NIMS™)	✓ 80 ✓ 81 ? 110 ? 143 ? 389 ✓ 443 ✓ 444 ? 636
Novell Modular Authentication Services (NMAS™)	? 1242
Portal Services	♦ 80 ♦ 443 ♦ 8080
Post Office Agent (POA)	✓ 1677 ✓ 2800 ✓ 7101 ✓ 7181
Radius	✓ 1812
Remote Console™ DOS	✓ 2034
Remote Console Java	✓ 2034 ✓ 2036 ✓ 2037
Server Compatibility Mode Driver (SCMD)	✘ 2302

Product or Service	Assigned Ports and Availability Status
Service Locator Protocol (SLP)	✘ 427
Simple Network Management Protocol (SNMP)	? 161
Telnet	✘ 23
Tomcat	✓ 8080
Virtual Private Network (VPN)	✘ 213 ✘ 353 ✓ 2010
Web Manager	✓ 2200
Zenworks™ for Desktops 3	✓ 2544 ✘ 2638 ✘ 8039
Zenworks for Servers 2	♦ 80 ♦ 443 ✘ 1229 ♦ 2037 ✓ 2544 ♦ 8008 ♦ 8009