XII WebAccess

57 Scaling WebAccess

If your GroupWise® system is relatively small (one domain and a few post offices) and all post offices reside in the same location, a basic installation of GroupWise WebAccess might very well meet your needs. However, if your GroupWise system is large, spans multiple locations, or requires failover support, you might need to scale your GroupWise WebAccess installation to better meet the reliability, performance, and availability needs of your users.

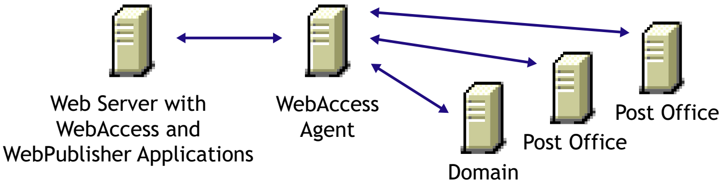
The following sections provide information about the various configurations you can implement and instructions to help you create the configuration you choose:

- "WebAccess Configurations" on page 805
- "Installing Additional WebAccess Components" on page 807
- "Configuring Redirection and Failover Support" on page 810

For information about creating a basic GroupWise WebAccess installation, see "Installing GroupWise WebAccess" in the *GroupWise 6.5 Installation Guide*.

WebAccess Configurations

A basic installation of GroupWise WebAccess requires the WebAccess Agent and the WebAccess Application, as shown in the following diagram. The WebPublisher Application is also required if you plan to use GroupWise WebPublisher.

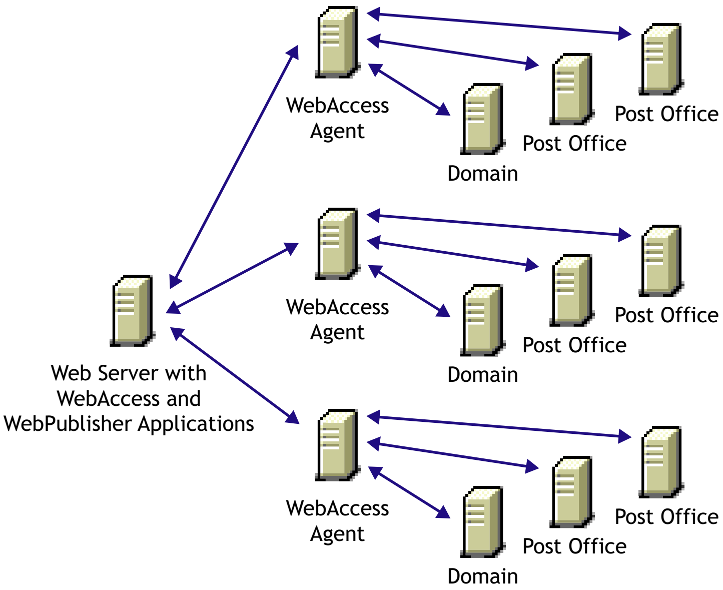


Depending on your needs, it might be necessary for you to add additional WebAccess Agents or to have multiple Web servers running the WebAccess Application and WebPublisher Application.

- "Multiple WebAccess Agents" on page 805
- "Multiple WebAccess and WebPublisher Applications" on page 806

Multiple WebAccess Agents

GroupWise WebAccess is designed to allow one installation of the WebAccess Application and WebPublisher Application to support multiple WebAccess Agents, as shown in the following diagram.

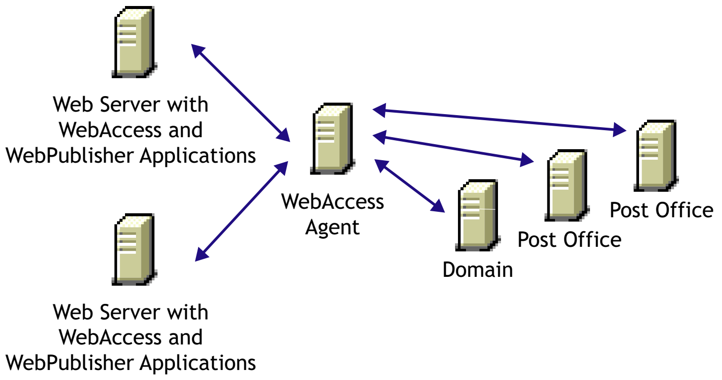


There are various reasons why you might want to add additional WebAccess Agents, including:

- **Improving reliability:** One WebAccess Agent might provide sufficient access and performance, but you want to protect against downtime that would occur if the WebAccess Agent became unavailable due to server failure or some other reason. Installing more than one WebAccess Agent enables you to set up failover support to make your system more reliable.
- **Improving performance:** The WebAccess Agent is designed to be close to the GroupWise databases. It requires direct access to a domain database and either direct access to post office databases or TCP/IP access to the Post Office Agents. For best performance, you should ensure that the WebAccess Agent is on the same local area network as the domain and post offices it needs access to. For example, in most cases you would not want a WebAccess Agent in Los Angeles accessing a post office in London.
- **Improving availability:** The WebAccess Agent has 12 threads assigned to process user requests, which means that it can process only 12 requests at one time regardless of the number of users logged in. If necessary, you can increase the number of threads allocated to the WebAccess Agent, but each thread requires additional server memory. If you reach a point where WebAccess is unavailable to users because thread utilization is at a peak and all server memory is being used, you might need to have several WebAccess Agents, installed on different network servers, servicing your post offices. For information about changing the number of allocated threads, see "Configuring the WebAccess Agent" on page 829.

Multiple WebAccess and WebPublisher Applications

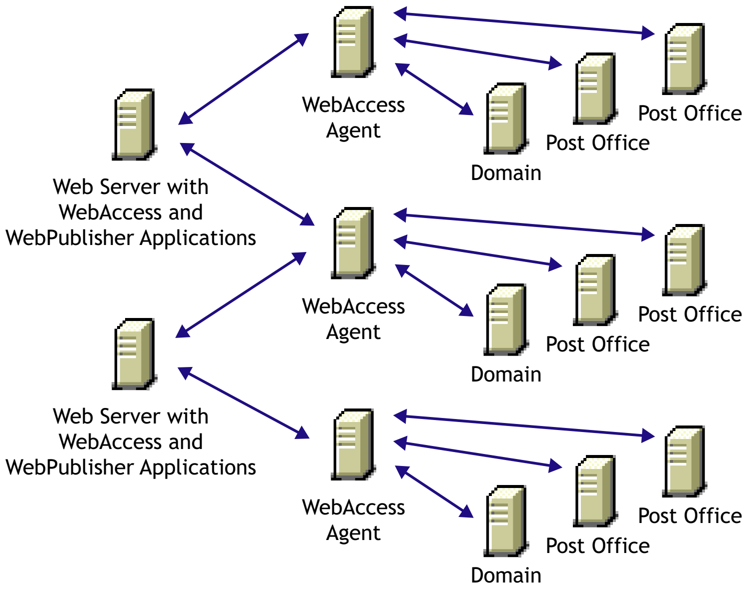
As with the WebAccess Agent, you can also install the WebAccess Application and WebPublisher Application to multiple Web servers, as shown in the following diagram.



Some reasons for wanting to use this type of configuration include:

- Enabling WebAccess users on an intranet to access GroupWise through an internal Web server and WebAccess users on the Internet to access GroupWise through an exposed Web server.
- Increasing Web server performance by balancing the workload among several Web servers, especially if you are using the Web server for other purposes in addition to GroupWise WebAccess.
- Hosting WebAccess (the WebAccess Application) on one Web server for your GroupWise users and WebPublisher (the WebPublisher Application) on another Web server for public Internet use.

If necessary, you can use multiple WebAccess Agents in this configuration, as shown below.



Installing Additional WebAccess Components

The following sections assume that you have installed at least one WebAccess Agent and one WebAccess Application (or WebPublisher Application) and now need to install additional agents or applications.

- "Installing Additional Components on NetWare or Windows" on page 808
- "Installing Additional Components on Linux" on page 809

Installing Additional Components on NetWare or Windows

- "Installing a NetWare or Windows WebAccess Agent" on page 808
- "Installing a NetWare or Windows WebAccess or WebPublisher Application" on page 808

For more information, see "Setting Up GroupWise WebAccess on NetWare or Windows" in the *GroupWise 6.5 Installation Guide*.

Installing a NetWare or Windows WebAccess Agent

1 Insert the *GroupWise 6.5 Administrator* CD into the CD drive to start the Installation program, click Install Products, click GroupWise WebAccess, then click Install GroupWise WebAccess. If the Installation program does not start automatically, run setup.exe from the root of the CD.

   or

   If you've already copied the GroupWise WebAccess software to a software distribution directory, run setup.exe from the internet\webacces directory.

2 Click Yes to accept the license agreement and display the Select Components dialog box.

3 Deselect all components except the GroupWise WebAccess Agent, then click Next.

4 Follow the prompts to create the WebAccess Agent's gateway directory, install the WebAccess Agent software, and create the WebAccess Agent's object in Novell® eDirectory™.

   If you are installing to a domain where another WebAccess Agent already exists, you must use a different directory and object name than the one used for the existing WebAccess Agent.

5 When installation is complete, you will need to configure your system so that the WebAccess and WebPublisher Applications know about the WebAccess Agent and can direct the appropriate user requests to it. For information, see "Configuring Redirection and Failover Support" on page 810.

Installing a NetWare or Windows WebAccess or WebPublisher Application

To install a WebAccess Application or a WebPublisher Application to a web server:

1 Insert the *GroupWise Administrator* CD into the CD drive to start the installation program, click Install Products, click Groupwise WebAccess, then click Install GroupWise WebAccess. If the installation program does not start automatically, run setup.exe from the root of the CD.

   or

   If you've already copied the Groupwise WebAccess software to a software distribution directory, run setup.exe from the internet/webacces directory.

2 Click Yes to accept the license agreement and display the Select Components dialog box.

3 Deselect all components except the GroupWise WebAccess application and/or the Groupwise WebPublisher Application, then click Next.

   The WebAccess Application and WebPublisher Application must be associated with a WebAccess Agent. For information on configuring a WebAccess or WebPublisher Application to connect to other WebAccess Agents, see "Configuring Redirection and Failover Support" on page 810.

4 Enter the path for the WebAccess Agent's gateway directory.

5 Follow the prompts to install the files to the web server. Restart the Web server.

Installing Additional Components on Linux

- "Installing a Linux WebAccess Agent" on page 809
- "Installing a Linux WebAccess and WebPublisher Application" on page 809

For more information, see "Setting Up GroupWise WebAccess on Linux" in the *GroupWise 6.5 Installation Guide*.

Installing a Linux WebAccess Agent

1. Make sure that LDAP is running on your eDirectory server and that it is configured to accept login from the WebAccess Agent Installation program.
2. Open a new terminal window, then enter the following command:

   ```
   xhost + localhost
   ```

3. In the same window, become root by entering **`su`** and the root password.
4. Change to the root of the *GroupWise 6.5 for Linux Administrator* CD.
5. Enter **`./install`**.
6. Select the language in which you want to run the Installation Advisor and install the WebAccess software, then click Next.
7. In the Installation Advisor, click Install Products > GroupWise WebAccess > Install WebAccess Agent.
8. When the installation is complete, click OK.
9. Click Configure WebAccess Agent.
10. Follow the prompts to configure the Linux WebAccess Agent.
11. When installation and configuration is complete, you need to configure your GroupWise system so that the WebAccess and WebPublisher Applications know about this instance of the WebAccess Agent and can direct the appropriate user requests to it. For instructions, see "Configuring Redirection and Failover Support" on page 810.

Installing a Linux WebAccess and WebPublisher Application

To install a WebAccess Application and a WebPublisher Application to a Web server:

1. After installing and configuring the WebAccess Agent, click Install GroupWise WebAccess Application with Apache and Tomcat if you want to create a new installation of Apache and Tomcat for this instance of the WebAccess Application.

   or

   If you want to use an existing Apache and Tomcat installations, click Install GroupWise WebAccess Application.

2. When the installation is complete, click OK.
3. Click Configure WebAccess Application.
4. Follow the prompts to configure the Linux WebAccess Application.
5. When the installation and configuration is complete, start or restart the Web server.

Configuring Redirection and Failover Support

Redirection enables the WebAccess Application to direct user requests to specific WebAccess Agents. For example, you might want WebAccess Agent 1 to process all requests from users on Post Office 1 and WebAccess Agent 2 to process all requests from users on Post Office 2.

Failover support enables the WebAccess Application to contact a second WebAccess Agent if the first WebAccess Agent is unavailable. For example, if the WebAccess Application receives a user request that should be processed by WebAccess Agent 1 but it is unavailable, the WebAccess Application can route the user request to WebAccess Agent 2 instead.

The following sections provide information to help you successfully configure redirection and failover support:

- "How the WebAccess Application Knows Which WebAccess Agents to Use" on page 810
- "Synchronizing the Encryption Key" on page 812
- "Specifying a WebAccess Agent in the WebAccess URL" on page 813
- "Assigning a Default WebAccess Agent to a Post Office" on page 815
- "Assigning a Default WebAccess Agent to a Domain" on page 816
- "Adding WebAccess Agents to the GroupWise Service Provider's List" on page 817

How the WebAccess Application Knows Which WebAccess Agents to Use

To redirect user requests or to fail over to a second WebAccess Agent, the WebAccess Application needs to know which WebAccess Agents you want it to use. This might be all of the WebAccess Agents in your system, or only specific WebAccess Agents.

Each time a user logs in, the WebAccess Application compiles a list, referred to as a redirection/failover list, of the WebAccess Agents defined in the locations listed below.

- **The WebAccess URL.** The standard URL does not contain a WebAccess Agent, but you can modify the URL to point to a specific agent.
- **The user's Post Office object.** You can assign a default WebAccess Agent to the post office to handle requests from the post office's users.
- **The user's Domain object.** You can assign a default WebAccess Agent to the domain to handle requests from the domain's users.
- **The GroupWiseProvider object.** This is the service provider used by the WebAccess Application to connect to WebAccess Agents.
- **The commgr.cfg file.** This file located in the WebAccess Application's home directory (novell\webaccess on the Web server or /opt/novell/groupwise/webaccess on Linux).

By default, only the GroupWise Provider object and the commgr.cfg file include a WebAccess Agent definition, as shown in the following table:

Location	WebAccess Agent
WebAccess URL	No agent defined
Post office	No agent defined
Domain	No agent defined

Location	WebAccess Agent
GroupWise service provider	Agent 1
Commgr.cfg	Agent 1

If no other WebAccess Agents are defined (as is the case by default), the WebAccess Application will direct all user requests to the WebAccess Agent (Agent 1) listed in the commgr.cfg file. This file is located in the WebAccess Application's home directory on the Web server. The commgr.cfg file contains the IP address and encryption key for the WebAccess Agent that was associated with the WebAccess Application during the application's installation.

If Agent 1 is not available, the user will receive an error message and will be unable to log in.

Redirection/Failover List: Example 1

Assume that the WebAccess Agents are defined as follows:

Location	WebAccess Agent
WebAccess URL	No agent defined
Post office	Agent 1
Domain	Agent 4
GroupWise service provider	Agent 2 Agent 3
Commgr.cfg	Agent 4

Using this information, the WebAccess Application would create the following redirection/failover list:

List Entry	Taken From
Agent 1	Post office
Agent 4	Domain
Agent 2	GroupWise service provider
Agent 3	GroupWise service provider

Because there is no WebAccess Agent defined in the WebAccess URL, the WebAccess Application will redirect the user's request to the default WebAccess Agent (Agent 1) assigned to the user's post office. If Agent 1 is unavailable, the WebAccess Application will fail over to the domain's default WebAccess Agent (Agent 4). If Agent 4 is unavailable, the WebAccess Application will fail over to Agent 2 and then Agent 3, both of which are defined in the GroupWise service provider's list.

Redirection/Failover List: Example 2

Assume that the WebAccess Agents are defined as follows:

Location	WebAccess Agent
WebAccess URL	No agent defined
Post office	No agent defined
Domain	No agent defined
GroupWise service provider	Agent 1 Agent 2 Agent 3
Commgr.cfg	Agent 2

Using this information, the WebAccess Application would create the following redirection/failover list:

List Entry	Taken From
Agent 1	GroupWise service provider
Agent 2	GroupWise service provider
Agent 3	GroupWise service provider

Because there is no WebAccess Agent defined in the WebAccess URL, user's post office, or user's domain, the WebAccess Application will redirect the user's request to the first WebAccess Agent (Agent 1) in the GroupWise service provider's list. If Agent 1 is unavailable, the WebAccess Application will fail over to Agent 2 and then Agent 3.

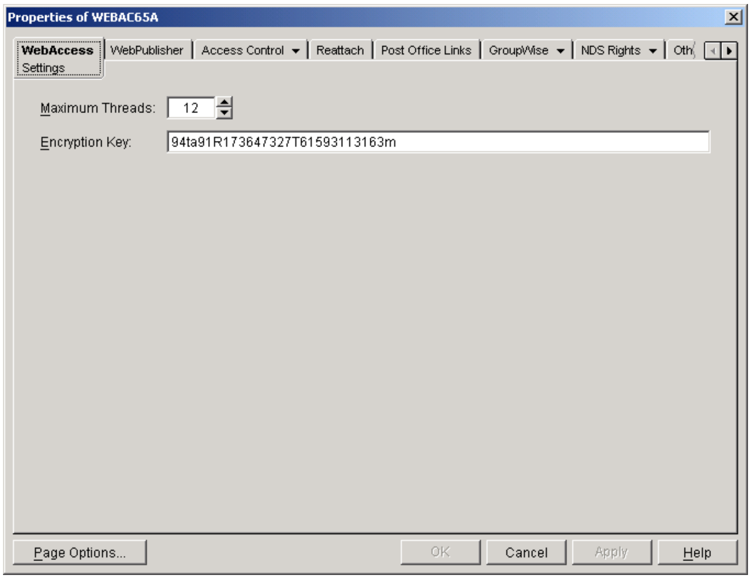
Synchronizing the Encryption Key

Every WebAccess Agent has an encryption key. In order to communicate with a WebAccess Agent, the WebAccess Application must know the agent's encryption key. The encryption key is randomly generated when the WebAccess Agent object is created in eDirectory, which means that every WebAccess Agent has a unique encryption key.

If a WebAccess Application will communicate with more than one WebAccess Agent, all the WebAccess Agents must use the same encryption key.

To modify a WebAccess Agents encryption key:

1. In ConsoleOne®, right-click the WebAccess Agent object, then click Properties.
2. If necessary, click the WebAccess tab to display the WebAccess Settings page.



Maximum Threads: 12

Encryption Key: 94ta91R173647327T61593113163m

3 Make the encryption key the same as the key for any other WebAccess Agents with which the WebAccess Application communicates.

4 Click OK to save the changes.

Specifying a WebAccess Agent in the WebAccess URL

To have the WebAccess Application connect to a WebAccess Agent other than the one specified in the commgr.cfg file, you can add the WebAccess Agent's IP address and port number to the URL that calls the WebAccess Application. For example, the default WebAccess Application URL is:

NetWare and Windows: http://*web_server_ip_address*/servlet/webacc
Linux: http://*web_server_ip_address*/gw/webacc

This URL causes the WebAccess Application to use the IP address and port number that is listed in the commgr.cfg file. To redirect the WebAccess Application to another WebAccess Agent, you would use the following URLs:

NetWare and Windows: http://*web_server_ip_address*/servlet/webacc
?GWAP.ip=*agent_ip_address*&GWAP.port=*port_number*
Linux: http://*web_server_ip_address*/gw/webacc
?GWAP.ip=*agent_ip_address*&GWAP.port=*port_number*

For example:

NetWare and Windows: http://151.155.123.45/servlet/webacc
?GWAP.ip=151.155.789.10&GWAP.port=7204
Linux: http://151.155.123.45/gw/webacc
?GWAP.ip=151.155.789.10&GWAP.port=7204

In this example, the WebAccess Application will redirect its requests to the WebAccess Agent at IP address 151.155.789.10 and port number 7204. If the WebAccess Agent is using the same port number that is listed in the commgr.cfg file, you do not need to include the GWAP.port parameter. Or, if the WebAccess Agent is using the same IP address that is listed in the commgr.cfg file, you do not need to include the GWAP.ip parameter.

If you want, you can use the WebAccess Agent's DNS hostname in the URL rather than its IP address.

You can also specify the user interface language by adding the &User.lang option. This allows you to bypass the initial WebAccess language page. For example:

```
NetWare and Windows: http://151.155.123.45/servlet/webpub
                  ?GWAP.ip=151.155.789.10&GWAP.port=7204&User.lang=en
Linux: http://151.155.123.45/gw/webpub
                ?GWAP.ip=151.155.789.10&GWAP.port=7204&User.lang=en
```

You can use the language codes listed below with the &User.lang parameter in the WebAccess URL.

Language	Code	Language	Code
Arabic	ar	Hebrew	iw
Brazilian Portuguese	pt	Hungarian	hu
Chinese Simplified	cs	Italian	it
Chinese Traditional	ct	Japanese	jp
Czechoslovakian	cz	Korean	kr
Danish	da	Norwegian	no
Dutch	nl	Polish	pl
English	us	Russian	ru
Finnish	su	Spanish	es
French	fr	Swedish	sv
German	de		

You can add the URL to any Web page. For example, if you are using the Web Services page as your initial WebAccess page, you could add the URL to that page. You will need to add one URL for each WebAccess Agent.

For example, suppose you had offices in three different locations and installed a WebAccess Agent at each location to service the post offices at those locations. To enable the WebAccess Application to redirect requests to the WebAccess Agent at the appropriate location, you could modify the Web Services page to display a list of the locations. The modified page would include the following HTML code (if WebAccess is running on NetWare or Windows):

```
<UL>

<LI><A HREF="http://151.155.123.45/servlet/
webacc?GWAP.ip=151.155.789.10&GWAP.port=7204>San Francisco
</A></LI>

<LI><A HREF="http://151.155.123.45/servlet/
webacc?GWAP.ip=151.155.456.12>New York
</A></LI>

<LI><A HREF="http://151.155.123.45/servlet/
```

```
webacc?GWAP.ip=151.155.654.33&GWAP.port=7203>London
</A></LI>

</UL>
```

In the preceding example, in Linux, the directory “servlet” is replaced by “gw”.

The displayed HTML page would contain the following list of locations:

- San Francisco
- New York
- London

When a user selectes a location, the WebAccess Application will route all requests to the WebAccess Agent at the selected location.

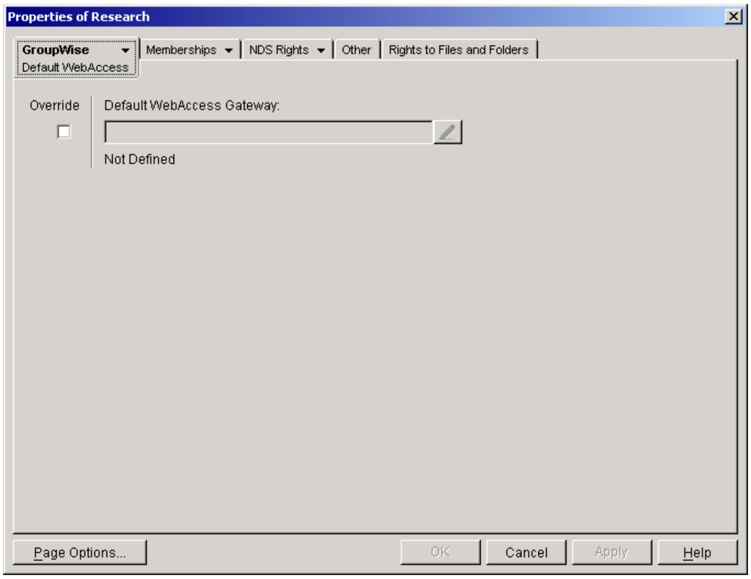
Assigning a Default WebAccess Agent to a Post Office

The WebAccess Application uses the post office’s default WebAccess Agent if no WebAccess Agent has been specified in the WebAccess URL (see “Specifying a WebAccess Agent in the WebAccess URL” on page 813) or if that WebAccess Agent is unavailable. This applies only if you have multiple WebAccess Agents installed in your GroupWise system. If you have only one WebAccess Agent, it services all post offices.

To assign a default WebAccess Agent to a post office:

1 In ConsoleOne, right-click the Post Office object, then click Properties.

2 Click GroupWise > Default WebAccess to display the Default WebAccess page.



Override ☐

Default WebAccess Gateway:

Not Defined

3 Select the Override box to turn on the option.

4 In the Default WebAccess Gateway box, browse for and select the WebAccess Agent that you want to assign as the default agent.

When you have multiple WebAccess Agents and a user logs in to GroupWise WebAccess, the GroupWise Application running on the Web server checks to see if a default WebAccess Agent has been assigned to the user’s post office. If so, the WebAccess Application connects to the assigned WebAccess Agent. If not, it connects to the default WebAccess Agent assigned to the post office's domain, as described in "Assigning a Default WebAccess Agent to a Domain" on page 816 or to one of the WebAccess Agents in its service provider list, as described in "Adding WebAccess Agents to the GroupWise Service Provider's List" on page 817. If possible, select a WebAccess Agent that has good access to the post office to ensure the best performance.

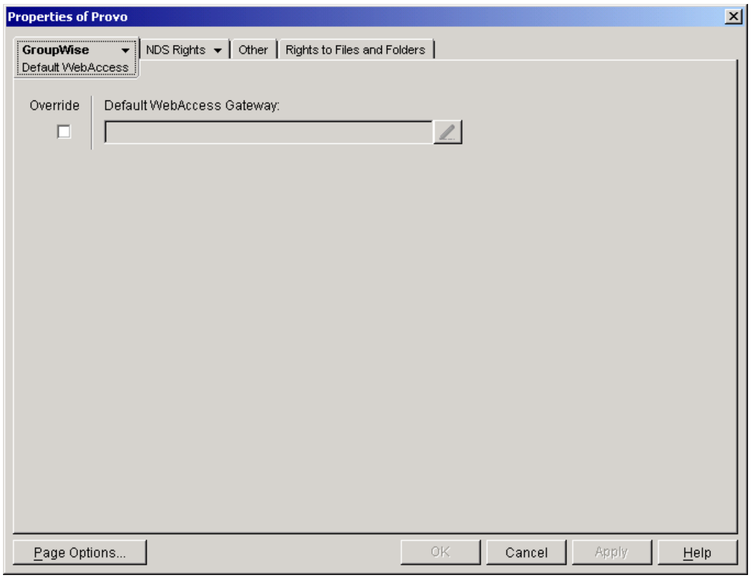
5. Click OK to save the changes.

Assigning a Default WebAccess Agent to a Domain

The WebAccess Application uses the domain's default WebAccess Agent if 1) no WebAccess Agent has been specified in the WebAccess URL (see "Specifying a WebAccess Agent in the WebAccess URL" on page 813), 2) no default WebAccess Agent has been defined for the user's post office, or 3) neither of those WebAccess Agents are available. This applies only if you have multiple WebAccess Agents installed in your GroupWise system. If you have only one WebAccess Agent, it services users in all domains.

To assign a default WebAccess Agent to a domain:

1. In ConsoleOne, right-click the Domain object, then click Properties.
2. Click GroupWise > Default WebAccess to display the Default WebAccess page.



3. Select the Override box to turn on the option.
4. In the Default WebAccess Gateway box, browse for and select the WebAccess Agent that you want to assign as the default agent.

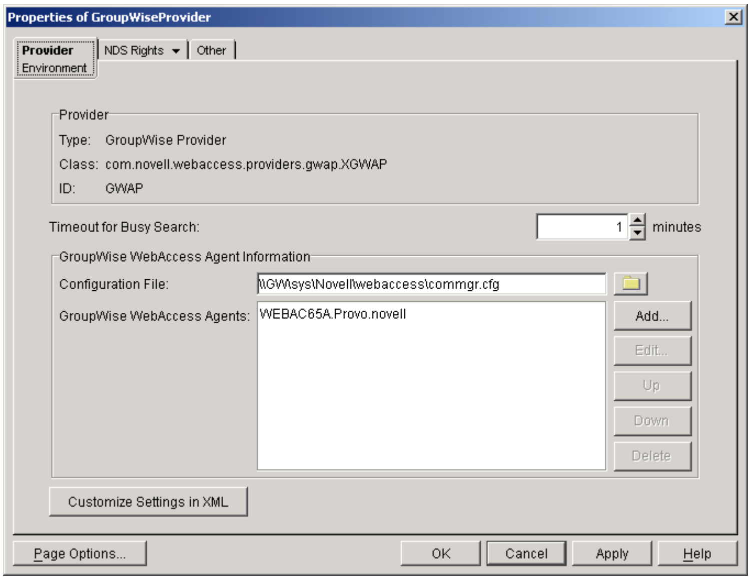
   When you have multiple WebAccess Agents and a user logs in to GroupWise WebAccess, the GroupWise Application running on the Web server checks to see if a default WebAccess Agent has been assigned to the user's post office, as described in "Assigning a Default WebAccess Agent to a Post Office" on page 815. If so, the WebAccess Application connects to the assigned WebAccess Agent. If not, it connects to the default WebAccess Agent assigned to the post office's domain or to one of the WebAccess Agents in its service provider list, as described in "Adding WebAccess Agents to the GroupWise Service Provider's List" on page 817. If possible, you should select a WebAccess Agent that has good access to the domain's post offices to ensure the best performance. Each post office uses the domain's default WebAccess Agent unless you override the default at the post office level.

5 Click OK to save the changes.

Adding WebAccess Agents to the GroupWise Service Provider's List

1 In ConsoleOne, right-click the GroupWise service provider object (GroupWiseProvider), then click Properties.

2 If necessary, click the Provider tab to display the Environment page.



Provider

Type: GroupWise Provider

Class: com.novell.webaccess.providers.gwap.XGWAP

ID: GWAP

Timeout for Busy Search: 1 minutes

GroupWise WebAccess Agent Information

Configuration File: \\GW\sys\Novell\webaccess\commgr.cfg

GroupWise WebAccess Agents: WEBAC65A.Provo.novell

The GroupWise WebAccess Agents list displays the WebAccess Agents the GroupWise service provider can communicate with when attempting to complete a request. By default, the list includes the WebAccess Agent that is defined in the commgr.cfg file (listed in the Configuration File field). If the first WebAccess Agent is unavailable, the GroupWise service provider will attempt to use the second, third, fourth, and so on until it is successful.

3 Click Add, select the WebAccess Agent you want to add to the list, then click OK.

4 Repeat Step 3 for each WebAccess Agent you want to add to the list, then click OK to save the changes.

58 Controlling User Access

To control users' access to their mailboxes through GroupWise® WebAccess, you can do the following:

- Prevent users from logging in to their mailboxes through GroupWise WebAccess. By default, all GroupWise users can use WebAccess. See "Controlling User Access to Mailboxes" on page 819.
- Determine how long WebAccess users can remain inactive (no requests) before they are automatically logged out. The default is 20 minutes. See "Setting the Timeout Interval for Inactive Sessions" on page 825.
- Determine which WebAccess features (spell checking, LDAP directory searches, password modification, an so forth) are available to users. When WebAccess runs on NetWare or Windows, all features are available by default. When WebAccess runs on Linux, all features except opening attachments and LDAP directory searches are available by default. See "Configuring User Access to WebAccess Features" on page 826.

Controlling User Access to Mailboxes

You control which users have access to their mailboxes by creating classes of service and assigning users membership in a class. For example, if you don't want users on a particular post office to have access to their mailboxes through WebAccess, you can create a class of service that prevents access and then assign the entire post office membership in that class.

The following sections provide information to help you create and manage classes of service:

- "Class Membership" on page 819
- "Creating a Class of Service" on page 820
- "Adding Users to a Class of Service" on page 822
- "Maintaining the Access Database" on page 823

Class Membership

When you create a class of service, you assign membership in the class at a domain level, post office level, distribution list (group) level, or individual user level, which means that a user could be assigned membership in multiple classes. For example, a user might be a member in one class because his or her domain is a member; at the same time, the user is a member in another class because his or her post office is a member of that class. Because each user can have only one class of service, membership conflicts are resolved hierarchically, as shown below:

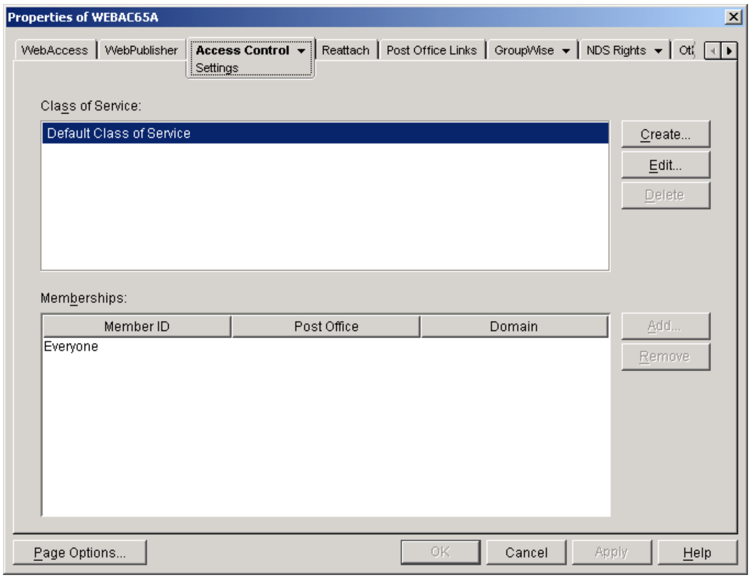
Membership assigned to a user through a...	Overrides membership assigned to the user through the...
domain	◆ default class of service
post office	◆ default class of service ◆ domain
distribution list	◆ default class of service ◆ domain ◆ post office
user	◆ default class of service ◆ domain ◆ post office

If a user's membership in two classes of service is based upon the same level of membership (for example, both through individual user membership), the class that applies is the one that allows the most privileges. For example, if the user belongs to one class of service that allows access to WebAccess and another class that prevents access, the class that allows access applies to the user.

Creating a Class of Service

1 In ConsoleOne®, right-click the WebAccess Agent object, then click Properties.

2 Click Access Control > Settings to display the Access Control Settings page.



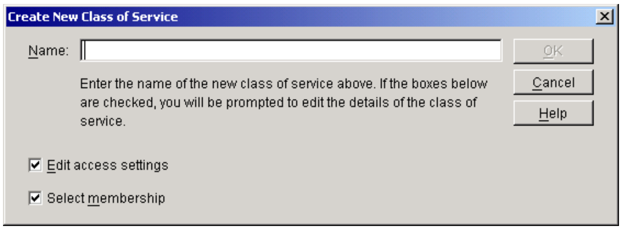
Class of Service:

Default Class of Service

Memberships:

Member ID	Post Office	Domain
Everyone		

3 Click Create to display the Create New Class of Service dialog box.



Create New Class of Service

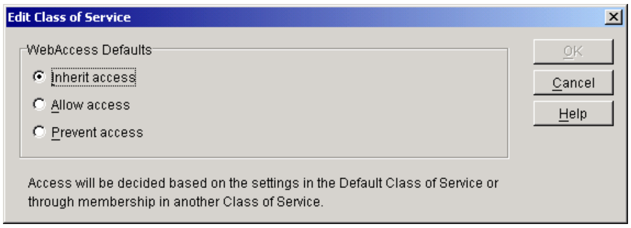
Name:

Enter the name of the new class of service above. If the boxes below are checked, you will be prompted to edit the details of the class of service.

☑ Edit access settings

☑ Select membership

4 Type a name for the class, then click OK to display the Edit Class of Service dialog box.



Edit Class of Service

WebAccess Defaults

● Inherit access

○ Allow access

○ Prevent access

Access will be decided based on the settings in the Default Class of Service or through membership in another Class of Service.

5 Select one of the following options:

Inherit Access: Select this option if you want members of this class of service to inherit their access from the default class of service or another class of service that they have membership in.

Allow Access: Select this option to enable members of the class to use WebAccess.

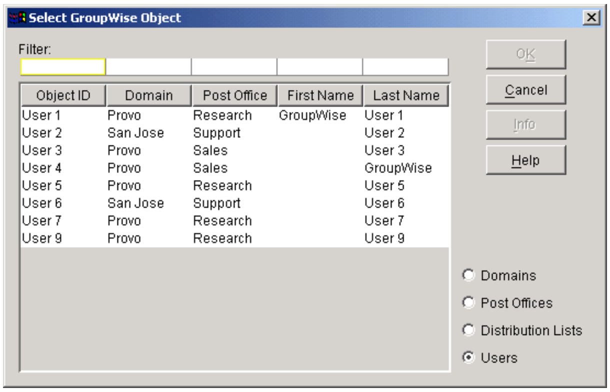
If you select Allow Access, you must also set a timeout interval. The timeout interval determines how long the WebAccess Agent keeps open a dedicated connection to the post office on behalf of the user. If the agent does not receive a user request within the specified interval, it closes the user's connection to the post office in order to free up its resources and the Post Office Agent's resources for other uses.

When the WebAccess Agent closes a user's connection to the post office, the user is not logged out of WebAccess. The user can continue to use WebAccess. As soon as the agent receives a request from the user, it opens the user's connection again. In general, you will want to leave the timeout interval set to the default 20 minutes.

You can also have users automatically logged out of WebAccess after a specified period of activity. WebAccess logout is handled by the WebAccess Application running on the Web server, not by the WebAccess Agent. For information, see "Setting the Timeout Interval for Inactive Sessions" on page 825.

Prevent Access: Select this option to prevent members of the class from using WebAccess.

6 Click OK to display the Select GroupWise Object dialog box.

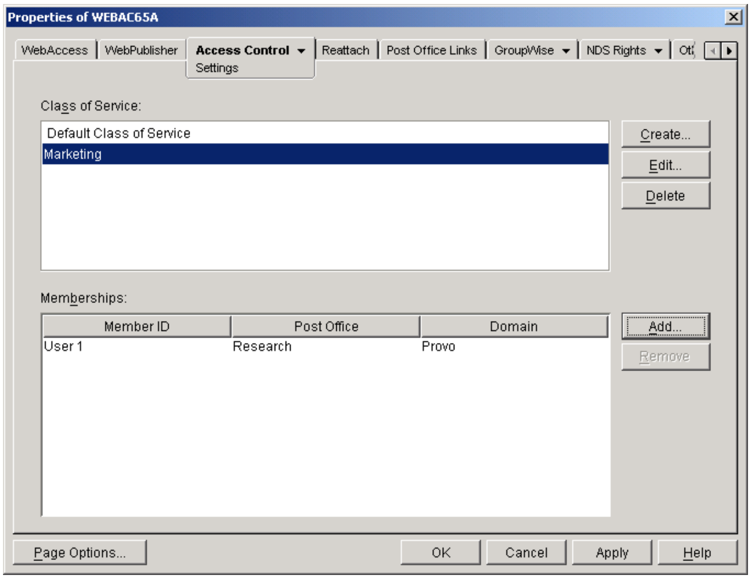


Object ID	Domain	Post Office	First Name	Last Name
User 1	Provo	Research	GroupWise	User 1
User 2	San Jose	Support		User 2
User 3	Provo	Sales		User 3
User 4	Provo	Sales		GroupWise
User 5	Provo	Research		User 5
User 6	San Jose	Support		User 6
User 7	Provo	Research		User 7
User 9	Provo	Research		User 9

○ Domains
○ Post Offices
○ Distribution Lists
● Users

7 Click Domains, Post Offices, Distribution Lists, or Users to display the list you want.

8 In the list, select the domain, post office, distribution list, or user you want, then click Add to add the object as a member in the class. You can Control-click or Shift-click to select multiple users.



Class of Service:

Default Class of Service
Marketing

Memberships:

Member ID	Post Office	Domain
User 1	Research	Provo

9 To add additional domains, post offices, distribution lists or users as members of the class of service, select the class of server, then click Add to display the Select GroupWise Object dialog box.

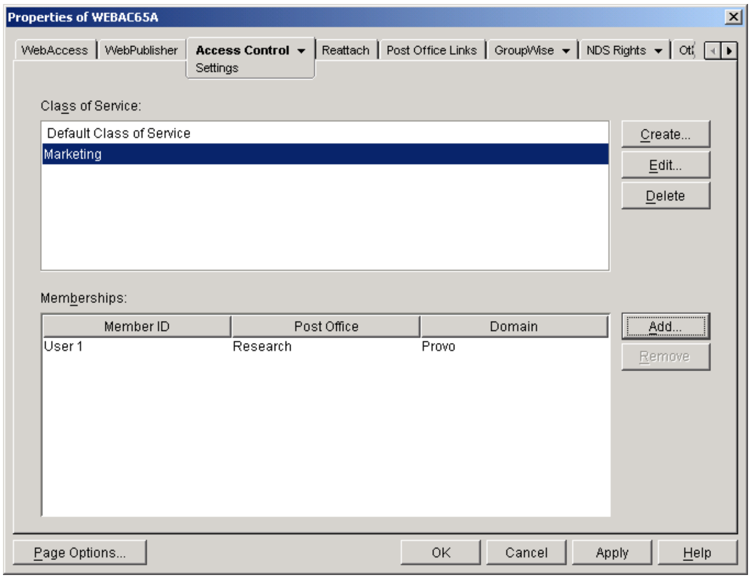
10 Click OK (on the Settings page) when finished adding members.

Adding Users to a Class of Service

The following steps help you add users to an existing class of service. For information about adding new classes of service, see "Creating a Class of Service" on page 820.

1 In ConsoleOne, right-click the WebAccess Agent object, then click Properties.

2 Click Access Control > Settings to display the Access Control Settings page.



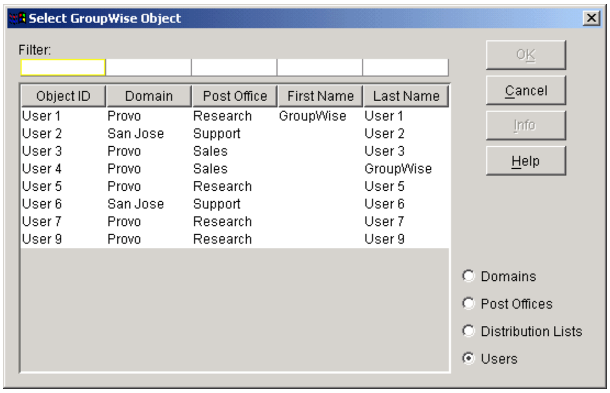
Class of Service:

Default Class of Service
Marketing

Memberships:

Member ID	Post Office	Domain
User 1	Research	Provo

3 In the Class of Service list, select the class you want to add members to, then click Add to display the Select GroupWise Object dialog box.



Object ID	Domain	Post Office	First Name	Last Name
User 1	Provo	Research	GroupWise	User 1
User 2	San Jose	Support		User 2
User 3	Provo	Sales		User 3
User 4	Provo	Sales		GroupWise
User 5	Provo	Research		User 5
User 6	San Jose	Support		User 6
User 7	Provo	Research		User 7
User 9	Provo	Research		User 9

○ Domains
○ Post Offices
○ Distribution Lists
● Users

4 Click Domains, Post Offices, Distribution Lists, or Users to display the list you want.

5 In the list, select the domain, post office, distribution list, or user you want, then click Add to add the object as a member in the class.

6 Repeat Step 3 through Step 5 for each object you want to add.

Maintaining the Access Database

The Access database stores the information for the classes of service you have set up to control user access to GroupWise WebAccess. When problems occur, you can validate the database to check for physical inconsistencies with the database's records and indexes. If inconsistencies are found, you can recover the database.

The Access database, gwac.db, is located in the *domain*\wpgate*webac65a* directory.

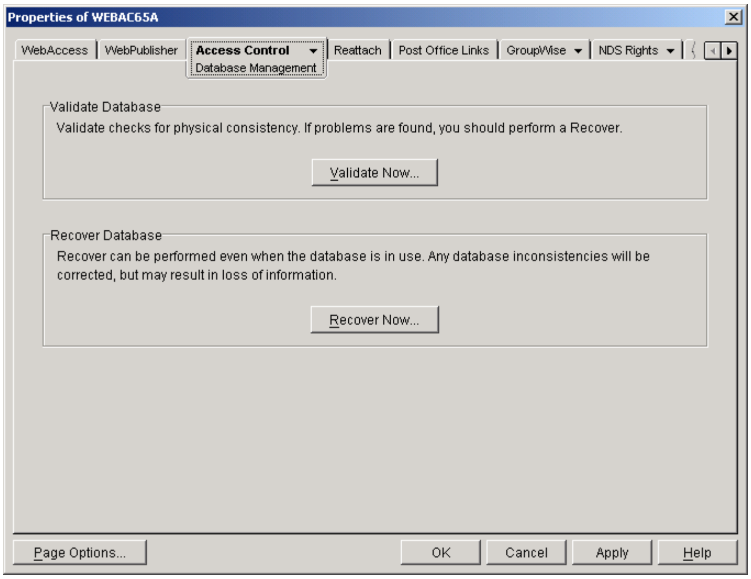
This section includes the following information:

- "Validating the Access Database" on page 824
- "Recovering the Access Database" on page 824

Validating the Access Database

Validating the Access database checks for physical inconsistencies with the database's records and indexes.

1. In ConsoleOne, right-click the WebAccess Agent object, then click Properties.
2. Click Access Control > Database Management to display the Database Management page.



Validate checks for physical consistency. If problems are found, you should perform a Recover.

Recover can be performed even when the database is in use. Any database inconsistencies will be corrected, but may result in loss of information.

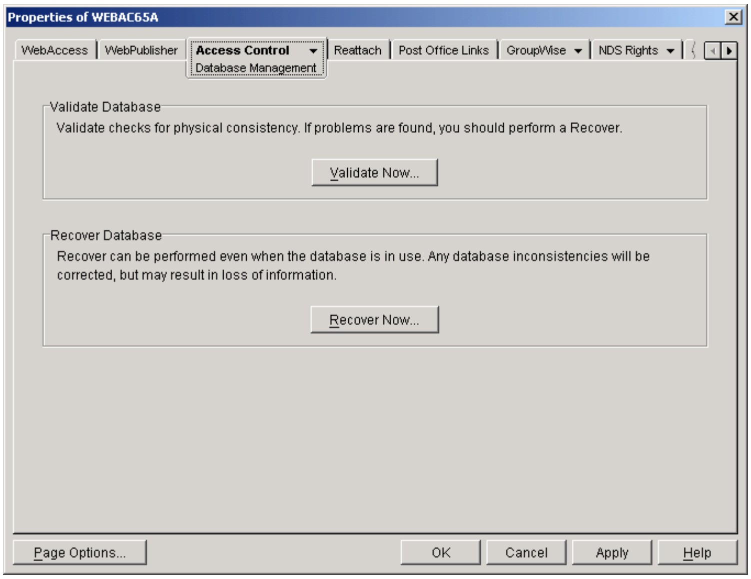
3. Click Validate Now.
4. After the database has been validated, click OK.

If inconsistencies were found, see "Recovering the Access Database" on page 824.

Recovering the Access Database

When you recover the Access database, a new database is created and all salvageable records are copied to the new database. Because some records might not be salvageable, after the recovery you will want to check the classes of services you have defined to see if any information was lost.

1. In ConsoleOne, right-click the WebAccess Agent object, then click Properties.
2. Click Access Control > Database Management to display the Database Management page.



Validate Database

Validate checks for physical consistency. If problems are found, you should perform a Recover.

Recover Database

Recover can be performed even when the database is in use. Any database inconsistencies will be corrected, but may result in loss of information.

3 Click Recover Now.

4 After the database has been recovered, click OK.

Setting the Timeout Interval for Inactive Sessions

By default, users will be logged out of GroupWise WebAccess after 20 minutes if they have not performed any actions that generate requests. Actions such as opening or sending a message generate requests. Other actions, such as scrolling through the Item List, composing a mail message without sending it, and reading Help topics, do not generate requests.

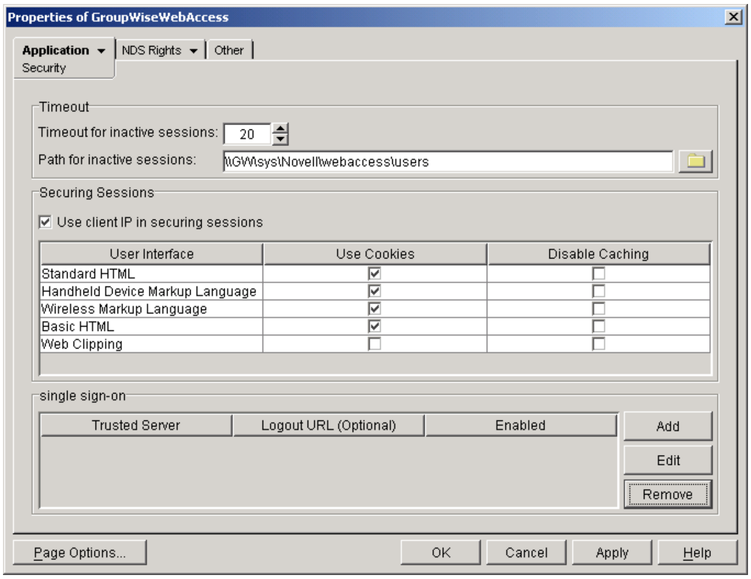
The timeout interval provides security for WebAccess users who forget to log out. It also helps the performance of the Web server by freeing the resources dedicated to that user's connection.

The WebAccess Application on the Web server controls the timeout. At the time the user is logged out, the WebAccess Application saves the user's current session to a directory on the Web server, where it is stored for 24 hours. If the logged-out user attempts to continue the session, he or she will be prompted to log in again, after which the WebAccess Application will renew the session. For example, suppose a user is composing a message when the timeout interval expires and then attempts to send the message. The user will be prompted to log in again, after which the message will be sent. No information is lost.

IMPORTANT: This timeout interval is different than the one you can establish when creating a class of service (see "Creating a Class of Service" on page 820). That timeout interval determines how long the WebAccess Agent will keep open a session with an inactive user, and this timeout interval determines how long the WebAccess Application will maintain an inactive session. In general, if the WebAccess Agent session times out, users will not notice; the next time they make a request, the WebAccess Agent will open a new session. However, if the WebAccess Application session times out, users will be prompted to log in again.

To modify the timeout interval:

1 In ConsoleOne, right-click the WebAccess Application object, click Properties, then click Application > Security to display the Security page.



Timeout

Timeout for inactive sessions: 20

Path for inactive sessions: \\GW\sys\Novell\webaccess\users

Securing Sessions

☑ Use client IP in securing sessions

User Interface	Use Cookies	Disable Caching
Standard HTML	☑	□
Handheld Device Markup Language	☑	□
Wireless Markup Language	☑	□
Basic HTML	☑	□
Web Clipping	□	□

single sign-on

Trusted Server	Logout URL (Optional)	Enabled

2 In the Timeout for Inactive Sessions box, select the number of minutes for the timeout interval.

3 In the Path for Inactive Sessions box, select the path for the directory where you want inactive sessions stored.

4 Click OK.

The timeout interval applies to all users who log in through the Web server where the WebAccess Application is running. You cannot set individual user timeout intervals. However, if you have multiple Web servers, you can set different timeout intervals for the Web servers by completing the above steps for each server's WebAccess Application.

Configuring User Access to WebAccess Features

By default, WebAccess users can:

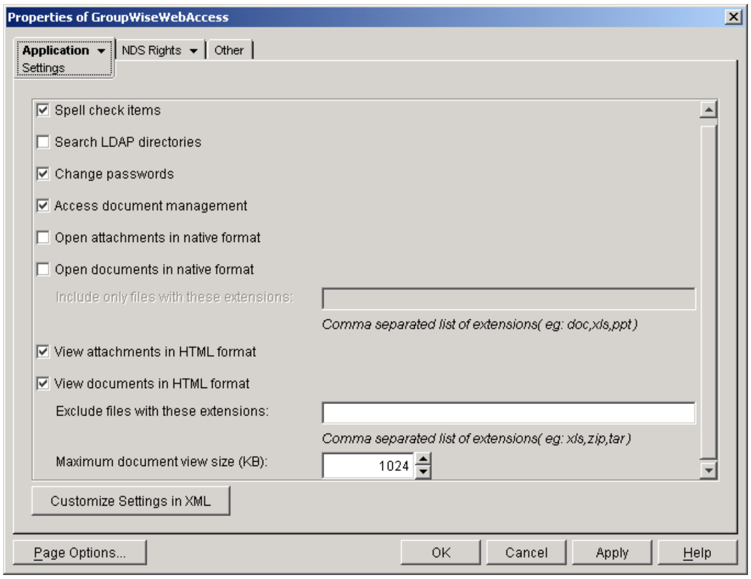
- Spell check messages
- Search LDAP directories (must be manually enabled on Linux)
- Change their GroupWise mailbox passwords
- Use Document Management Services
- Open attachments in native format (must be manually enabled on Linux)
- Open documents in native format (must be manually enabled on Linux)
- View attachments in HTML format
- View documents in HTML format

Access to these features is controlled by the WebAccess Application on the Web server. All users who log in through the Web server will have the same feature access. You cannot configure individual user settings. However, if you have multiple Web servers, you can establish different settings for the Web servers by completing the following steps for each server's WebAccess Application.

To configure the WebAccess feature settings:

1 In ConsoleOne, right-click the WebAccess Application object, then click Properties.

2 Click Application > Settings to display the Application Settings page.



☑ Spell check items

☐ Search LDAP directories

☑ Change passwords

☑ Access document management

☐ Open attachments in native format

☐ Open documents in native format

Include only files with these extensions:

Comma separated list of extensions(eg: doc,xls,ppt)

☑ View attachments in HTML format

☑ View documents in HTML format

Exclude files with these extensions:

Comma separated list of extensions(eg: xls,zip,tar)

Maximum document view size (KB): 1024

3 Configure the following settings:

Spell Check Items: Enable this option if you want users to be able to use the Novell® Speller to spell check an item’s text before sending the item. Disable this option to remove all Spell Check features from the user interface.

Search LDAP Directories: Enable this option if you have an LDAP server and you want users to be able to search any LDAP address books you have defined. Disable this option to remove all LDAP features from the user interface.

Change Passwords: Enable this option if you want users to be able to change their Mailbox passwords. Disable this option to remove all Password features from the user interface.

Access Document Management: Enable this option if you want users to be able to use the Document Management features. Disable this option to remove all Document Management features from the user interface.

Open Attachments in Native Format: By default, the Save As option enables users to save message attachments to their local drives and then open them in their native applications. You can turn on this option to enable the Open option. The Open option enables users to open message attachments directly in their native applications without first saving the files to the local drive.

This option requires that 1) each user’s Web browser knows the correct application or plug-in to associate with the attachment, according to its file extension or MIME type, and 2) the application or plug-in is available to the user. Otherwise, the user will be prompted to save the file to disk or specify the application to open it.

This option and the View Attachments in HTML Format option can both be enabled at the same time. Doing so gives users both the Open option and the View option, which means they have the choice of opening an attachment in its native application or viewing it as HTML.

Open Documents in Native Format: By default, the Save As option enables user to save library documents to their local drives and then open them in their native applications. You can turn on this option to enable the Open option. The Open option enables users to open documents directly in their native applications without first saving the files to the local drive.

This option requires that 1) each user's Web browser knows the correct application or plug-in to associate with the document, according to its file extension or MIME type, and 2) the application or plug-in is available to the user. Otherwise, the user will be prompted to save the file to disk or specify the application to open it.

This option and the View Documents in Native Format option can both be enabled at the same time. Doing so gives users both the Open option and the View option, which means they have the choice of opening a document in its native application or viewing it as HTML.

If you want only certain file types to be have the Open option, enter the file types in the Include Only Files With These Extensions field. Include only the extension and separate each extension with a comma (for example, doc, xls, ppt). The Open option will not be available for any file types not entered in this field.

View Attachments in HTML Format: Enable this option if you want users to be able to view any type of attachments in HTML format. Disable this option to require users to save an attachment to a local drive and view it in its native application. WebAccess uses Stellent* Outside In* HTML Export to convert files to HTML format.

For a list of the supported file format conversions, download the following document from the Stellent Web site:

OutSide In Supported Platforms and File Formats (http://www.stellent.com/stellent3/groups/mkt/documents/nativepage/outside_in_supported_platforms.pdf)

This option and the Open Attachments in Native Format option can both be enabled at the same time. Doing so gives users both the View option and the Open option, which means they have the choice of viewing an attachment as HTML or opening it in its native application.

View Documents in HTML Format: Enable this option if you want users to be able to view library documents in HTML format. Disable this option to require users to save a document to a local drive and view it in its native application. WebAccess uses Stellent Outside In HTML Export to convert files to HTML format.

For a list of the supported file format conversions, download the following document from the Stellent Web site:

OutSide In Supported Platforms and File Formats (http://www.stellent.com/stellent3/groups/mkt/documents/nativepage/outside_in_supported_platforms.pdf)

This option and the Open Documents in Native Format option can both be enabled at the same time. Doing so gives users both the View option and the Open option, which means they have the choice of viewing a document as HTML or opening it in its native application.

If you want to exclude certain file types from having the View option, enter the file types in the Exclude Files With These Extensions field. Include only the extension and separate each extension with a comma (for example, doc, xls, ppt). The View option will be available for any file types not entered in this field.

4 Click OK.

59 Configuring WebAccess Components

WebAccess consists of a number of components that can be configured to meet the specific needs of your GroupWise system:

- "Configuring the WebAccess Agent" on page 829
- "Configuring the WebAccess Application" on page 841
- "Configuring the Novell Speller Application" on page 854
- "Configuring the WebPublisher Application" on page 857
- "Configuring the GroupWise Service Provider" on page 867
- "Configuring the LDAP Service Provider" on page 868
- "Configuring the GroupWise Document Service Provider" on page 869

Configuring the WebAccess Agent

During installation, the GroupWise® WebAccess Agent is set up with a default configuration. On Linux, this happens during the configuration step. However, you can use the information in the following sections to optimize the WebAccess Agent for your environment:

- "Modifying WebAccess Settings" on page 829
- "Modifying WebPublisher Settings" on page 830
- "Controlling WebAccess Agent Logging" on page 832
- "Managing Access to Post Offices" on page 837
- "Changing the WebAccess Agent's Network Address or Port Numbers" on page 840

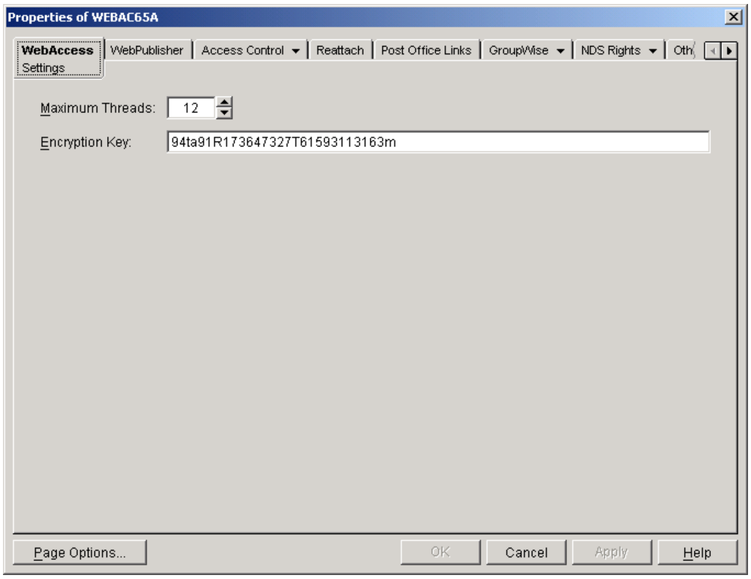
Modifying WebAccess Settings

Using ConsoleOne®, you can configure the following GroupWise WebAccess settings for the WebAccess Agent:

- The maximum number of threads the agent will use to process WebAccess messages
- The key used to encrypt information sent between the agent and the WebAccess Application

To modify the configuration information:

1. In ConsoleOne, right-click the WebAccess Agent object, then click Properties.
2. If necessary, click WebAccess > Settings to display the WebAccess Settings page.



Maximum Threads: 12

Encryption Key: 94ta91R173647327T61593113163m

3 Modify any of the following fields:

Maximum Threads: This is the maximum number of threads the agent will use at one time to process requests. The default (12) enables the agent to process 12 requests at one time, which is usually sufficient. If the agent regularly receives more requests than it has threads, you might want to increase the maximum number of threads. Increasing the threads increases the amount of server memory used by the agent.

To determine the maximum number of threads that have been in use at one time (for example, 8 of the 12 threads), you can view the server console screen for the NetWare® WebAccess Agent or view the status information displayed through the Web console. See "Monitoring the WebAccess Agent" on page 875.

Encryption Key: The encryption key is used to encrypt and decrypt the information sent between the WebAccess Agent and the WebAccess Application. If you do not want to use the default encryption key, you can type your own key. The encryption key must be identical to the encryption keys of any other WebAccess Agents that the WebAccess Application communicates with. For more information, see "Configuring Redirection and Failover Support" on page 810.

4 Click OK to save the changes.

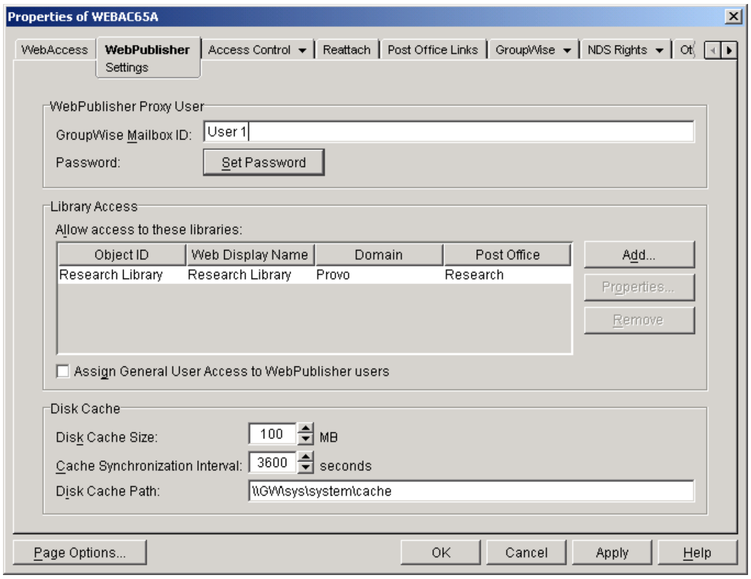
Modifying WebPublisher Settings

Using ConsoleOne, you can configure the following WebPublisher settings for the WebAccess Agent:

- The GroupWise account used by the WebAccess Agent to retrieve documents for WebPublisher users
- The GroupWise libraries where the WebAccess Agent will look for documents that have been shared with GroupWise WebPublisher users
- The maximum amount of disk space to use when caching documents that have been converted to HTML for viewing by GroupWise WebPublisher users
- How often to synchronize the cached documents with the original documents in the GroupWise libraries
- The location where the files will be cached

To modify the configuration information:

1. In ConsoleOne, right-click the WebAccess Agent object, then click Properties.
2. Click WebPublisher > Settings to display the WebPublisher Settings page.



WebPublisher Proxy User

GroupWise Mailbox ID: User 1

Password: Set Password

Library Access

Allow access to these libraries:

Object ID	Web Display Name	Domain	Post Office
Research Library	Research Library	Provo	Research

☐ Assign General User Access to WebPublisher users

Disk Cache

Disk Cache Size: 100 MB

Cache Synchronization Interval: 3600 seconds

Disk Cache Path: \\GW\sys\system\cache

3. Modify any of the following fields:

GroupWise Mailbox ID: The WebPublisher proxy user serves two purposes: 1) GroupWise users make documents available to GroupWise WebPublisher users by sharing the documents with the WebPublisher proxy user and 2) the WebAccess Agent logs in to GroupWise through the WebPublisher proxy user. This enables the WebAccess Agent to search for and retrieve documents that have been shared with the WebPublisher proxy user. Specify the ID for the GroupWise mailbox you want to use.

Password: Click Set Password to specify the mailbox password.

Allow Access to These Libraries: This list displays the libraries that the WebAccess Agent has access to. If a library is not in the list, WebPublisher users cannot see the library's documents. If a library is listed, WebPublisher users can view any of the library's documents that have been shared (by the document owner) with the WebPublisher proxy user.

To add a library to the list, click Add, then browse for and select the library.

To change the display name or description for the library, select the library, then click Properties. By default, the library's Novell® eDirectory™ object name is used for the display name.

To remove a library from the list, select the library, then click Remove.

Assign General User Access to WebPublisher Users: When sharing documents with GroupWise users, a document's owner can assign individual access rights and general access rights (through the General User Access option). The General User Access rights determine the access for all GroupWise users who do not receive individual access rights. For example, if a document's owner sets the General User Access to View, all GroupWise users with access to that library can view the document.

This option lets you determine whether or not you, as the GroupWise system administrator, want to give General User Access rights to WebPublisher users. For example, with this option enabled, WebPublisher users can view any documents that have General User Access set to View.

Disk Cache Size: When a GroupWise WebPublisher user requests a document from a GroupWise library, the WebAccess Agent retrieves the document, renders it to HTML, displays it to the GroupWise WebPublisher user, and then saves it to a disk cache. If the document is requested again, the cached version is used.

The disk cache size determines the maximum amount of disk space to be used for the cache. The default is 100 MB. If there is not room in the cache for a newly rendered document, the least recently requested document is removed from the cache to make room for the new document.

If GroupWise users are publishing large numbers of documents, you might want to increase the cache size. The advantage of a large cache is that cached documents are displayed more quickly to GroupWise WebPublisher users because the WebAccess Agent does not have to first render them to HTML. The disadvantage of a large cache is the disk space used and the amount of time required by the WebAccess Agent to keep the cached documents synchronized with the original documents.

Cache Synchronization Interval: The cache synchronization interval determines how often the WebAccess Agent checks for differences between the cached documents and the original documents in the library. Based on the default interval, the WebAccess Agent checks for differences every one hour (3600 seconds). If differences exist, the WebAccess Agent replaces the cached document with a newly rendered version of the original document.

Disk Cache Path: The disk cache path indicates the directory where documents are stored. By default, this is a directory on the WebAccess Agent's server (c:\groupwise\cache for Windows or sys:\system\cache for NetWare® or /opt/novell/groupwise/webpublisher/cache for Linux). If necessary, you can change the path to specify a new location.

To increase speed and reduce network traffic, we recommend that you keep the cache directory on the WebAccess Agent's server.

4 Click OK to save the changes.

Controlling WebAccess Agent Logging

The WebAccess Agent provides logging options to help you monitor the operation of the agent.

The following sections explain the how to control logging:

- "Controlling the Agent's Logging" on page 832
- "Assigning Operators to Receive Warning and Error Messages" on page 836

Controlling the Agent's Logging

The WebAccess Agent logs information to the console and to a log file on disk (by default, disk logging is turned off). You can control the following logging features:

- The type of information to log.
- Whether disk logging is on or off.
- How long to retain log files.
- The maximum amount of disk space to use for log files.
- Where to store log files.

You can control logging through ConsoleOne, WebAccess Agent startup switches, and the WebAccess Agent console. The following table shows which logging options you can control from each location.

	ConsoleOne	Startup Switches	NetWare Console	NetWare Console	Linux Console
Logging Level	Yes	Yes	Yes	Yes	No
Disk Logging	Yes	Yes	Yes	No	No
Maximum Log File Age	Yes	Yes	Yes	No	No
Maximum Disk Space	Yes	Yes	Yes	No	No
Log File Location	Yes	Yes	No	Yes	No

The log settings in ConsoleOne are used as the default settings. Startup switches override the ConsoleOne log settings, and agent console settings override startup switches and ConsoleOne settings for the current agent session. For information about modifying log settings through ConsoleOne, startup switches, or the WebAccess Agent console, see the following sections:

- "Modifying Log Settings in ConsoleOne" on page 833
- "Modifying Log Settings through Startup Switches" on page 834
- "Modifying Log Settings through the NetWare Agent's Console" on page 835
- "Modifying Log Settings through the Windows Agent's Console" on page 836
- "Modifying Log Settings on Linux" on page 836

Modifying Log Settings in ConsoleOne

Through ConsoleOne, you can select the following log settings:

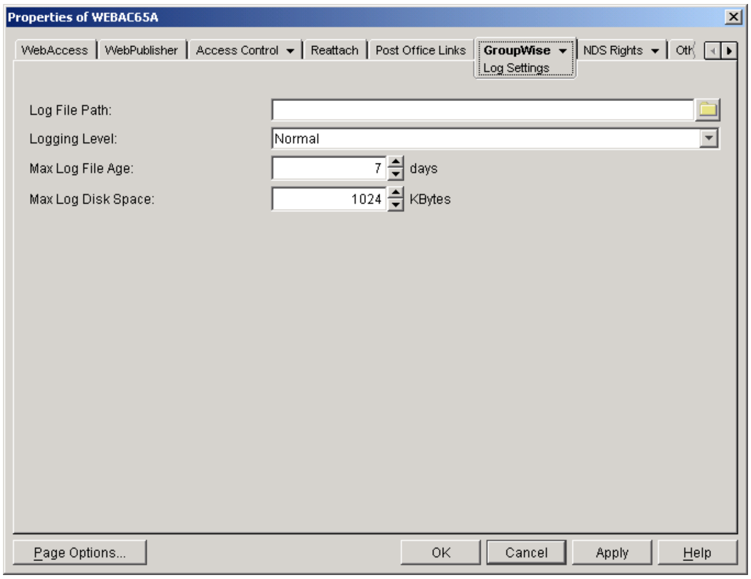
- Log file location
- Logging level
- Maximum age for log files
- Maximum disk spaced used for log files

By default, the WebAccess Agent does not log information to a file on disk on NetWare and Windows. (However, on Linux, it does.) To turn disk logging on, you can use the /logdiskon startup switch. See "Modifying Log Settings through Startup Switches" on page 834. If you are using the NetWare WebAccess Agent, you can turn disk logging on through the agent's console. See "Modifying Log Settings through the NetWare Agent's Console" on page 835.

The WebAccess Agent creates a new log file each day and each time it is started. The log file is named *mmdd*web.*nnn*, where *mm* is the month, *dd* is the day, and *nnn* is a sequenced number (001 for the first log file of the day, 002 for the second, and so forth). On NetWare and Windows, the default location for the log files is the *domain*\wpgate*webac65a*\000.prc directory. On Linux, the location is /var/log/novell/groupwise/*domain_name.gateway_name*/000.prc.

To modify log settings in ConsoleOne:

1. In ConsoleOne, right-click the WebAccess Agent object, then click Properties.
2. Click GroupWise > Log Settings to display the Log Settings page.



Log File Path:

Logging Level: Normal

Max Log File Age: 7 days

Max Log Disk Space: 1024 KBytes

3 Modify any of the following properties:

Log File Path: By default, this field is empty. If you have turned on disk logging by using the /logdiskon startup switch (see "Modifying Log Settings through Startup Switches" on page 834), the log files will be saved to the default directory or to the directory specified by the /log startup switch. If you want to specify a different location, enter the directory path or browse to and select the directory.

If you have not used the /logdiskon startup switch to turn on logging, entering a log file path will activate disk logging (after you restart the WebAccess Agent).

Logging Level: There are four logging levels: Off, Normal, Verbose, and Diagnostic. Off turns logging off; Normal displays initial statistics, user logins, warnings, and errors; Verbose displays normal logging plus user requests; and Diagnostic displays Verbose logging plus thread information. The default is Normal logging. Use Diagnostic only if you are troubleshooting a problem with WebAccess.

The verbose and diagnostic logging levels do not degrade WebAccess Agent performance, but log files saved to disk consume more disk space when verbose or diagnostic logging is in use.

Max Log File Age: Specify the number of days you want the WebAccess Agent to retain old log files. The WebAccess Agent will retain the log file for the specified number of days unless the maximum disk space for the log files is exceeded. The default age is 7 days.

Max Log Disk Space: Specify the maximum amount of disk space you want to use for log files. If the disk space limit is exceeded, the WebAccess Agent will delete log files, beginning with the oldest file, until the limit is no longer exceeded. The default disk space is 1024 KB.

4 Click OK to save the log settings.

Modifying Log Settings through Startup Switches

Startup switches override any log settings you specified through ConsoleOne. See "Modifying Log Settings in ConsoleOne" on page 833.

To use a switch, you can:

- Add the switch to the command line. For example, load gwinter.nlm /ph-j:\domain\wpgate\webac65a.

- On NetWare, include the switch in the WebAccess NetWare configuration file (strtweb.ncf), typically located in sys:\system.
- On Linux, include the switch in the WebAccess startup script (*gateway_name*.waa) located in /opt/novell/groupwise/agents/bin.
- On Windows, include the switch in the WebAccess startup batch file (strtweb.bat), typically located in c:\webacc.

For information about startup switches that can be used to modify log settings, see "Using WebAccess Agent Startup Switches" on page 895.

Modifying Log Settings through the NetWare Agent's Console

You can use the NetWare WebAccess Agent's console to modify the following log settings:

- Logging level
- Disk logging on or off
- Maximum age for log files
- Maximum disk space used for log files

Changes you make to log settings at the console apply only to the current session. When you restart the WebAccess Agent, the log settings are reset to the settings specified in ConsoleOne or the startup switches. See "Modifying Log Settings in ConsoleOne" on page 833 and "Modifying Log Settings through Startup Switches" on page 834.

To modify the log settings:

1. At the NetWare WebAccess Agent's console, press F10, select Logging Options, then modify any of the following settings:

   Logging Level: Off turns logging off; Normal displays initial statistics, user logins, warnings, and errors; Verbose displays normal logging plus user requests; and Diagnostic displays Verbose logging plus thread information. The default is Normal logging. Use Diagnostic only if you are troubleshooting a problem with GroupWise WebAccess.

   The verbose and diagnostic logging levels do not degrade WebAccess Agent performance, but log files saved to disk consume more disk space when verbose or diagnostic logging is in use.

   File Logging: Turns disk logging on or off. When disk logging is turned on, the WebAccess Agent creates a new log file each day and each time it is restarted. The log file is named *mmdd*web.*nnn*, where *mm* is the month, *dd* is the day, and *nnn* is a sequenced number (001 for the first log file of the day, 002 for the second, and so forth). On NetWare and Windows, the default location for the log files is the *domain*\wpgate*webac65a*\000.prc directory. On Linux, the default location is /var/log/novell/groupwise/*domain.webac65*a/000.prc.

   Max Log File Age: Specifies the number of days you want the WebAccess Agent to retain old log files. The WebAccess Agent will retain the log file for the specified number of days unless the maximum disk space for the log files is exceeded. The default age is 7 days.

   Max Log Disk Space: Specifies the maximum amount of disk space you want to use for log files. If the disk space limit is exceeded, the WebAccess Agent will delete log files, beginning with the oldest file, until the limit is no longer exceeded. The default disk space is 65536 KB.

2. Press Esc to save the information.

Modifying Log Settings through the Windows Agent's Console

You can use the Windows WebAccess Agent's console to modify the logging level. All other log settings must be modified through ConsoleOne or startup switches. See "Modifying Log Settings in ConsoleOne" on page 833 and "Modifying Log Settings through Startup Switches" on page 834.

Changes you make to the log level at the console apply only to the current session. When you restart the WebAccess Agent, the log level is reset to the level specified in ConsoleOne or the startup switches.

To modify the logging level:

1. In the NetWare WebAccess Agent's console (the DOS window), press F2 to cycle the log level between Normal, Verbose, and Diagnostic. Each level is described below:

   Normal: Normal displays initial statistics, user logins, warnings, and errors. This is the default level.

   Verbose: Verbose displays Normal logging plus user requests.

   Diagnostic: Diagnostic displays Verbose logging plus thread information. Use Diagnostic only if you are troubleshooting a problem with GroupWise WebAccess.

   The verbose and diagnostic logging levels do not degrade WebAccess Agent performance, but log files saved to disk consume more disk space when verbose or diagnostic logging is in use.

Modifying Log Settings on Linux

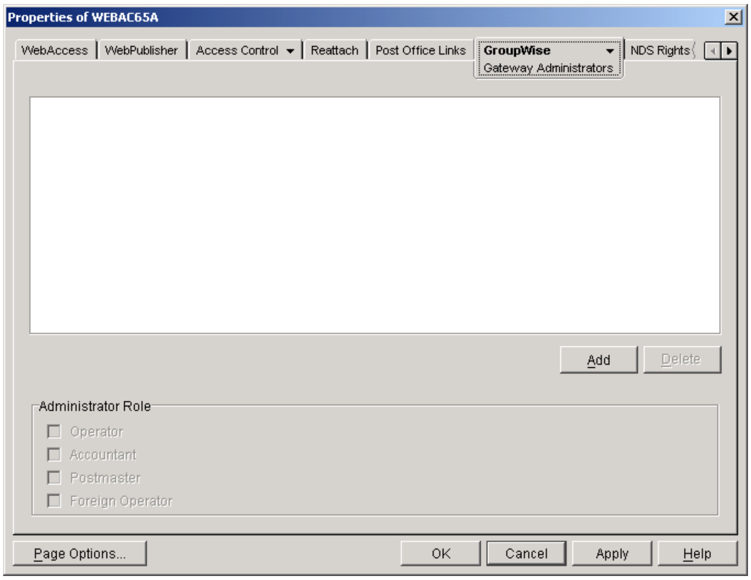
On Linux, these settings must be modified through ConsoleOne. See "Modifying Log Settings in ConsoleOne" on page 833.

Assigning Operators to Receive Warning and Error Messages

You can select GroupWise users to receive warning and error messages issued by the WebAccess Agent. Whenever the agent issues a warning or error, these users, called operators, receive a message in their mailboxes. You can specify one or more operators.

To assign an operator:

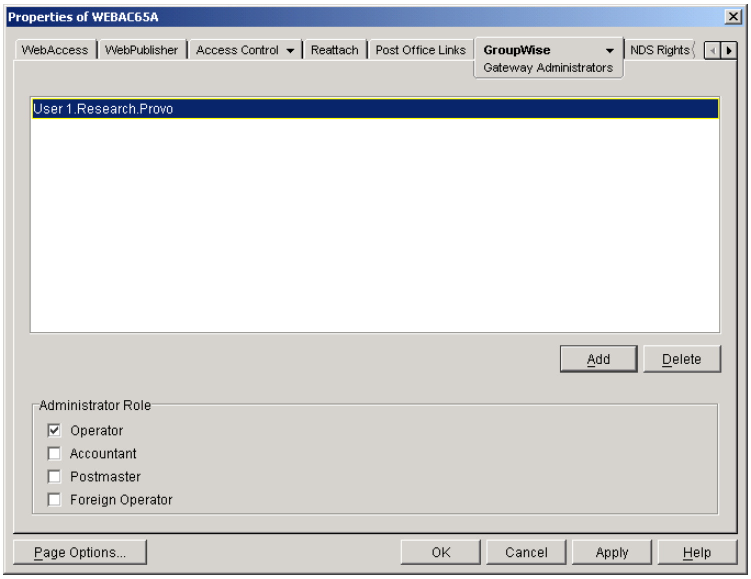
1. In ConsoleOne, right-click the WebAccess Agent object, then click Properties.
2. Click GroupWise > Gateway Administrators to display the Gateway Administrators page.



Administrator Role

□ Operator
□ Accountant
□ Postmaster
□ Foreign Operator

3 Click Add, select a user, then click OK to add the user to the Gateway Administrators list.



User 1.Research.Provo

Administrator Role

☑ Operator
□ Accountant
□ Postmaster
□ Foreign Operator

4 Make sure Operator is selected as the Administrator Role.

5 If desired, add additional operators.

6 Click OK.

Managing Access to Post Offices

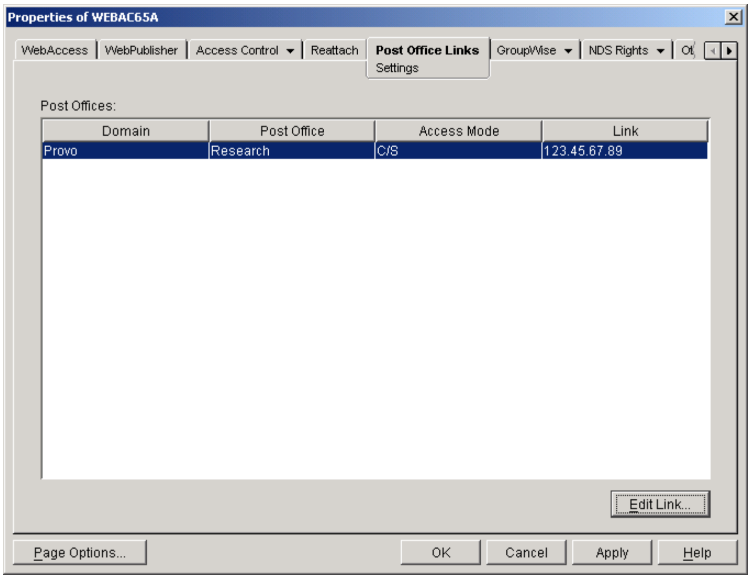
The WebAccess Agent requires access to all post offices where WebAccess users’ mailboxes or GroupWise libraries reside. The agent can access a post office using client/server mode, direct mode, or both. By default, it uses whichever mode is defined on the Post Office object’s Post Office Settings page (located on the GroupWise tab).

- “Modifying Links to Post Offices” on page 838 explains how to set the access mode to client/server, direct, or both.
- “Automating Reattachment to NetWare Servers” on page 839 explains how to configure the agent to automatically reconnect to post offices on NetWare servers.

Modifying Links to Post Offices

1 In ConsoleOne, right-click the WebAccess Agent object, then click Properties.

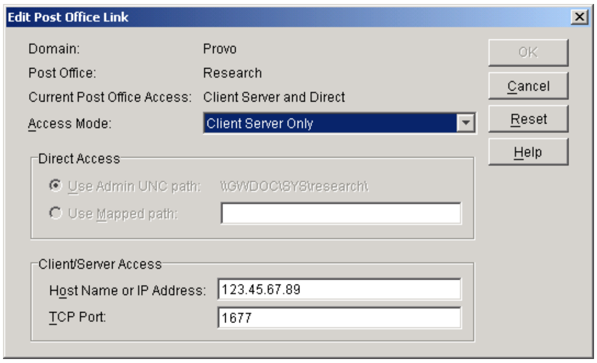
2 Click Post Office Links > Settings.



Post Offices:

Domain	Post Office	Access Mode	Link
Provo	Research	C/S	123.45.67.89

3 In the Post Offices list, select the post office whose link information you want to change, then click Edit Link to display the Edit Post Office Link dialog box.



Domain: Provo

Post Office: Research

Current Post Office Access: Client Server and Direct

Access Mode: Client Server Only

Direct Access

● Use Admin UNC path: \\GWDOC\SYS\research\

○ Use Mapped path:

Client/Server Access

Host Name or IP Address: 123.45.67.89

TCP Port: 1677

4 Define the following properties:

Access Mode: The access mode determines whether the WebAccess Agent will use client/server access, direct access, or both client/server and direct access to connect to the post office. With client/server and direct, the WebAccess Agent first tries client/server access; if client/server access fails, it then tries direct access. You can also choose to use the same access mode currently defined for the post office (on the Post Office object's Post Office Settings page). The current access mode is displayed in the Current Post Office Access field.

Direct Access: When connecting to the post office in direct mode, the WebAccess Agent can use the post office's UNC path (as defined on the Post Office object's Identification page) or a mapped path that you enter.

Client/Server Access: When connecting to the post office in client/server mode, the WebAccess Agent must know the hostname (or IP address) and port number of the Post Office Agent running against the post office.

5 Click OK.

6 Repeat Step 3 through Step 5 for each post office whose link you want to change.

Automating Reattachment to NetWare Servers

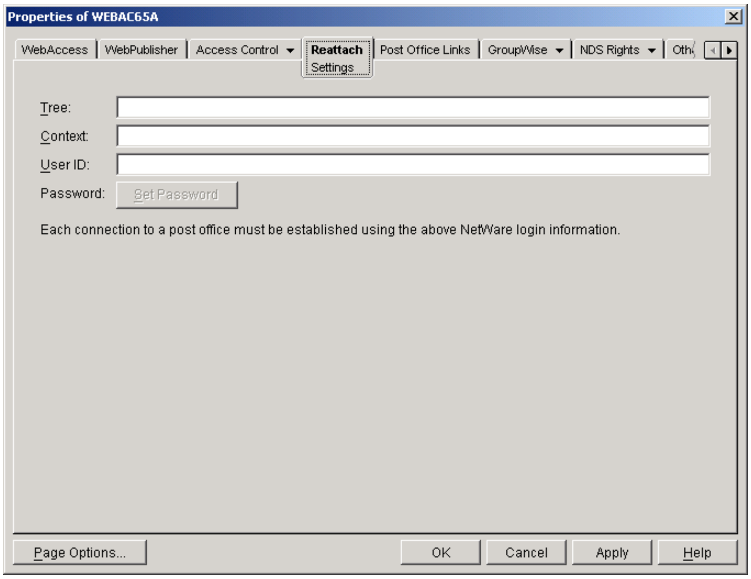
You can specify the reattach information for the Windows WebAccess Agent in ConsoleOne. Whenever the Windows WebAccess Agent loses its connection to a post office that is on a NetWare server, it will read the reattach information from the domain database and attempt to reattach to the NetWare server.

The NetWare WebAccess Agent does not use this information. To reattach to NetWare servers where users' post offices reside, the NetWare WebAccess Agent uses the user ID and password specified during installation. This user ID and password are entered in the strtweb.ncf file

To specify the reattachment information for the NetWare WebAccess Agent:

1 In ConsoleOne, right-click the WebAccess Agent object, then click Properties.

2 Click Reattach > Settings.



Each connection to a post office must be established using the above NetWare login information.

3 Define the following properties:

Tree: Enter the eDirectory tree that the WebAccess Agent logs in to. If the WebAccess Agent does not use an eDirectory user account, leave this field blank.

Context: Enter the eDirectory context of the WebAccess Agent's user account. If the WebAccess Agent does not use an eDirectory user account, leave this field blank.

User ID: Enter the name of the user account.

Password: Enter the password for the user account.

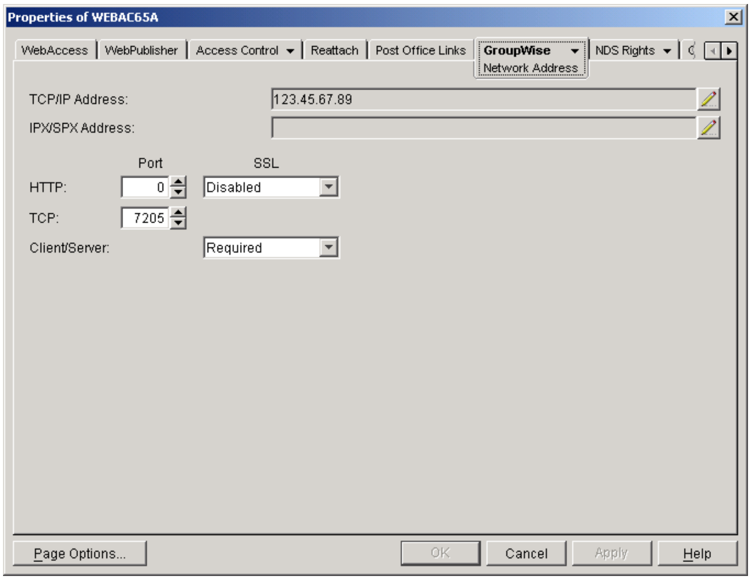
4 Click OK.

Changing the WebAccess Agent’s Network Address or Port Numbers

If you change the network address (IP address or DNS hostname) of the WebAccess Agent’s server or move the WebAccess Agent to a new server, you will need to change the network address in ConsoleOne. You can also change the port numbers used by the WebAccess Agent.

1 In ConsoleOne, right-click the WebAccess Agent object, then click Properties.

2 Click GroupWise > Network Address to display the Network Address page.

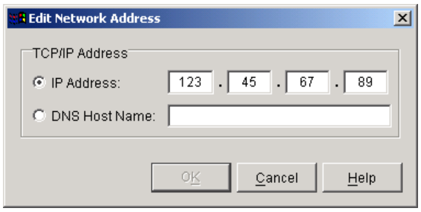


TCP/IP Address: 123.45.67.89

IPX/SPX Address:

	Port	SSL
HTTP:	0	Disabled
TCP:	7205	
Client/Server:		Required

3 To change the WebAccess Agent’s IP address, click the Edit button next to the TCP/IP Address field to display the Edit Network Address dialog box.



● IP Address: 123 . 45 . 67 . 89

○ DNS Host Name:

4 Change the IP address or DNS hostname as necessary, then click OK to return to the Network Address page.

5 To change the port numbers used by the WebAccess Agent, enter the new port number in the appropriate field.

HTTP Port: This is the port used to listen for requests from its Web console. The default port number is 7211.

TCP Port: This is the port used to listen for requests from the WebAccess Application and WebPublisher Application. The default port is 7205.

6 Click OK to save the changes.

Configuring the WebAccess Application

During installation, the WebAccess Application is set up with a default configuration. However, you can use the information in the following sections to optimize the WebAccess Application configuration:

- "Modifying the WebAccess Application Environment Settings" on page 841
- "Controlling WebAccess Application Logging" on page 842
- "Adding or Removing Service Providers" on page 844
- "Modifying WebAccess Application Template Settings" on page 845
- "Securing WebAccess Application Sessions" on page 850
- "Controlling Availability of WebAccess Features" on page 852

Modifying the WebAccess Application Environment Settings

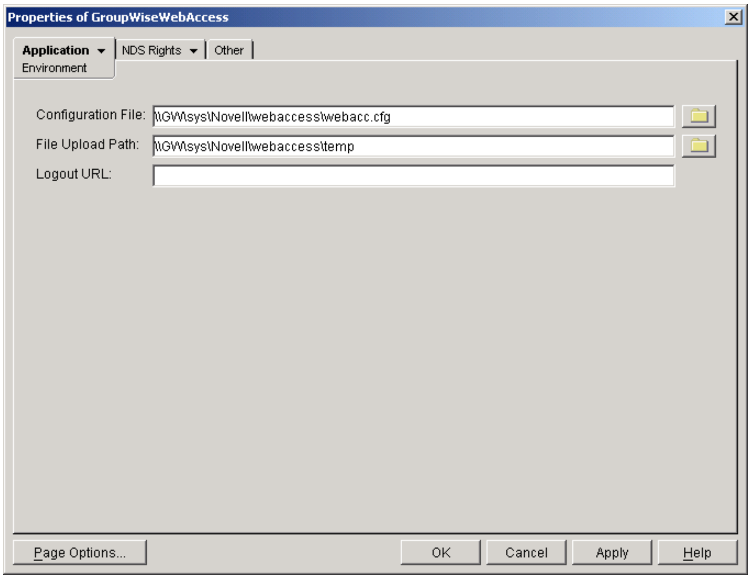
Using ConsoleOne$^{®}$, you can modify the WebAccess Application's environment settings. The environment settings determine such things as the location where ConsoleOne stores the WebAccess Application's configuration file and how long the WebAccess Application will maintain an open session with an inactive user.

To modify the environment settings:

1 In ConsoleOne, right-click the WebAccess Application object (GroupWiseWebAccess), then click Properties.

NOTE: The WebAccess Application object is not available in the GroupWise View. To locate the WebAccess Application object, you must use the Console View.

2 If necessary, click Applications > Environment to display the Environment page.



Configuration File: \\GW\sys\Novell\webaccess\webacc.cfg
File Upload Path: \\GW\sys\Novell\webaccess\temp
Logout URL:

3 Modify any of the following fields:

Configuration File: The WebAccess Application does not have access to Novell$^{®}$ eDirectory™ or the GroupWise$^{®}$ domain database. Therefore, ConsoleOne writes the application's configuration information to the file specified in this field. By default, this is the webacc.cfg file located in the WebAccess Application's home directory (novell\webaccess on the Web server or /opt/novell/groupwise/webaccess on Linux).

In general, you should avoid changing the location of the file. If you do, you need to make sure to modify the webacc.cfg path in the Java* servlet engine's property file or (web.xml for Tomcat or servlets.properties for the Novell Servlet Gateway). If you do not, the WebAccess Application will continue to look for its configuration information in the old location.

File Upload Path: When a user attaches a file to an item, the file is uploaded to the directory displayed in this field. By uploading the file before the item is sent, less time is required to send the item when the user clicks the Send button. After the user sends the item (or cancels it), the WebAccess Application deletes the file from the directory.

Specify the upload directory you want to use. The default path is to the temp directory, located in the WebAccess Application's home directory (by default, novell\webaccess\temp on the Web server or /opt/novell/groupwise/webaccess/temp on Linux).

Logout URL: By default, users who log out of GroupWise WebAccess are returned to the login page. If desired, you can enter the URL for a different page.

The logout URL can be defined in this location and two additional locations. These locations are listed below, in the order that the WebAccess Application will check them.

- Trusted server logout URL (configured on the Security page)
- Template-specific logout URL (configured on the Templates page)
- General logout URL (configured on the Environment page)

For example, you define a general logout URL (WebAccess Application object > Environment page) and a Standard HTML template logout URL (WebAccess Application object > Templates page). You are not using trusted servers, so you do not set any trusted server logout URLs.

When a Standard HTML template user logs out of WebAccess, the Standard HTML template logout URL is used. However, when a Basic HTML template user logs out, the general logout URL is used.

If none of these locations include a logout URL, the WebAccess Application defaults to the standard login page.

4. Click OK to save the changes.

Controlling WebAccess Application Logging

The WebAccess Application logs information to log files on disk. You can control the following logging features:

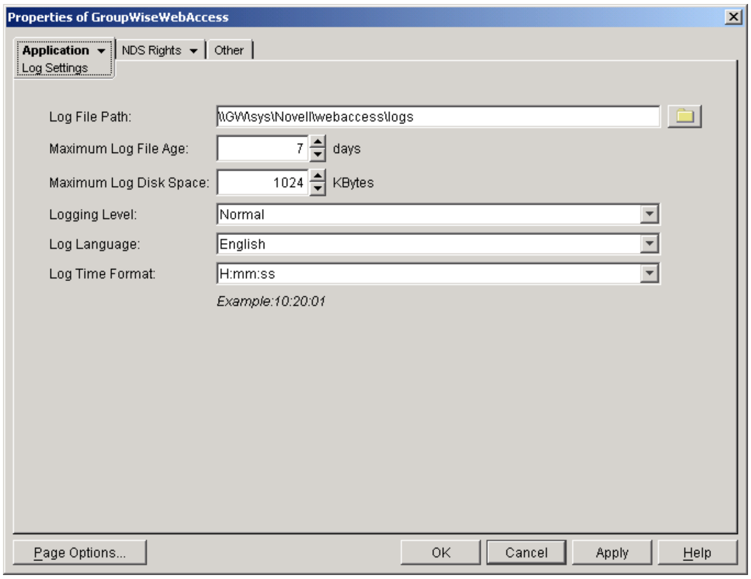
- The type of information to log
- How long to retain log files
- The maximum amount of disk space to use for log files
- Where to store log files

The WebAccess Application creates a new log file each day and each time it is restarted (as part of the Web server startup). The log file is named *mmdd*was.*nnn*, where *mm* is the month, *dd* is the year, and *nnn* is a sequenced log file number (001 for the first log file of the day, 002 for the second, and so forth).

To modify the log settings:

1 In ConsoleOne, right-click the WebAccess Application object, then click Properties.

2 Click Application > Log Settings to display the Log Settings page.



Log File Path: \\GW\sys\Novell\webaccess\logs

Maximum Log File Age: 7 days

Maximum Log Disk Space: 1024 KBytes

Logging Level: Normal

Log Language: English

Log Time Format: H:mm:ss

Example:10:20:01

3 Modify any of the following properties:

Log File Path: Specify the path to the directory where you want to store the log files.

On NetWare and Windows, the log files are stored in the novell\webaccess\logs directory on the Web server by default. On Linux, they are stored in /opt/novell/groupwise/webaccess/logs.

Maximum Log File Age: Specify the number of days you want to retain the log files. The WebAccess Application will retain the log file for the specified number of days unless the maximum disk space for the log files is exceeded. The default age is 7 days.

Maximum Log Disk Space: Specify the maximum amount of disk space you want to use for the log files. If the disk space limit is exceeded, the WebAccess Application will delete log files, beginning with the oldest file, until the limit is no longer exceeded. The default disk space is 1024 KB.

Logging Level: There are four logging levels: None, Normal, Verbose, and Diagnostic. None turns logging off; Normal displays warnings and errors; Verbose displays Normal logging plus information messages and user requests; and Diagnostic displays all possible information. The default is Normal logging. Use Diagnostic only if you are troubleshooting a problem with WebAccess.

The verbose and diagnostic logging levels do not degrade WebAccess Application performance, but log files saved to disk consume more disk space when verbose or diagnostic logging is in use.

Log Language: Select the language in which you want information written to the log files. The list contains many languages, some of which the WebAccess Application might not support. If you select an unsupported language, the information will be written in English.

Log Time Format: Choose from the following formats to use when the WebAccess Application records dates and times in the log files: HH:mm:ss:SS, MM/dd: H:mm:ss.SS, or dd/MM: H:mm:ss.SS. H and HH represent hours, mm represents minutes, ss and SS represent seconds, MM represents months, and dd represents days.

4 Click OK to save the log settings.

Adding or Removing Service Providers

The WebAccess Application receives requests from users and then passes the requests to the appropriate service provider. The service provider fills the requests and returns the required information to the WebAccess Application. The WebAccess Application merges the information into the appropriate template and displays it to the user.

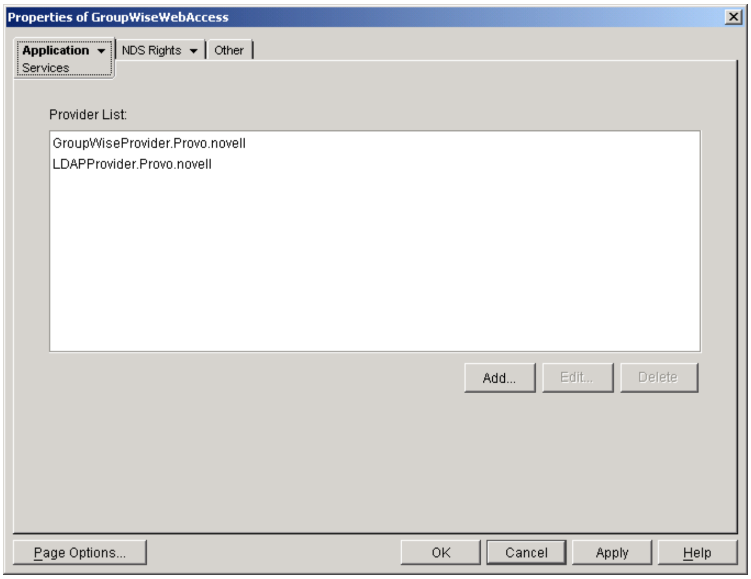
To function properly, the WebAccess Application must know which service providers are available. On NetWare and Windows, WebAccess includes two service providers: a GroupWise service provider (GroupWiseProvider) and an LDAP service provider (LDAPProvider). On Linux, there is also a separate GroupWiseDocumentProvider for WebPublisher. The GroupWise provider communicates with the WebAccess Agent to fill GroupWise requests. The LDAP provider communicates with LDAP servers to fill LDAP requests, such as LDAP directory searches initiated through the GroupWise Address Book.

Both the GroupWise service provider and the LDAP service provider are installed and configured at the same time as the WebAccess Application. You can disable the GroupWise service or LDAP service by removing the GroupWise service provider or LDAP service provider. On Linux, the GroupWiseDocumentProvider is also created by default. If you've created new service providers to expose additional services through GroupWise WebAccess, you must define those service providers so that the WebAccess Application knows about them.

To define service providers:

1. In ConsoleOne, right-click the WebAccess Application object, then click Properties.
2. Click Application > Services to display the Services page.

   The Provider List displays all service providers that the WebAccess Application is configured to use.



Properties of GroupWiseWebAccess

Provider List:

GroupWiseProvider.Provo.novell
LDAPProvider.Provo.novell

3. Choose from the following options:

   Add: To add a service provider to the list, click Add, browse for and select the service provider's object, then click OK.

   Edit: To edit a service provider's information, select the provider in the list, then click Edit. For information about the modifications you can make, see Chapter , "Configuring the

GroupWise Service Provider," on page 867 and Chapter , "Configuring the LDAP Service Provider," on page 868.

Delete: To remove a service provider from the list, select the provider, then click Delete.

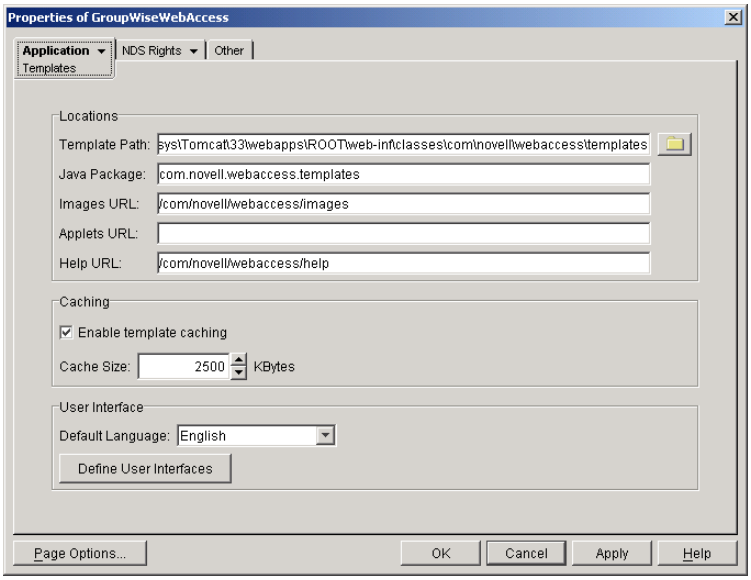
4 Click OK to save the changes.

Modifying WebAccess Application Template Settings

When the WebAccess Application receives information from a service provider, it merges the information into the appropriate WebAccess template before displaying the information to the user. Using ConsoleOne, you can modify the WebAccess Application's template settings. The template settings determine such things as the location of the templates, the maximum amount of server memory to use for caching the templates, and the default template language.

1 In ConsoleOne, right-click the WebAccess Application object, then click Properties.

2 Click Application > Templates to display the Templates page.



Template Path: sys\Tomcat\33\webapps\ROOT\web-inf\classes\com\novell\webaccess\templates

Java Package: com.novell.webaccess.templates

Images URL: /com/novell/webaccess/images

Applets URL:

Help URL: /com/novell/webaccess/help

☑ Enable template caching

Cache Size: 2500 KBytes

Default Language: English

3 Modify any of the following fields:

Template Path: Select the location of the template base directory. The template base directory contains the subdirectories (simple, frames, hdml, and wml) for each of the templates provided with GroupWise WebAccess. If you create your own templates, you need to place the templates in a new subdirectory in the template base directory.

On a NetWare® server with the Novell Servlet Gateway, the default installation directory is java\servlets\com\novell\webaccess\templates.

On a Windows server with the Novell Servlet Gateway, the default installation directory is novell\java\servlets\com\novell\webaccess\templates.

On a NetWare or Windows server with Tomcat, the default installation directory is *tomcat_dir*\webapps\ROOT\web-inf\classes\com\novell\webaccess\templates.

On a Linux server with Tomcat, the default installation directory is /var/opt/novell/tomcat/webapps/gw/WEB-INF/classes/com/novell/webaccess/templates.

Java Package: Specify the Java package that contains the template resources used by the WebAccess Application. The default package is com.novell.webaccess.templates.

Images URL: Specify the URL for the GroupWise WebAccess image files. These images are merged into the templates along with the GroupWise information. This URL must be relative to the Web server's document root directory. On NetWare and Windows, the default relative URL is /com/novell/webaccess/images. On Linux, the default relative URL is /gw/com/novell/webaccess/images.

Applets URL: In some instances (Address Book and Month Calendar, for example), applets can be used instead of the standard templates. Specify the URL for the GroupWise WebAccess applets (Address Book, Month Calendar, and so forth). This URL must be relative to the Web server's document root directory. On NetWare and Windows, the default relative URL is /com/novell/webaccess/applets. On Linux, the default relative URL is /gw/com/novell/webaccess/applets.

Help URL: Specify the URL for the GroupWise WebAccess Help files. This URL must be relative to the Web server's document root directory. On NetWare and Windows, the default relative URL is /com/novell/webaccess/help. On Linux, the default relative URL is /gw/com/novell/webaccess/help.

Enable Template Caching: To speed up access to the template files, the WebAccess Application can cache the files to the server's memory. Select this option to turn on template caching.

Cache Size: Select the maximum amount of memory, in kilobytes, you want to use when caching the templates. The default cache size, 2500 KB, is sufficient to cache all templates shipped with GroupWise WebAccess. If you modify or add templates, you can turn on Verbose logging (WebAccess Application object > Application tab > Log Settings page) to view the size of the template files. Using this information, you can then change the cache size appropriately.

Default Language: If you have more than one language installed, select the language to use when displaying the initial GroupWise WebAccess page. If users want the GroupWise WebAccess interface (templates) displayed in a different language, they can change it on the initial page.

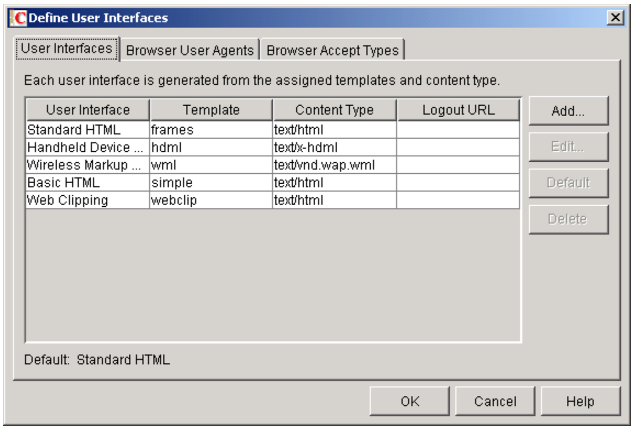
Define User Interfaces: GroupWise WebAccess supports Web browsers on many different devices (for example, computers and wireless telephones). Each device supports specific content types such as HTML, HDML, and WML. When returning information to a device's Web browser, the WebAccess Application must merge the information into a set of templates to create an interface that supports the content type required by the Web browser.

GroupWise WebAccess ships with five predefined user interfaces (Standard HTML, Basic HTML, Handheld Device Markup Language, Wireless Markup Language, and Web Clipping). These interfaces support Web browsers that require HTML, HDML, and WML content types. Click the User Interface button to view, add, modify, or delete user interfaces. For more information, see Defining User Interfaces below.

4 Click OK to save the changes.

Defining User Interfaces

1 From the WebAccess Application object's Templates page, click Define User Interfaces to display the Define User Interfaces dialog box.



Each user interface is generated from the assigned templates and content type.

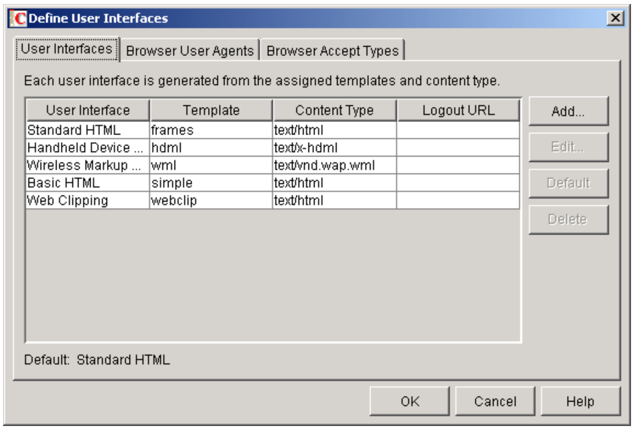
User Interface	Template	Content Type	Logout URL
Standard HTML	frames	text/html	
Handheld Device ...	hdml	text/x-hdml	
Wireless Markup ...	wml	text/vnd.wap.wml	
Basic HTML	simple	text/html	
Web Clipping	webclip	text/html	

Default: Standard HTML

The dialog box includes three tabs:

- **User Interfaces:** The User Interfaces tab lets you add, modify, and remove user interfaces, as well as determine whether or not GroupWise data added to an interface should be cached on proxy servers. Each interface consists of template files that support a specific content type. For example, the predefined Standard HTML interface uses frame-based HTML templates, located in the frames directory, that support the text/html content type.
- **Browser User Agents:** The Browser User Agents tab lets you associate a user interface with a Web browser. The association is based on the browser's User Agent information (signature, platform, version, and so forth). For example, if a browser's User Agent information includes "Windows CE" (one of the predefined entries), the WebAccess Application will use the Basic HTML interface (no-frames interface).
- **Browser Accept Types:** The Browser Accept Types tab lets you associate a user interface with a Web browser. The association is based on the content type the browser will accept. For example, if a browser accepts text/html (one of the predefined entries), the WebAccess Application will use the Standard HTML interface (frames-based interface).

2 To add, remove, or modify user interfaces, click the User Interfaces tab.



Each user interface is generated from the assigned templates and content type.

User Interface	Template	Content Type	Logout URL
Standard HTML	frames	text/html	
Handheld Device ...	hdml	text/x-hdml	
Wireless Markup ...	wml	text/vnd.wap.wml	
Basic HTML	simple	text/html	
Web Clipping	webclip	text/html	

Default: Standard HTML

The User Interface list displays all available user interfaces. The list includes the following information:

- **User Interface:** This column displays the name assigned to the user interface (for example, Standard HTML or Wireless Markup Language).

- **Template:** This column displays the directory in which the template files are located. Only the directory name is shown. You can append this directory name to the template path shown on the Templates page to see the full template directory path.

- **Content Type:** This column displays the content type required by the templates (for example, text/html, text/x-hdml, or text/vnd.wap.wml).

- **Logout URL:** By default, when a user logs out, he or she is returned to the standard login page. When adding or editing the user interface, you can use the logout URL to define a different page. If you do so, this column displays the URL. This URL overrides the logout URL specified on the WebAccess Application object's Environment page (see "Modifying the WebAccess Application Environment Settings" on page 841). It is overridden by the logout URL specified for a trusted server on the WebAccess Application object's Security page (see "Securing WebAccess Application Sessions" on page 850).

Choose from the following options to manage the user interfaces:

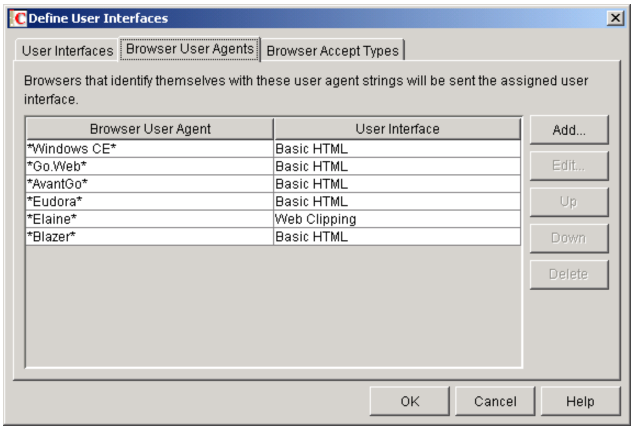
- **Add:** Click Add to add a user interface to the list.

- **Edit:** Select a user interface in the list, then click Edit to edit the interface's name, template directory, content type, or proxy caching setting.

- **Default:** Select a user interface in the list, then click Default to make that interface the default interface. The WebAccess Application will use the default interface only if it can't determine the appropriate interface based on the browser's User Agent (Browser User Agent tab) or the browser's accepted content types (Browser Accept Types tab).

- **Delete:** Select a user interface in the list, then click Delete to remove the interface. This only removes the entry from the list. It does not delete the template files from the template directory.

3 To associate a user interface with a Web browser based on the browser's User Agent information, click the Browser User Agents tab.



Browsers that identify themselves with these user agent strings will be sent the assigned user interface.

Browser User Agent	User Interface
Windows CE	Basic HTML
Go.Web	Basic HTML
AvantGo	Basic HTML
Eudora	Basic HTML
Elaine	Web Clipping
Blazer	Basic HTML

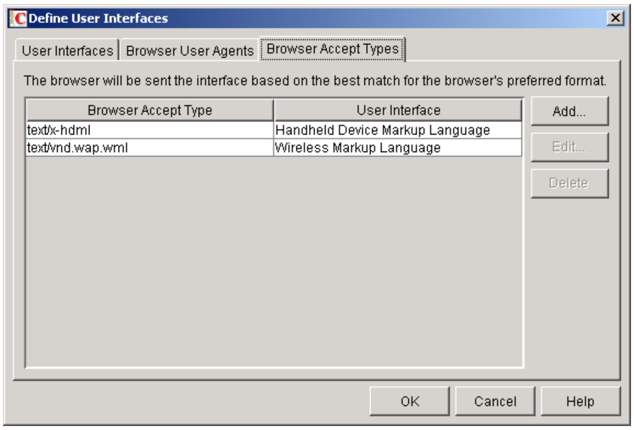
The Browser User Agents tab lets you associate a user interface with a Web browser. The association is based on the browser's User Agent information (signature, platform, version, and so forth). For example, if a browser's User Agent information includes "Windows CE" (one of the predefined entries), the WebAccess Application will use the Basic HTML interface (no-frames interface).

If a browser’s User Agent information matches more than one entry in the list, the application uses the first entry. If the browser’s User Agent information does not match any entries in the list, the WebAccess Application tries to select an interface based on the content types the browser will accept (Browser Accept Types tab). If no match is made based on the Accept Types information, the WebAccess Application uses the default user interface listed on the User Interfaces tab.

Choose from the following options to manage the associations:

- **Add:** Click Add to add an entry to the list.
- **Edit:** Select an entry from the list, then click Edit to edit the entry’s information.
- **Up:** Select an entry from the list, then click Up to move it up in the list. If two entries match the information in a browser’s User Agent header, the WebAccess Application uses the interface associated with the first entry listed.
- **Down:** Select an entry from the list, then click Down to move it down in the list.
- **Delete:** Select an entry from the list, then click Delete to remove the entry.

4 To associate a user interface with a Web browser based on the content type that the browser will accept, click the Browser Accept Types tab.



The browser will be sent the interface based on the best match for the browser's preferred format.

Browser Accept Type	User Interface
text/x-hdml	Handheld Device Markup Language
text/vnd.wap.wml	Wireless Markup Language

The Browser Accept Types tab lets you associate a user interface with a Web browser. The association is based on the content type the browser will accept. For example, if a browser accepts text/html (one of the predefined entries), the WebAccess Application will use the Standard HTML interface (frames-based interface).

Many browsers accept more than one content type (for example, both text/html and text/plain). If the list contains more than one acceptable content type, the WebAccess Application uses the browser’s preferred content type, which is the type that is listed first in the browser’s Accept Type header.

If no interface can be determined based on the entries in the list, the WebAccess Application uses the default user interface listed on the User Interfaces tab.

Choose from the following options to manage the associations:

- **Add:** Click Add to add an entry to the list.
- **Edit:** Select an entry from the list, then click Edit to edit the entry’s information.
- **Delete:** Select an entry from the list, then click Delete to remove the entry.

5 Click OK to save your changes and return to the WebAccess Application object’s Templates page.

Securing WebAccess Application Sessions

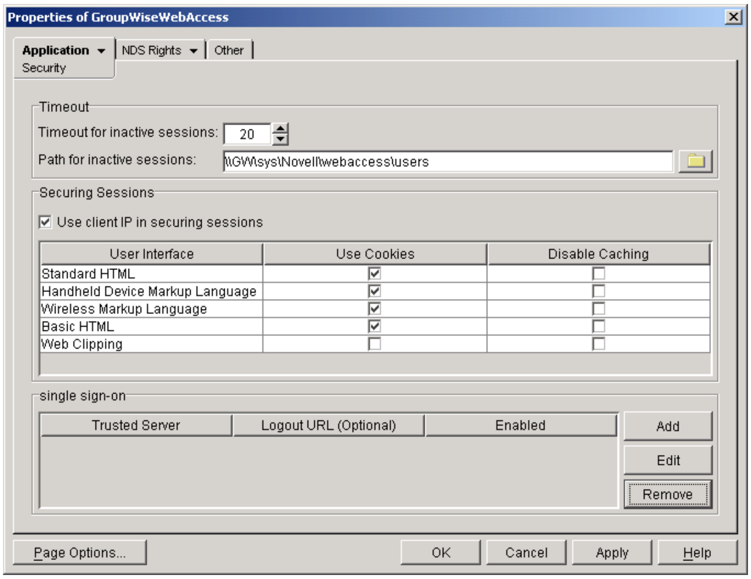
The WebAccess Application includes several settings to help you ensure that users' information is secure. You can:

- Specify a period of time after which inactive sessions will be closed. The default is 20 minutes.
- Secure sessions through the use of client IP binding or browser session cookies.
- Disable information caching by proxy servers and Web browsers.
- Enable GroupWise authentication through a trusted server.

To modify the security settings:

1 In ConsoleOne, right-click the WebAccess Application object, then click Properties.

2 Click Application > Security to display the Security page.



Timeout

Timeout for inactive sessions: 20

Path for inactive sessions: \\GW\sys\Novell\webaccess\users

Securing Sessions

☑ Use client IP in securing sessions

User Interface	Use Cookies	Disable Caching
Standard HTML	☑	□
Handheld Device Markup Language	☑	□
Wireless Markup Language	☑	□
Basic HTML	☑	□
Web Clipping	□	□

single sign-on

Trusted Server	Logout URL (Optional)	Enabled

3 Modify any of the following fields:

Timeout for Inactive Sessions: When a user logs in, the WebAccess Application opens a session with the user. This option lets you specify a period of time after which the WebAccess Application will close a session that has become inactive. A session becomes inactive when the user does not perform any actions, such as opening a message, that generate calls to the WebAccess Application. Having a timeout period not only provides security for users' e-mail but also ensures that GroupWise WebAccess runs efficiently.

Select how long the WebAccess Application should wait before ending an inactive session. If the user attempts to perform an action after the session has timed out, he or she will be prompted to log in again.

Path for Inactive Sessions: Browse for and select the folder where you want the WebAccess Application to save information about inactive sessions. This allows the WebAccess Application to return the user to the exact state he or she was in when the session timed out. Inactive sessions are automatically deleted after a period of time.

The default path is to the users directory, located in the WebAccess Application's home directory (by default, novell\webaccess\users on the Web server, or /opt/novell/groupwise/webaccess/users on Linux).

Use Client IP in Securing Sessions: Select this option if you want the WebAccess Application to bind the client IP address to the session. For that session, the WebAccess Application will accept requests from the bound IP address only. If you are using a proxy server that masks the client IP address, you should use the Use Cookies option instead.

User Interface/Use Cookies/Disable Caching: You can increase security by using session cookies and disabling caching of WebAccess information. Session cookies and caching are configurable on a per-user interface (template basis). For example, you could use session cookies and disable caching for the Standard HTML interface and not use session cookies or disable caching for the Wireless Markup Language interface.

- **Use Cookies:** Select this option if you want the WebAccess Application to use a session cookie to secure the user's session. The session cookie, which is created when the user opens the session, ties the session to the browser and ensures that the WebAccess Application will accept session requests from that browser only. The session cookie is held in memory and exists only as long as the user is logged in.

  By default, session cookies are enabled for all interfaces, with the exception of the Web Clippings interface, which does not support session cookies.

- **Disable Caching:** This option affects both Web browser caching and proxy server caching. Because the WebAccess Application sends sensitive mailbox information (such as message text and passwords) to users, caching of files by Web browsers and proxy servers can pose an information security risk.

  If you select the Disable Caching option, the WebAccess Application includes a "disable caching" request in the header of each file that it sends. By default, Web browsers honor this request and will not cache files that include the request. Proxy servers, on the other hand, might or might not honor the request, depending on how they are configured. If the proxy server honors the request, the file will not be cached; if it does not honor the request, the file will be cached, regardless of this setting.

Single Sign-On: The WebAccess Application supports authentication to GroupWise using Base64 authentication header credentials generated by a trusted server (for example, a Novell® iChain® Authentication Server). The authentication header generated by the trusted server must contain the username and password required to log the user into GroupWise. For this to occur, one of the following conditions must be met:

- The regular GroupWise username and password must match the credentials passed from the trusted server.

  or

- The LDAP authentication credentials used by each POA (if LDAP has been enabled) must match the credentials passed from the trusted server (ConsoleOne > Post Office object > GroupWise tab > Security page).

If the credentials passed from the trusted server match the credentials being used by the GroupWise system, then the GroupWise WebAccess login page is bypassed and the user has immediate access to the requested mailbox.

To specify a trusted server whose authentication header credentials will be accepted by the WebAccess Application, click Add to display the Add Trusted Server Information dialog box, then enter the server's IP address or DNS hostname. For more information about the fields in the Add Trusted Server Information dialog box, click the dialog box's Help button.

Controlling Availability of WebAccess Features

By default, WebAccess users can:

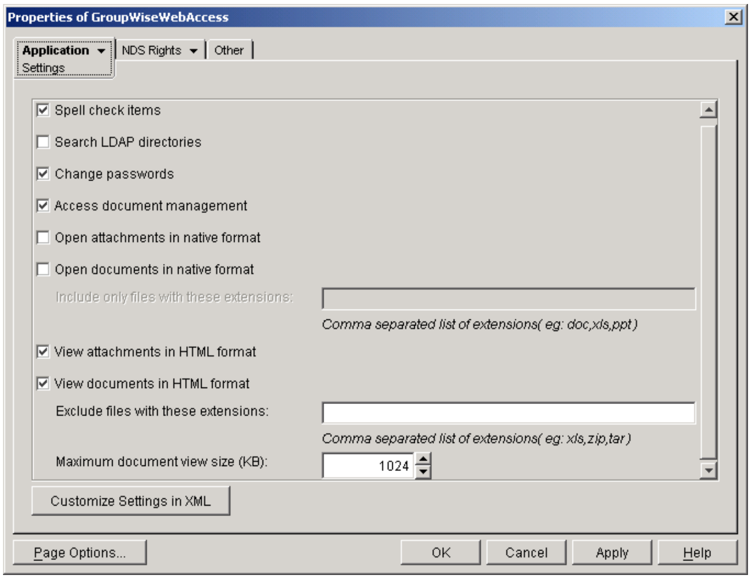
- Spell check messages
- Search LDAP directories (must be manually enabled on Linux)
- Change their GroupWise mailbox passwords
- Use Document Management Services
- Open attachments in native format (must be manually enabled on Linux)
- Open documents in native format (must be manually enabled on Linux)
- View attachments in HTML format
- View documents in HTML format

All users who log in through a single Web server will have the same feature access. You cannot configure individual user settings. However, if you have multiple Web servers, you can establish different settings for the Web servers by completing the following steps for each server's WebAccess Application.

To configure the WebAccess Application's user settings:

1 In ConsoleOne, right-click the WebAccess Application object, then click Properties.

2 Click Application > Settings to display the Settings page.



☑ Spell check items

☐ Search LDAP directories

☑ Change passwords

☑ Access document management

☐ Open attachments in native format

☐ Open documents in native format

Include only files with these extensions:

Comma separated list of extensions(eg: doc,xls,ppt)

☑ View attachments in HTML format

☑ View documents in HTML format

Exclude files with these extensions:

Comma separated list of extensions(eg: xls,zip,tar)

Maximum document view size (KB): 1024

3 Configure the following settings:

Spell Check Items: Enable this option if you want users to be able to use the Novell Speller to spell check an item's text before sending the item. Disable this option to remove all Spell Check features from the user interface.

Search LDAP Directories: Enable this option if you have an LDAP server and you want users to be able to search any LDAP address books you have defined. Disable this option to remove all LDAP features from the user interface.

Change Passwords: Enable this option if you want users to be able to change their Mailbox passwords. Disable this option to remove all Password features from the user interface.

Access Document Management: Enable this option if you want users to be able to use the Document Management features. Disable this option to remove all Document Management features from the user interface.

Open Attachments in Native Format: By default, the Save As option enables users to save message attachments to their local drives and then open them in their native applications. You can turn on this option to enable the Open option. The Open option enables users to open message attachments directly in their native applications without first saving the files to the local drive.

This option requires that 1) each user's Web browser knows the correct application or plug-in to associate with the attachment, according to its file extension or MIME type, and 2) the application or plug-in is available to the user. Otherwise, the user will be prompted to save the file to disk or specify the application to open it.

This option and the View Attachments in HTML Format option can both be enabled at the same time. Doing so gives users both the Open option and the View option, which means they have the choice of opening an attachment in its native application or viewing it as HTML.

Open Documents in Native Format: By default, the Save As option enables user to save library documents to their local drives and then open them in their native applications. You can turn on this option to enable the Open option. The Open option enables users to open documents directly in their native applications without first saving the files to the local drive.

This option requires that 1) each user's Web browser knows the correct application or plug-in to associate with the document, according to its file extension or MIME type, and 2) the application or plug-in is available to the user. Otherwise, the user will be prompted to save the file to disk or specify the application to open it.

This option and the View Documents in Native Format option can both be enabled at the same time. Doing so gives users both the Open option and the View option, which means they have the choice of opening a document in its native application or viewing it as HTML.

- **Include Only Files With These Extensions:** If you want only certain file types to be have the Open option, enter the file types in the Include Only Files With These Extensions field. Include only the extension and separate each extension with a comma (for example, doc, xls, ppt). The Open option will not be available for any file types not entered in this field. This setting applies when opening either library documents or attachments.

View Attachments in HTML Format: Enable this option if you want users to be able to view any type of attachments in HTML format. Disable this option to require users to save an attachment to a local drive and view it in its native application. WebAccess uses Stellent Outside In HTML Export to convert files to HTML format.

For a list of the supported file format conversions, download the following document from the Stellent Web site:

OutSide In Supported Platforms and File Formats (http://www.stellent.com/stellent3/groups/mkt/documents/nativepage/outside_in_supported_platforms.pdf)

This option and the Open Attachments in Native Format option can both be enabled at the same time. Doing so gives users both the View option and the Open option, which means they have the choice of viewing an attachment as HTML or opening it in its native application.

View Documents in HTML Format: Enable this option if you want users to be able to view library documents in HTML format. Disable this option to require users to save a document to a local drive and view it in its native application. WebAccess uses Stellent Outside In HTML Export to convert files to HTML format.

For a list of the supported file format conversions, download the following document from the Stellent Web site:

OutSide In Supported Platforms and File Formats (http://www.stellent.com/stellent3/groups/mkt/documents/nativepage/outside_in_supported_platforms.pdf)

This option and the Open Documents in Native Format option can both be enabled at the same time. Doing so gives users both the View option and the Open option, which means they have the choice of viewing a document as HTML or opening it in its native application.

- **Exclude Files With These Extensions:** If you want to exclude certain file types from having the View option, enter the file types in the Exclude Files With These Extensions field. Include only the extension and separate each extension with a comma (for example, doc, xls, ppt). The View option will be available for any file types not entered in this field. This setting applies when viewing either library documents or attachments.
- **Maximum Document View Size:** Specify the maximum size file that can be viewed in HTML format. If a file exceeds the maximum size, it must be opened in native format (if allowed) rather than viewed in HTML format. The default maximum size is 1024 KB. This setting applies when viewing either library documents or attachments.

4 Click OK.

Configuring the Novell Speller Application

The Novell® Speller Application enables users to spell check their messages. The Speller Application is installed automatically with the WebAccess Application. During installation, the Speller Application is set up with a default configuration. However, you can use the information in the following sections to optimize the Speller Application configuration:

- "Modifying the Speller Application Environment Settings" on page 854
- "Controlling Speller Application Logging" on page 855

Modifying the Speller Application Environment Settings

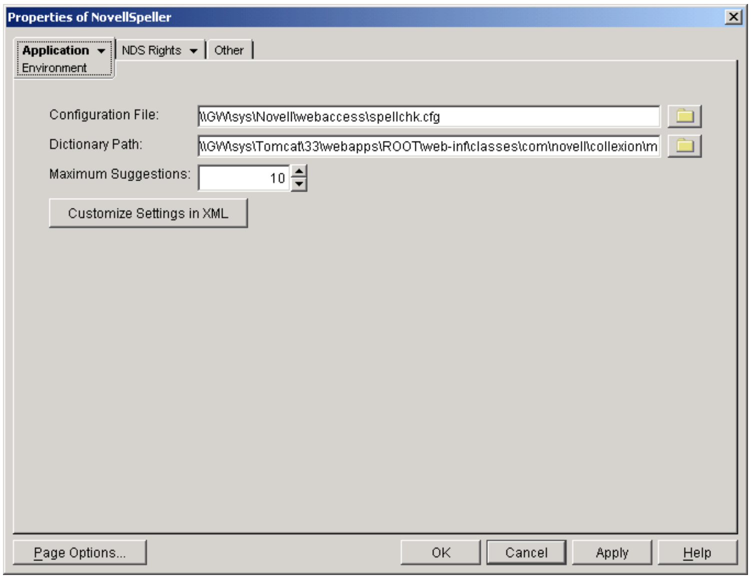
Using ConsoleOne®, you can modify the Speller Application's environment settings. The environment settings determine such things as the location where ConsoleOne stores the Speller Application's configuration file.

To modify the environment settings:

1 In ConsoleOne, right-click the Speller Application object (NovellSpeller), then click Properties.

NOTE: The Speller Application object is not available in the GroupWise View. To locate the Speller Application object, you must use the Console View.

2 If necessary, click Application > Environment to display the Environment page.



Configuration File: \\GW\sys\Novell\webaccess\spellchk.cfg

Dictionary Path: \\GW\sys\Tomcat\33\webapps\ROOT\web-inf\classes\com\novell\collexion\m

Maximum Suggestions: 10

3 Modify any of the following fields:

Configuration File: The Speller Application does not have access to Novell eDirectory™ or the GroupWise® domain database. Therefore, ConsoleOne writes the application's configuration information to the file specified in this field. By default, this is the spellchk.cfg file located in the WebAccess Application's home directory (novell\webaccess on the Web server or /opt/novell/groupwise/webaccess on Linux).

In general, you should avoid changing the location of the file. If you do change the location of the file, you need to make sure to modify the spellchk.cfg path in the Java servlet engine's properties file. If you do not, the Speller Application will continue to look for its configuration information in the old location.

Dictionary Path: Displays the path to the dictionary files used by the Speller Application.

On a NetWare® server with the Novell Servlet Gateway, the default installation directory is java\servlets\com\novell\collexion\morphology\data.

On a Windows server with the Novell Servlet Gateway, the default installation directory is novell\java\servlets\com\novell\collexion\morphology\data.

On a NetWare or Windows server with Tomcat, the default installation directory is *tomcat_dir*\webapps\ROOT\web-inf\classes\com\novell\collexion\morphology\data.

On Linux with Tomcat, the default installation directory is /var/opt/novell/tomcat/webapps/gw/WEB-INF/classes/com/novell/collexion/morphology/data.

Maximum Suggestions: Select the maximum number of suggestions the Speller Application will return for misspelled words. The default is 10.

Customize Settings in XML: Click this button to launch the XML editor. You can use the editor to add, modify, or delete settings.

4 Click OK to save the changes.

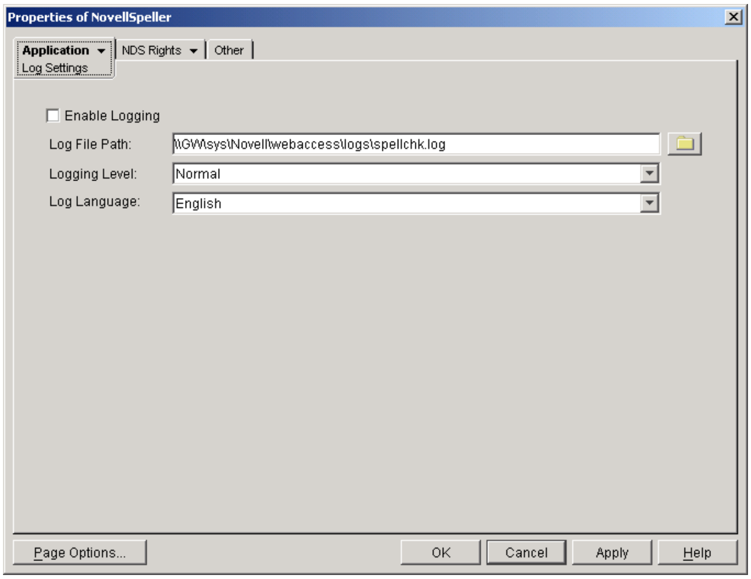
Controlling Speller Application Logging

The Speller Application can log information to a log file on disk. By default, logging is turned off. You can control the following logging features:

- Enable or disable logging
- Where to store the log file
- The type of information to log
- The language to use for the log file

To modify the log settings:

1. In ConsoleOne, right-click the Speller Application object, then click Properties.
2. Click Application > Log Settings to display the Log Settings page.



□ Enable Logging

Log File Path: \\GW\sys\Novell\webaccess\logs\spellchk.log

Logging Level: Normal

Log Language: English

3. Modify any of the following properties:

   Enable Logging: By default, logging is disabled. Select this option to enable it.

   Log File Path: Specify the path and filename for the log file.

   The log file is named spellchk.log by default. On NetWare and Windows, the log file is stored in the novell\webaccess\logs directory on the Web server by default. On Linux, the log file is stored in /opt/novell/groupwise/webaccess/logs.

   Logging Level: There are four logging levels: None, Normal, Verbose, and Diagnostic. None turns logging off; Normal displays warnings and errors; Verbose displays Normal logging plus information messages and user requests; and Diagnostic displays all possible information. The default is Normal logging. Use Diagnostic only if you are troubleshooting a problem with WebAccess.

   The verbose and diagnostic logging levels do not degrade Speller Application performance, but log files saved to disk consume more disk space when verbose or diagnostic logging is in use.

   Log Language: Select the language in which you want information written to the log files. The list contains many languages, some of which the Speller Application might not support. If you select an unsupported language, the information will be written in English.

4. Click OK to save the log settings.

Configuring the WebPublisher Application

During installation, the WebPublisher Application is set up with a default configuration. However, you can use the information in the following sections to optimize the WebPublisher Application configuration:

- "Modifying the WebPublisher Application Environment Settings" on page 857
- "Controlling WebPublisher Application Logging" on page 858
- "Adding or Removing Service Providers" on page 859
- "Modifying WebPublisher Application Template Settings" on page 860
- "Controlling Availability of WebPublisher Features" on page 865

Modifying the WebPublisher Application Environment Settings

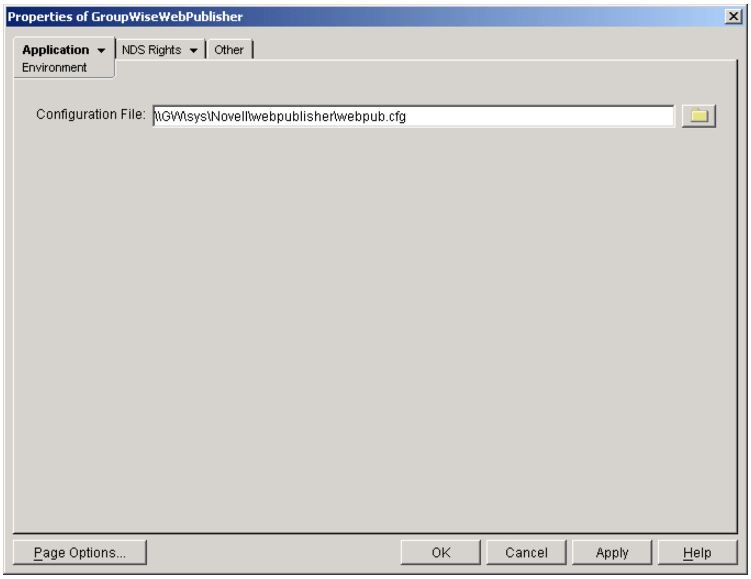
Using ConsoleOne®, you can modify the WebPublisher Application's environment settings. The environment settings determine such things as the location where ConsoleOne stores the WebPublisher Application's configuration file.

To modify the environment settings:

1 In ConsoleOne, right-click the WebPublisher Application object (GroupWiseWebPublisher), click Properties.

NOTE: The WebPublisher Application object is not available in the GroupWise View. To locate the WebPublisher Application object, you must use the Console View.

2 If necessary, click Application > Environment to display the Environment page.



Configuration File: \\GW\sys\Novell\webpublisher\webpub.cfg

3 Modify any of the following fields:

Configuration File: The WebPublisher Application does not have access to Novell® eDirectory™ or the GroupWise® domain database. Therefore, ConsoleOne writes the application's configuration information to the file specified in this field. By default, this is the webpub.cfg file located in the WebPublisher Application's home directory (novell\webpublisher on the Web server or /opt/novell/groupwise/webpublisher on Linux).

In general, you should avoid changing the location of the file. If you do change the location of the file, you need to make sure to modify the webpub.cfg path in the Java servlet engine's properties file. If you do not, the WebPublisher Application will continue to look for its configuration information in the old location.

4 Click OK to save the changes.

Controlling WebPublisher Application Logging

The WebPublisher Application logs information to log files on disk. You can control the following logging features:

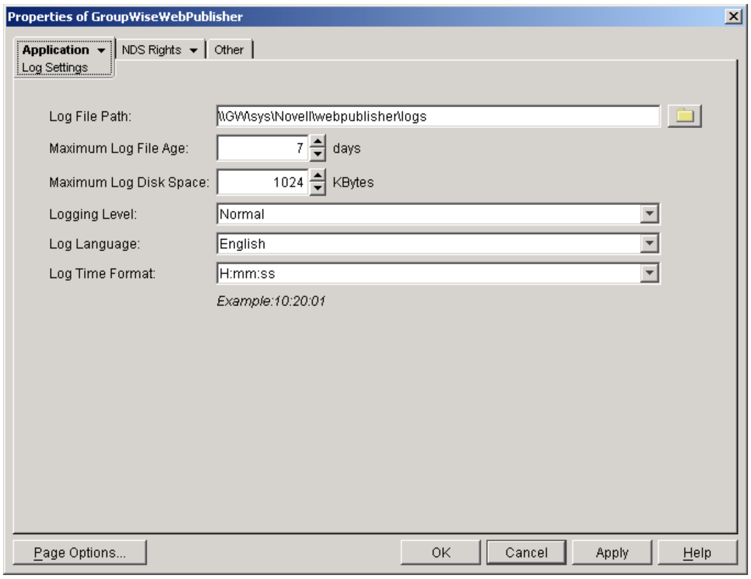
- The type of information to log
- How long to retain log files
- The maximum amount of disk space to use for log files
- Where to store log files

The WebPublisher Application creates a new log file each day and each time it is restarted (as part of the Web server startup). The log file is named *mmdd*wps.*nnn*, where *mm* is the month, *dd* is the year, and *nnn* is a sequenced log file number (001 for the first log file of the day, 002 for the second, and so forth).

To modify the log settings:

1 In ConsoleOne, right-click the WebPublisher Application object, then click Properties.

2 Click Application > Log Settings to display the Log Settings page.



Log File Path: \\GW\sys\Novell\webpublisher\logs

Maximum Log File Age: 7 days

Maximum Log Disk Space: 1024 KBytes

Logging Level: Normal

Log Language: English

Log Time Format: H:mm:ss

Example:10:20:01

3 Modify any of the following properties:

Log File Path: Specify the path to the directory where you want to store the log files.

On NetWare and Windows, the log files are stored in the novell\webpublisher\logs directory on the Web server by default. On Linux, the log files are stored in /opt/novell/groupwise/webpublisher/logs.

Maximum Log File Age: Specify the number of days you want to retain the log files. The WebPublisher Application will retain the log file for the specified number of days unless the maximum disk space for the log files is exceeded. The default age is 7 days.

Maximum Log Disk Space: Specify the maximum amount of disk space you want to use for the log files. If the disk space limit is exceeded, the WebPublisher Application will delete log files, beginning with the oldest file, until the limit is no longer exceeded. The default disk space is 1024 KB.

Logging Level: There are four logging levels: None, Normal, Verbose, and Diagnostic. None turns logging off; Normal displays warnings and errors; Verbose displays Normal logging plus information messages and user requests; and Diagnostic displays all possible information. The default is Normal logging. Use Diagnostic only if you are troubleshooting a problem with WebPublisher.

The verbose and diagnostic logging levels do not degrade WebPublisher Application performance, but log files saved to disk consume more disk space when verbose or diagnostic logging is in use.

Log Language: Select the language in which you want information written to the log files. The list contains many languages, some of which the WebPublisher Application might not support. If you select an unsupported language, the information will be written in English.

Log Time Format: Choose from the following formats to use when the WebPublisher Application records dates and times in the log files: HH:mm:ss:SS, MM/dd: H:mm:ss.SS, or dd/MM: H:mm:ss.SS. H and HH represent hours, mm represents minutes, ss and SS represent seconds, MM represents months, and dd represents days.

4 Click OK to save the log settings.

Adding or Removing Service Providers

The WebPublisher Application receives requests from users and then passes the requests to the appropriate service provider. The service provider fills the requests and returns the required information to the WebPublisher Application. The WebPublisher Application merges the information into the appropriate template and displays it to the user.

To function properly, the WebPublisher Application must know which service providers are available. By default, WebPublisher includes one service provider, the GroupWise Document service provider (GroupWiseDocumentProvider). The GroupWise Document service provider communicates with the WebAccess Agent to fill WebPublisher requests.

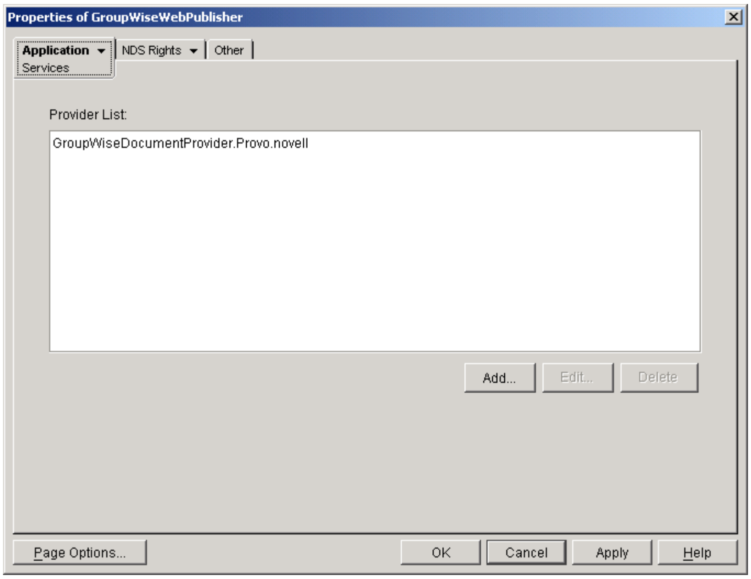
The GroupWise Document service provider is installed and configured at the same time as the WebPublisher Application. You can disable the GroupWise Document service by removing the GroupWise Document service provider. If you've created new service providers to expose additional services through GroupWise WebPublisher, you must define those service providers so that the WebPublisher Application knows about them.

To define service providers:

1 In ConsoleOne, right-click the WebPublisher Application object, then click Properties.

2 Click Application > Services to display the Services page.

The Provider List displays all service providers that the WebPublisher Application is configured to use.



3 Choose from the following options:

Add: To add a service provider to the list, click Add, browse for and select the service provider's object, then click OK.

Edit: To edit a service provider's information, select the provider in the list, then click Edit. For information about the modifications you can make, see Chapter , "Configuring the GroupWise Document Service Provider," on page 869.

Delete: To remove a service provider from the list, select the provider > click Delete.

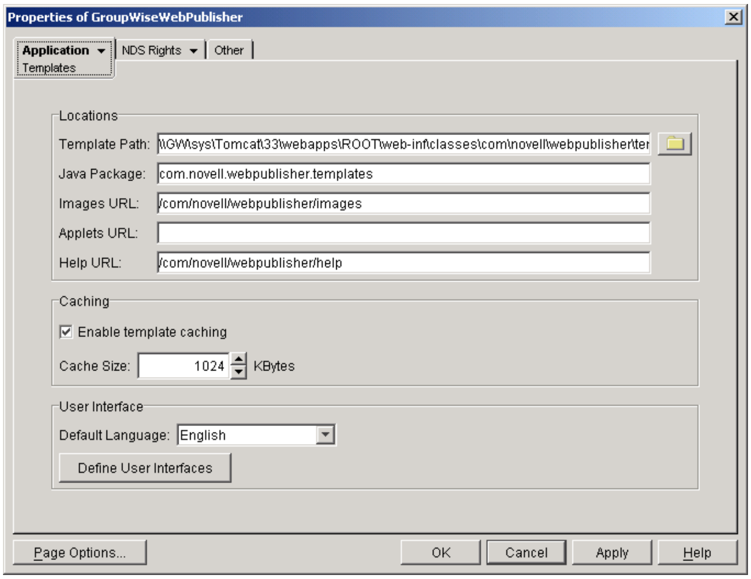
4 Click OK to save the changes.

Modifying WebPublisher Application Template Settings

When the WebPublisher Application receives information from a service provider, it merges the information into the appropriate WebPublisher template before displaying the information to the user. Using ConsoleOne, you can modify the WebPublisher Application's template settings. The template settings determine such things as the location of the templates, the maximum amount of server memory to use for caching the templates, and the default template language.

1 In ConsoleOne, right-click the WebPublisher Application object, then click Properties.

2 Click Application > Templates to display the Templates page.



Properties of GroupWiseWebPublisher

Locations

Template Path: \\GW\sys\Tomcat\33\webapps\ROOT\web-inf\classes\com\novell\webpublisher\ter

Java Package: com.novell.webpublisher.templates

Images URL: /com/novell/webpublisher/images

Applets URL:

Help URL: /com/novell/webpublisher/help

Caching

☑ Enable template caching

Cache Size: 1024 KBytes

User Interface

Default Language: English

3 Modify any of the following fields:

Template Path: Select the location of the template base directory. The template base directory contains the subdirectories for each of the templates provided with GroupWise WebAccess. Currently, only one template is provided for WebPublisher. This is an HTML template that uses frames; the template files are stored in the FRAMES subdirectory. If you create your own templates, you need to place the templates in a new subdirectory in the template base directory.

On a NetWare® server with the Novell Servlet Gateway, the default installation directory is java\servlets\com\novell\webpublisher\templates.

On a Windows server with the Novell Servlet Gateway, the default installation directory is novell\java\servlets\com\novell\webpublisher\templates.

On a NetWare or Windows server with Tomcat, the default installation directory is *tomcat_dir*\webapps\ROOT\web-inf\classes\com\novell\webpublisher\templates.

On a Linux server with Tomcat, the default installation directory is /var/opt/tomcat/webapps/gw/WEB-INF/classes/com/novell/webpublisher/templates.

Java Package: Specify the Java package that contains the template resources used by the WebPublisher Application. The default package is com.novell.webpublisher.templates.

Images URL: Specify the URL for the GroupWise WebPublisher image files. These images are merged into the templates along with the GroupWise document information. This URL must be relative to the Web server's document root directory. On NetWare and Windows, the default relative URL is /com/novell/webpublisher/images. On Linux, the default relative URL is /gw/com/novell/webpublisher/images.

Applets URL: GroupWise WebPublisher does not include any applets. If you create GroupWise WebPublisher applets, you need to specify the URL for the applets. To mirror the storage location of the GroupWise WebAccess applets, you can store the applets in a com\novell\webpublisher\applets directory under the Web server's document root directory. The applets URL would then be relative to the Web server's document root directory (for example, /com/novell/webpublisher/applets on NetWare or Windows, and /gw/com/novell/webpublisher/applets on Linux).

Help URL: Specify the URL for the GroupWise WebPublisher Help files. This URL must be relative to the Web server's document root directory. On NetWare and Windows, the default relative URL is /com/novell/webpublisher/help. On Linux, the default relative URL is /gw/com/novell/webpublisher/help.

Enable Template Caching: To speed up access to the template files, the WebPublisher Application can cache the files to the server's memory. Select this option to turn on template caching.

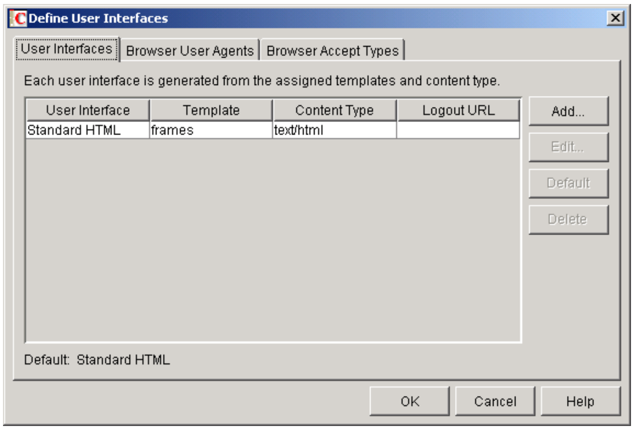
Cache Size: Select the maximum amount of memory, in kilobytes, you want to use when caching the templates. The default cache size, 1024 KB, is sufficient to cache all templates shipped with GroupWise WebPublisher. If you modify or add templates, you can turn on Verbose logging (WebPublisher Application object > Application tab > Log Settings page) to view the size of the template files. Using this information, you can then change the cache size appropriately.

Default Language: Select the language to use when displaying the initial GroupWise WebPublisher page. If users want the GroupWise WebPublisher interface (templates) displayed in a different language, they can change it on the initial page.

4 Click OK to save the changes.

Defining User Interfaces

1 From the WebPublisher Application object's Templates page, click Define User Interfaces to display the Define User Interfaces dialog box.



Each user interface is generated from the assigned templates and content type.

User Interface	Template	Content Type	Logout URL
Standard HTML	frames	text/html	

Default: Standard HTML

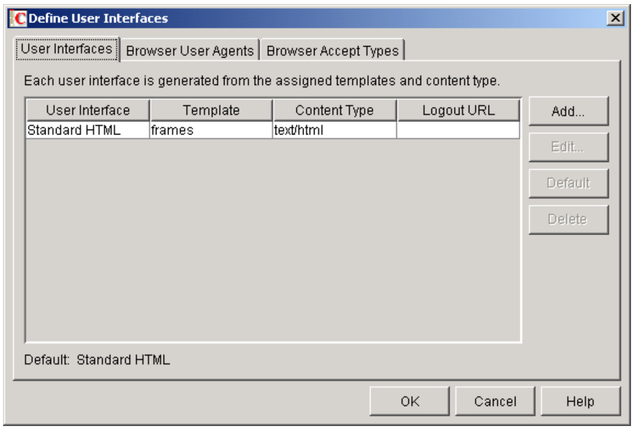
The dialog box includes three tabs:

- **User Interfaces:** The User Interfaces tab lets you add, modify, and remove user interfaces, as well as determine whether or not GroupWise data added to an interface should be cached on proxy servers. Each interface consists of template files that support a specific content type. For example, the predefined Standard HTML interface uses frame-based HTML templates, located in the frames directory, that support the text/html content type.

- **Browser User Agents:** The Browser User Agents tab lets you associate a user interface with a Web browser. The association is based on the browser's User Agent information (signature, platform, version, and so forth).

- **Browser Accept Types:** The Browser Accept Types tab lets you associate a user interface with a Web browser. The association is based on the content type the browser will accept.

2 To add, remove, or modify user interfaces, click the User Interfaces tab.



Each user interface is generated from the assigned templates and content type.

User Interface	Template	Content Type	Logout URL
Standard HTML	frames	text/html	

Default: Standard HTML

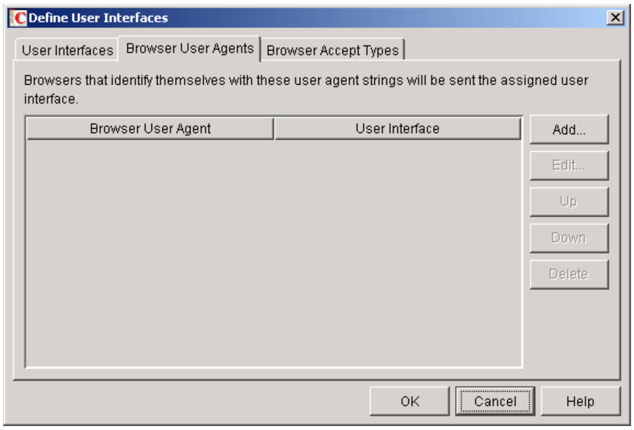
The User Interface list displays all available user interfaces. The list includes the following information:

- **User Interface:** This column displays the name assigned to the user interface (for example, Standard HTML).
- **Template:** This column displays the directory in which the template files are located. Only the directory name is shown. You can append this directory name to the template path shown on the Templates page to see the full template directory path.
- **Content Type:** This column displays the content type required by the templates (for example, text/html, text/x-hdml, or text/vnd.wap.wml).
- **Logout URL:** By default, when a user logs out, he or she is returned to the standard login page. When adding or editing the user interface, you can use the logout URL to define a different page. If you do so, this column displays the URL. This URL overrides the logout URL specified on the WebPublisher Application object's Environment page (see "Modifying the Speller Application Environment Settings" on page 854).

Choose from the following options to manage the user interfaces:

- **Add:** Click Add to add a user interface to the list.
- **Edit:** Select a user interface in the list, then click Edit to edit the interface's name, template directory, content type, or proxy caching setting.
- **Default:** Select a user interface in the list, then click Default to make that interface the default interface. The WebPublisher Application will use the default interface only if it can't determine the appropriate interface based on the browser's User Agent (Browser User Agent tab) or the browser's accepted content types (Browser Accept Types tab).
- **Delete:** Select a user interface in the list, then click Delete to remove the interface. This only removes the entry from the list. It does not delete the template files from the template directory.

3 To associate a user interface with a Web browser based on the browser's User Agent information, click the Browser User Agents tab.



Browsers that identify themselves with these user agent strings will be sent the assigned user interface.

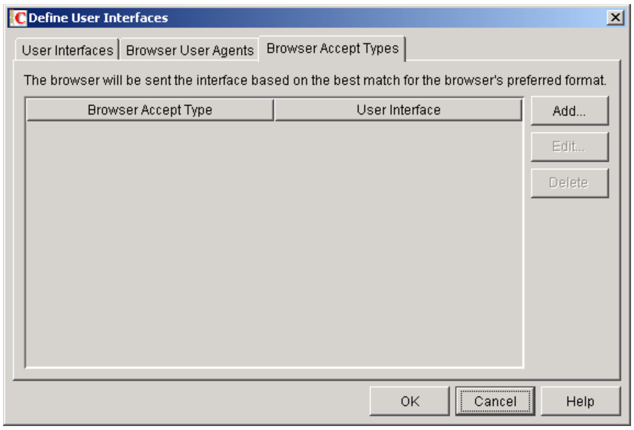
The Browser User Agents tab lets you associate a user interface with a Web browser. The association is based on the browser's User Agent information (signature, platform, version, and so forth). For example, if a browser's User Agent information includes "Windows CE" and you've created a specialized Windows CE user interface (templates), you could associate the User Agent and user interface so that Windows CE users would see your specialized Windows CE user interface.

If a browser's User Agent information matches more than one entry in the list, the application uses the first entry. If the browser's User Agent information does not match any entries in the list, the WebPublisher Application tries to select an interface based on the content types the browser will accept (Browser Accept Types tab). If no match is made based on the Accept Types information, the WebPublisher Application uses the default user interface listed on the User Interfaces tab.

Choose from the following options to manage the associations:

- **Add:** Click Add to add an entry to the list.
- **Edit:** Select an entry from the list, then click Edit to edit the entry's information.
- **Up:** Select an entry from the list, then click Up to move it up in the list. If two entries match the information in a browser's User Agent header, the WebPublisher Application uses the interface associated with the first entry listed.
- **Down:** Select an entry from the list, then click Down to move it down in the list.
- **Delete:** Select an entry from the list, then click Delete to remove the entry.

4 To associate a user interface with a Web browser based on the content type that the browser will accept, click the Browser Accept Types tab.



The browser will be sent the interface based on the best match for the browser's preferred format.

Browser Accept Type	User Interface

The Browser Accept Types tab lets you associate a user interface with a Web browser. The association is based on the content type the browser will accept.

Many browsers accept more than one content type (for example, both text/html and text/plain). If the list contains more than one acceptable content type, the WebPublisher Application uses the browser's preferred content type, which is the type that is listed first in the browser's Accept Type header.

If no interface can be determined based on the entries in the list, the WebPublisher Application uses the default user interface listed on the User Interfaces tab.

Choose from the following options to manage the associations:

- **Add:** Click Add to add an entry to the list.
- **Edit:** Select an entry from the list, then click Edit to edit the entry's information.
- **Delete:** Select an entry from the list, then click Delete to remove the entry.

5 Click OK to save your changes and return to the WebPublisher Application object's Templates page.

Controlling Availability of WebPublisher Features

WebPublisher users can:

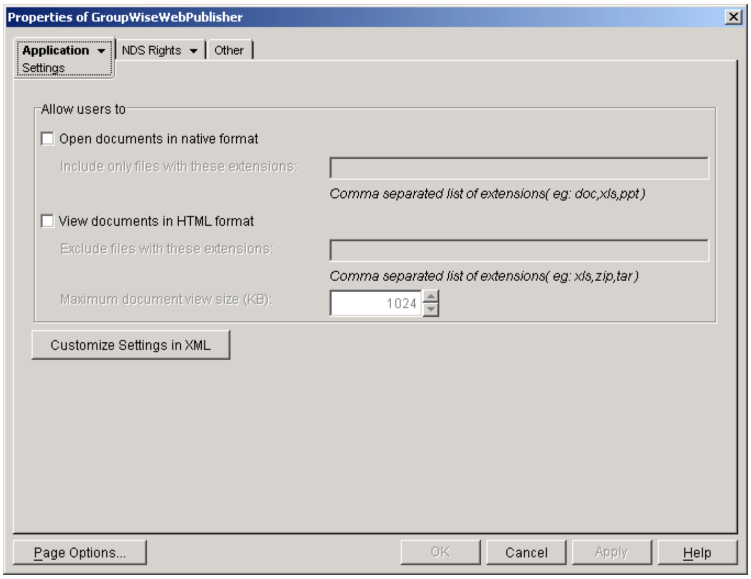
- View documents in HTML format.
- Open documents in native format.

All users who access WebPublisher through a single Web server will have the same feature access. You cannot configure individual user settings. However, if you have multiple Web servers, you can establish different settings for the Web servers by completing the following steps for each server's WebPublisher Application.

To configure the WebPublisher Application's user settings:

1 In ConsoleOne, right-click the WebAccess Application object, then click Properties.

2 Click Application > Settings to display the Settings page.



Properties of GroupWiseWebPublisher

Allow users to

□ Open documents in native format

Include only files with these extensions:

Comma separated list of extensions(eg: doc,xls,ppt)

□ View documents in HTML format

Exclude files with these extensions:

Comma separated list of extensions(eg: xls,zip,tar)

Maximum document view size (KB): 1024

3 Configure the following settings:

Open Documents in Native Format: By default, the Save As option enables user to save library documents to their local drives and then open them in their native applications. You can turn on this option to enable the Open option. The Open option enables users to open documents directly in their native applications without first saving the files to the local drive.

This option requires that 1) each user's Web browser knows the correct application or plug-in to associate with the document, according to its file extension or MIME type, and 2) the application or plug-in is available to the user. Otherwise, the user will be prompted to save the file to disk or specify the application to open it.

This option and the View Documents in Native Format option can both be enabled at the same time. Doing so gives users both the Open option and the View option, which means they have the choice of opening a document in its native application or viewing it as HTML.

- **Include Only Files With These Extensions:** If you want only certain file types to be have the Open option, enter the file types in the Include Only Files With These Extensions field. Include only the extension and separate each extension with a comma (for example, doc, xls, ppt). The Open option will not be available for any file types not entered in this field.

View Documents in HTML Format: Enable this option if you want users to be able to view library documents in HTML format. Disable this option to require users to save a document to a local drive and view it in its native application. WebAccess uses Stellent Outside In HTML Export to convert files to HTML format.

For a list of the supported file format conversions, download the following document from the Stellent Web site:

OutSide In Supported Platforms and File Formats (http://www.stellent.com/stellent3/groups/mkt/documents/nativepage/outside_in_supported_platforms.pdf)

This option and the Open Documents in Native Format option can both be enabled at the same time. Doing so gives users both the View option and the Open option, which means they have the choice of viewing a document as HTML or opening it in its native application.

- **Exclude Files With These Extensions:** If you want to exclude certain file types from having the View option, enter the file types in the Exclude Files With These Extensions field. Include only the extension and separate each extension with a comma (for example, doc, xls, ppt). The View option will be available for any file types not entered in this field.

- **Maximum Document View Size:** Specify the maximum size file that can be viewed in HTML format. If a file exceeds the maximum size, it must be opened in native format (if allowed) rather than viewed in HTML format. The default maximum size is 1024 KB.

4 Click OK.

Configuring the GroupWise Service Provider

The GroupWise® service provider is installed and configured when you install the WebAccess Application to a Web server. The GroupWise service provider receives GroupWise requests from the WebAccess Application and communicates with the WebAccess Agent to fill the requests.

The WebAccess installation program creates a Novell® eDirectory™ object for the GroupWise service provider in the same context as the WebAccess Application. The object is named GroupWiseProvider. Using ConsoleOne®, you can modify the GroupWiseProvider object to:

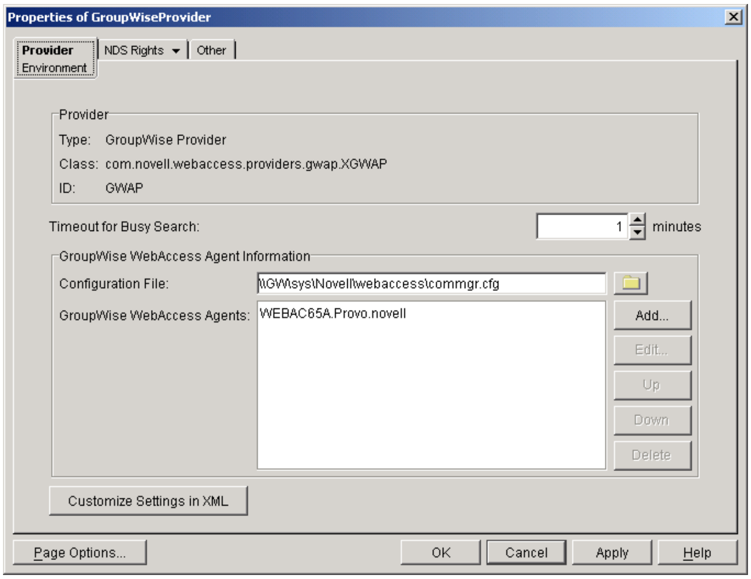
- Change how long the service provider will wait for the WebAccess Agent to return information for a Busy Search. Users can perform Busy Searches when scheduling appointments to ensure that the appointment's recipients will be available at the scheduled time. The default timeout interval is 1 minute.
- Define the WebAccess Agents that the service provider will contact to fill GroupWise requests. If your GroupWise system includes more than one WebAccess Agent, you can use this feature to provide failover support.

To modify the GroupWise service provider's configuration:

1 In ConsoleOne, right-click the GroupWise service provider object (GroupWiseProvider), then click Properties.

NOTE: The GroupWise service provider object is not available in the GroupWise View. To locate the GroupWise service provider object, you must use the Console View.

2 If necessary, click Provider > Environment to display the Environment page.



Properties of GroupWiseProvider

Provider

Type: GroupWise Provider

Class: com.novell.webaccess.providers.gwap.XGWAP

ID: GWAP

Timeout for Busy Search: 1 minutes

GroupWise WebAccess Agent Information

Configuration File: \\GW\sys\Novell\webaccess\commgr.cfg

GroupWise WebAccess Agents: WEBAC65A.Provo.novell

3 Choose from the following options:

Timeout for Busy Search: Select how long you want the GroupWise service provider to wait for the WebAccess Agent to return information when a user performs a Busy Search.

Configuration File: The WebAccess Agent's configuration file (commgr.cfg) contains the agent's IP address and the encryption key required by the GroupWise service provider to communicate with the WebAccess Agent. By default, the commgr.cfg file is stored in the WebAccess Application's home directory (novell\webaccess on the Web server or /opt/novell/groupwise/webaccess on Linux).

In general, you should not need to change this setting. However, if you have multiple WebAccess Agents in your GroupWise system and you are optimizing WebAccess to provide greater scalability and availability, you might need to change the setting. For information, see "Configuring Redirection and Failover Support" on page 810.

GroupWise WebAccess Agents: This list displays the WebAccess Agents the GroupWise service provider can communicate with when attempting to complete a request. If the first one listed is unavailable, the GroupWise service provider will attempt to use the second, third, fourth, and so on until it is successful. This provides failover support and ensures greater availability for your WebAccess users. For more information about optimizing availability, see "Configuring Redirection and Failover Support" on page 810.

The list must include at least one WebAccess Agent.

To add a WebAccess Agent to the list, click Add to browse for and select the WebAccess Agent object, then click OK.

To edit a WebAccess Agent's information, select the WebAccess Agent in the list, then click Edit.

To remove a WebAccess Agent from the list, select the WebAccess Agent in the list, then click Delete.

Customize Settings in XML: Click this button to launch the XML editor. You can use the editor to add, modify, or delete GroupWise service provider settings.

4 Click OK to save the changes.

Configuring the LDAP Service Provider

The LDAP service provider is installed and configured when you install the WebAccess Application to a Web server. The LDAP service provider receives LDAP directory requests from the WebAccess Application and communicates with LDAP services to fill the requests.

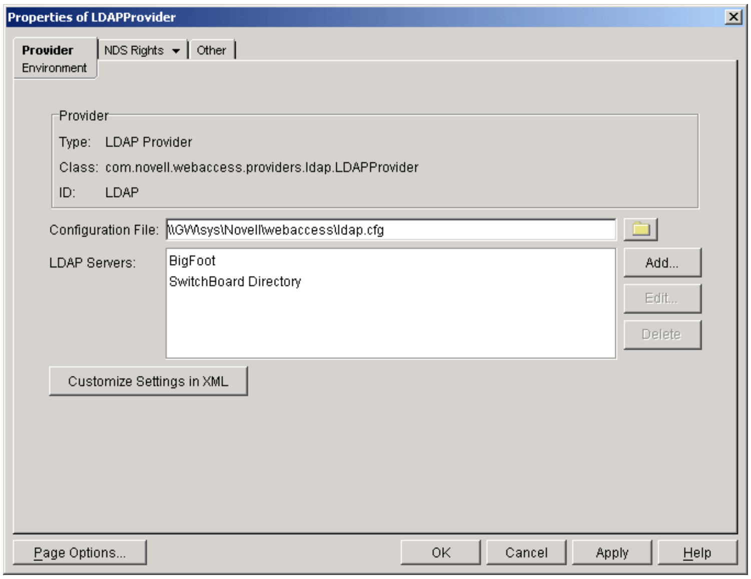
The GroupWise® WebAccess installation program creates a Novell® eDirectory™ object for the LDAP service provider in the same context as the WebAccess Application. The object is named LDAPProvider. Using ConsoleOne®, you can modify the LDAPProvider object to define the LDAP services that the service provider can contact.

To modify the LDAP service provider's configuration:

1 In ConsoleOne, right-click the LDAP service provider object (LDAPProvider), then click Properties.

NOTE: The LDAP service provider object is not available in the GroupWise View. To locate the LDAP service provider object, you must use the Console View.

2 If necessary, click Provider > Environment to display the Environment page.



Provider

Type: LDAP Provider

Class: com.novell.webaccess.providers.ldap.LDAPProvider

ID: LDAP

Configuration File: \\GW\sys\Novell\webaccess\ldap.cfg

LDAP Servers: BigFoot
SwitchBoard Directory

3 Choose from the following options:

Configuration File: The LDAP service provider's configuration file (ldap.cfg) contains the information for the LDAP services defined in the LDAP servers list. Because the LDAP service provider cannot access eDirectory or the GroupWise databases for this information, ConsoleOne writes the information to the ldap.cfg file.

By default, the ldap.cfg file is stored in the WebAccess Application's home directory (novell\webaccess on the Web server or /opt/novell/groupwise/webaccess on Linux). You should avoid changing the location of the file. If you do change the location of the file, you need to make sure to modify the ldap.cfg path in the Java servlet engine's properties file. If you do not, the LDAP service provider will continue to look for its configuration information in the old location.

LDAP Servers: This list displays the LDAP services the LDAP service provider can communicate with. The GroupWise WebAccess Address Book will list all LDAP services shown in the list.

To add an LDAP service to the list, click Add to display the Add LDAP Server dialog box, fill in the required information, then click OK. For information about each of the fields, click Help in the Add LDAP Server dialog box.

To edit an LDAP service's information, select the LDAP service in the list, then click Edit.

To remove an LDAP service from the list, select the LDAP service in the list, then click Delete.

Customize Settings in XML: Click this button to launch the XML editor. You can use the editor to add, modify, or delete LDAP service provider settings.

4 Click OK to save the changes.

Configuring the GroupWise Document Service Provider

The GroupWise® Document service provider is installed and configured when you install the WebPublisher Application to a Web server. The GroupWise Document service provider receives GroupWise document requests from the WebPublisher Application and communicates with the WebAccess Agent to fill the requests.

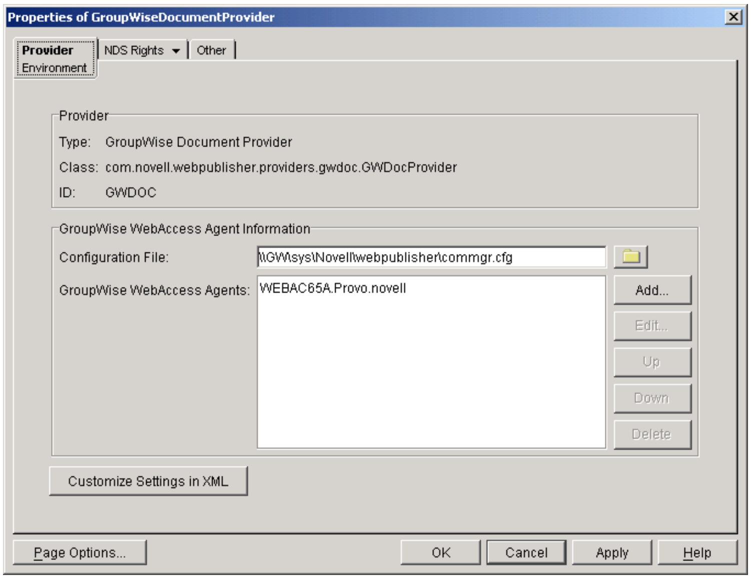
The WebAccess installation program creates a Novell® eDirectory™ object for the GroupWise Document service provider in the same context as the WebPublisher Application. The object is named GroupWiseDocumentProvider. Using ConsoleOne®, you can modify the GroupWiseDocumentProvider object to define the WebAccess Agents that the service provider will contact to fill GroupWise document requests. If your GroupWise system includes more than one WebAccess Agent, you can use this feature to provide failover support.

To modify the GroupWise Document service provider's configuration:

1 In ConsoleOne, right-click the GroupWise Document service provider object (GroupWiseDocumentProvider), then click Properties.

   NOTE: The GroupWise Document service provider object is not available in the GroupWise View. To locate the GroupWise Document service provider object, you must use the Console View.

2 If necessary, click Provider > Environment to display the Environment page.


Properties of GroupWiseDocumentProvider

Provider

Type: GroupWise Document Provider

Class: com.novell.webpublisher.providers.gwdoc.GWDocProvider

ID: GWDOC

GroupWise WebAccess Agent Information

Configuration File: \\GW\sys\Novell\webpublisher\commgr.cfg

GroupWise WebAccess Agents: WEBAC65A.Provo.novell

3 Choose from the following options:

   Configuration File: The WebAccess Agent's configuration file (commgr.cfg) contains the agent's IP address and the encryption key required by the GroupWise Document service provider to communicate with the WebAccess Agent. By default, the commgr.cfg file is stored in the WebPublisher Application's home directory (novell\webpublisher on the Web server or /opt/novell/groupwise/webpublisher on Linux).

   In general, you should not need to change this setting. However, if you have multiple WebAccess Agents in your GroupWise system and you are optimizing WebPublisher to provide greater scalability and availability, you might need to change the setting. For information, see "Configuring Redirection and Failover Support" on page 810.

   GroupWise WebAccess Agents: This list displays the WebAccess Agents the GroupWise Document service provider can communicate with when attempting to complete a request. If the first one listed is unavailable, the GroupWise Document service provider will attempt to use the second, third, fourth, and so on until it is successful. This provides failover support and ensures greater availability for your WebPublisher users. For more information about optimizing availability, see "Configuring Redirection and Failover Support" on page 810.

   The list must include at least one WebAccess Agent.

To add a WebAccess Agent to the list, click Add to browse for and select the WebAccess Agent object, then click OK.

To edit a WebAccess Agent's information, select the WebAccess Agent in the list, then click Edit.

To remove a WebAccess Agent from the list, select the WebAccess Agent in the list, then click Delete.

Customize Settings in XML: Click this button to launch the XML editor. You can use the editor to add, modify, or delete GroupWise Document service provider settings.

4 Click OK to save the changes.

60 Customizing the WebAccess Interface

GroupWise® WebAccess enables you to change the default Novell® logo and colors used in the WebAccess interface. For example, you can add your company logo to the main WebAccess window and change the colors to match your company colors.

You use the customization.properties file to change the logo and colors.

1 Open the customization.properties file with a text editor.

The file is located in the following directory:

NetWare and Windows: *tomcat_dir*\webapps\ROOT\WEB-INF\classes\com\novell\webaccess\templates
Linux: *tomcat_dir*/webapps/gw/WEB-INF/classes/com/novell/webaccess/templates

2 If you want to change the logo image:

2a Locate the CUSTOMIZABLE IMAGE FOR GROUPWISE WEBACCESS section at the beginning of the file.

2b To turn on customization for the logo image, set the WebAccess.Customize.Image.enable property to TRUE:

```
WebAccess.Customize.Image.enable=true
```

2c Modify the image properties as desired. The customization.properties file contains descriptions of each property.

3 If you want to change the WebAccess colors:

3a Locate the CUSTOMIZABLE COLORS SCHEME FOR GROUPWISE WEBACCESS section in the file.

3b To turn on customization of the colors, set the WebAccess.Customize.Color.enable setting to TRUE:

```
WebAccess.Customize.Color.enable=true
```

3c Modify the color properties as desired. The customization.properties file contains descriptions of each property.

4 Save the customization.properties file.

5 Restart the Web server.

6 In a Web browser, clear the browser cache, then log in to GroupWise WebAccess.

61 Monitoring WebAccess Operations

The WebAccess Agent can be monitored at the server where it runs and also in your Web browser. The WebAccess Application can be monitored in your Web browser.

- "Monitoring the WebAccess Agent" on page 875
- "Monitoring the WebAccess Application" on page 882

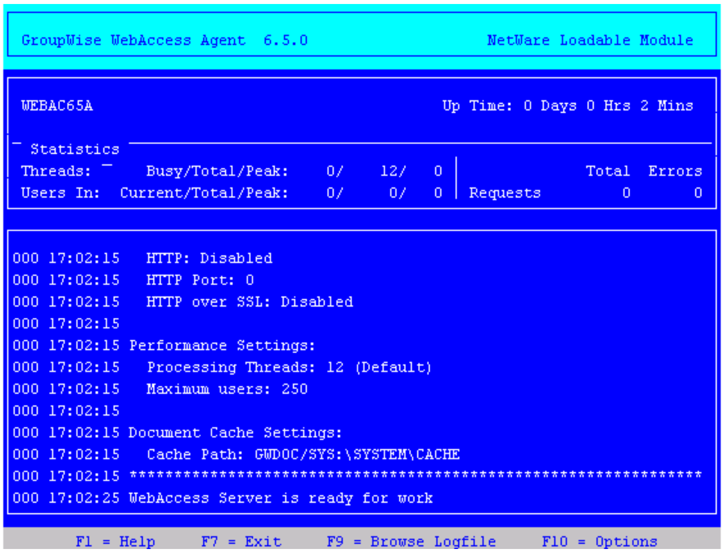
Monitoring the WebAccess Agent

The following sections explain the various methods you can use to monitor the GroupWise® WebAccess Agent to ensure that it is operating properly.

- "Monitoring the NetWare WebAccess Agent" on page 875
- "Monitoring the Windows WebAccess Agent" on page 877
- "Monitoring the Linux WebAccess Agent" on page 878
- "Monitoring the WebAccess Agent through the Web Console" on page 879
- "Monitoring the WebAccess Agent through NetWare 6.5 Remote Manager" on page 881

Monitoring the NetWare WebAccess Agent

The NetWare® WebAccess Agent console, shown below, lets you monitor the operation of the agent, view the agent's log information, and change the log settings while at the server.



```
GroupWise WebAccess Agent  6.5.0                     NetWare Loadable Module

WEBAC65A                                         Up Time: 0 Days 0 Hrs 2 Mins

 Statistics
Threads:       Busy/Total/Peak:    0/    12/    0 |               Total  Errors
Users In:   Current/Total/Peak:    0/     0/    0 | Requests          0       0

000 17:02:15   HTTP: Disabled
000 17:02:15   HTTP Port: 0
000 17:02:15   HTTP over SSL: Disabled
000 17:02:15
000 17:02:15 Performance Settings:
000 17:02:15   Processing Threads: 12 (Default)
000 17:02:15   Maximum users: 250
000 17:02:15
000 17:02:15 Document Cache Settings:
000 17:02:15   Cache Path: GWDOC/SYS:\SYSTEM\CACHE
000 17:02:15 *****************************************************************
000 17:02:25 WebAccess Server is ready for work

        F1 = Help      F7 = Exit      F9 = Browse Logfile      F10 = Options
```

The console and its options are described below.

Up Time

The Up Time field displays how long it has been since the WebAccess Agent was started.

Threads

The default of 12 threads enables the WebAccess Agent to service 12 user requests at one time. The Busy field displays the number of threads that are currently servicing user requests. The Total field displays the total number of threads available to service requests (by default, 12). The Peak field displays the most threads used at one time to service requests. If all threads are busy much of the time, you can increase the number of threads available for use. See "Modifying WebAccess Settings" on page 829.

Users In

The Users In field displays the number of users who currently are logged in. During startup, if you have enabled WebPublisher, the WebAccess Agent logs in one time for each available thread; these logins are reflected in the Users In fields. The Total field displays the total number of users who have logged in during the current up time. The Peak field displays the most users who have been logged in at one time.

By default, a maximum of 250 users can be logged in at one time. You can use the /maxusers startup switch to change the default. See "Using WebAccess Agent Startup Switches" on page 895.

Requests

The Total field displays the total number of requests the WebAccess Agent has processed during its current up time. The Errors field lists the number of requests that could not be processed because of errors.

Logging Box

The Logging box displays the logged information. The current log level determines the amount of information that is displayed (see "F10 = Options" on page 876). For each line, the first item is the number of the thread that processed the user's request, the second item is the time of the request, and the third item is the information associated with the request.

F7 = Exit

Press F7 to shut down the WebAccess Agent.

F9 = Browse Logfile

Press F9 to view the log file. If disk logging is turned on, the current log file is displayed. If disk logging is turned off, a list of old log files is displayed (if any exist). You can then choose which log file you want to view.

F10 = Options

Press F10, then select View Log Files or Logging Options. Using the logging options, you can specify the logging level, turn disk logging on or off, specify the number of days to keep old log files, and specify the maximum amount of disk space to use for log files.

Any changes you make to the logging options apply only to the current session. When you restart the WebAccess Agent, the logging level is reset to the level specified in ConsoleOne® or in the startup file (strtweb.ncf).

Log Level: Off turns logging off; Normal displays initial statistics, user logins, warnings, and errors; Verbose displays Normal logging plus user requests; and Diagnostic displays Verbose logging plus thread information. The default is Normal logging. Use Diagnostic only if you are troubleshooting a problem with WebAccess.

File Logging: Turns disk logging on or off. When disk logging is turned on, the WebAccess Agent creates a new log file each day and each time it is restarted. The log file is named *mmdd*web.*nnn*, where *mm* is the month, *dd* is the day, and *nnn* is a sequenced number (001 for the first log file of the day, 002 for the second, and so forth). The default location for the log files is the *domain*\wpgate\webac65a\*xxx*.prc directory.

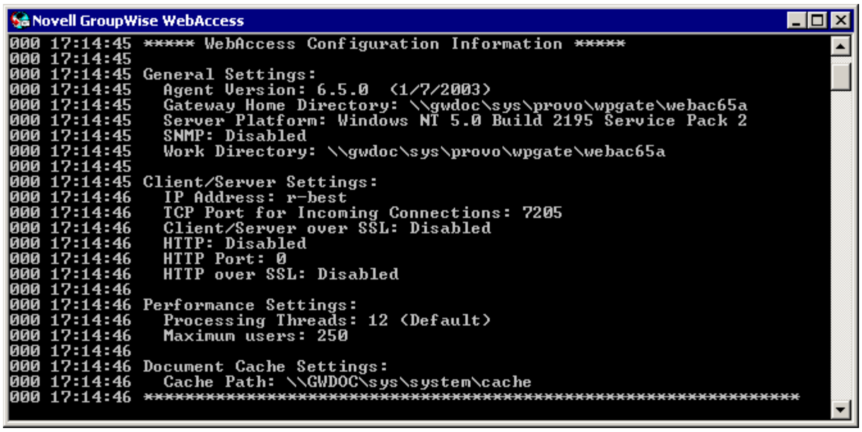
The verbose and diagnostic logging levels do not degrade WebAccess Agent performance, but log files saved to disk consume more disk space when verbose or diagnostic logging is in use.

Max Log File Age: Specifies the number of days you want the WebAccess Agent to retain old log files. The WebAccess Agent will retain the log file for the specified number of days unless the maximum disk space for the log files is exceeded. The default age is 7 days.

Max Log Disk Space: Specifies the maximum amount of disk space you want to use for log files. If the disk space limit is exceeded, the WebAccess Agent will delete log files, beginning with the oldest file, until the limit is no longer exceeded. The default disk space is 65536 KB.

Monitoring the Windows WebAccess Agent

The Windows WebAccess Agent console lets you monitor the operation of the agent. The console, shown below, is displayed in a DOS window.



```
000 17:14:45 ***** WebAccess Configuration Information *****
000 17:14:45
000 17:14:45 General Settings:
000 17:14:45   Agent Version: 6.5.0  (1/7/2003)
000 17:14:45   Gateway Home Directory: \\gwdoc\sys\provo\wpgate\webac65a
000 17:14:45   Server Platform: Windows NT 5.0 Build 2195 Service Pack 2
000 17:14:45   SNMP: Disabled
000 17:14:45   Work Directory: \\gwdoc\sys\provo\wpgate\webac65a
000 17:14:45
000 17:14:45 Client/Server Settings:
000 17:14:46   IP Address: r-best
000 17:14:46   TCP Port for Incoming Connections: 7205
000 17:14:46   Client/Server over SSL: Disabled
000 17:14:46   HTTP: Disabled
000 17:14:46   HTTP Port: 0
000 17:14:46   HTTP over SSL: Disabled
000 17:14:46
000 17:14:46 Performance Settings:
000 17:14:46   Processing Threads: 12 (Default)
000 17:14:46   Maximum users: 250
000 17:14:46
000 17:14:46 Document Cache Settings:
000 17:14:46   Cache Path: \\GWDOC\sys\system\cache
000 17:14:46 ****************************************************************
```

The console and its options are described below.

Logging Window

The current logging level determines the amount of information that is displayed. You can specify the logging level through ConsoleOne, through startup switches, or by using the F2 function key. See "Modifying Log Settings in ConsoleOne" on page 833, "Modifying Log Settings through Startup Switches" on page 834, and "F2" on page 878.

The verbose and diagnostic logging levels do not degrade WebAccess Agent performance, but log files saved to disk consume more disk space when verbose or diagnostic logging is in use.

For each line, the first item is the number of the thread that processed the user's request, the second item is the time of the request, and the third item is the information associated with the request.

F1 or F7

Shuts down and exits the agent.

F2

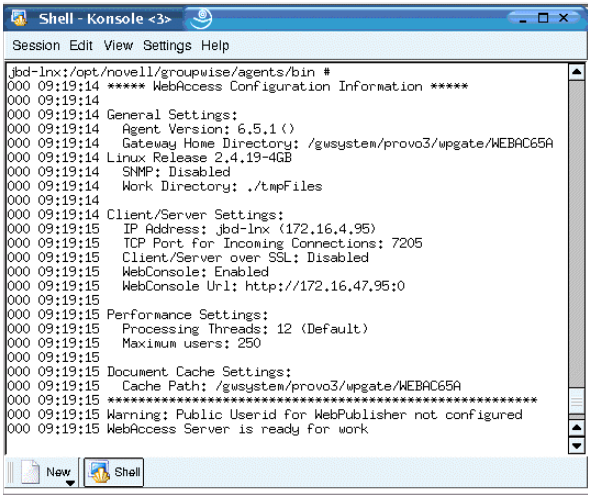
Cycles the logging level between Normal, Verbose, and Diagnostic. Normal displays initial statistics, user logins, warnings, and errors; Verbose displays Normal logging plus user requests; and Diagnostic displays Verbose logging plus thread information. The default is Normal logging. Use Verbose only if you are troubleshooting a problem with WebAccess.

The verbose and diagnostic logging levels do not degrade WebAccess Agent performance, but log files saved to disk consume more disk space when verbose or diagnostic logging is in use.

Any changes you make to the logging level using F2 apply only to the current session. When you restart the WebAccess Agent, the logging level is reset to the level specified in ConsoleOne or in the startup file (strtweb.bat).

Monitoring the Linux WebAccess Agent

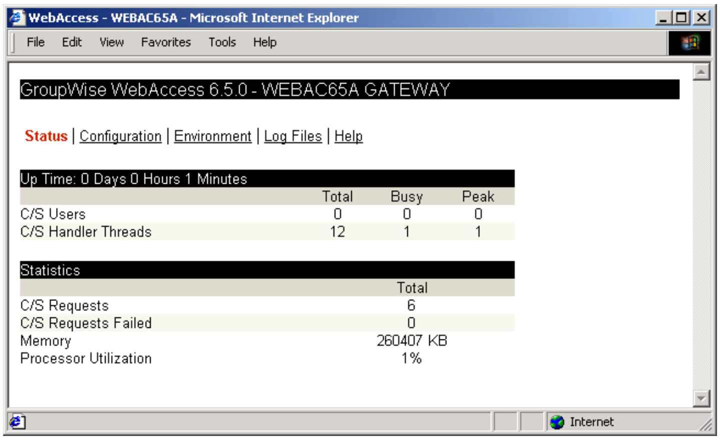
By default, the Linux Agent runs as a daemon with no user interface. To display information on the server where the WebAccess Agent runs, you must start the WebAccess Agent with the --show startup switch. The console is displayed in a terminal window.



```
jbd-lnx:/opt/novell/groupwise/agents/bin #
000 09:19:14 ***** WebAccess Configuration Information *****
000 09:19:14
000 09:19:14 General Settings:
000 09:19:14   Agent Version: 6.5.1 ()
000 09:19:14   Gateway Home Directory: /gwsystem/provo3/wpgate/WEBAC65A
000 09:19:14 Linux Release 2.4.19-4GB
000 09:19:14   SNMP: Disabled
000 09:19:14   Work Directory: ./tmpFiles
000 09:19:14
000 09:19:14 Client/Server Settings:
000 09:19:15   IP Address: jbd-lnx (172.16.4.95)
000 09:19:15   TCP Port for Incoming Connections: 7205
000 09:19:15   Client/Server over SSL: Disabled
000 09:19:15   WebConsole: Enabled
000 09:19:15   WebConsole Url: http://172.16.47.95:0
000 09:19:15
000 09:19:15 Performance Settings:
000 09:19:15   Processing Threads: 12 (Default)
000 09:19:15   Maximum users: 250
000 09:19:15
000 09:19:15 Document Cache Settings:
000 09:19:15   Cache Path: /gwsystem/provo3/wpgate/WEBAC65A
000 09:19:15 ********************************************************
000 09:19:15 Warning: Public Userid for WebPublisher not configured
000 09:19:15 WebAccess Server is ready for work
```

Monitoring the WebAccess Agent through the Web Console

You can use a Web browser interface, referred to as the Web console, to monitor the WebAccess Agent.



GroupWise WebAccess 6.5.0 - WEBAC65A GATEWAY

Status | Configuration | Environment | Log Files | Help

Up Time: 0 Days 0 Hours 1 Minutes			
	Total	Busy	Peak
C/S Users	0	0	0
C/S Handler Threads	12	1	1

Statistics	
	Total
C/S Requests	6
C/S Requests Failed	0
Memory	260407 KB
Processor Utilization	1%

Through the Web console you can view the following information:

- **Status:** Displays how long the WebAccess Agent has been up; the number of client/server users who have logged in, the number of threads dedicated to handling requests, and the number of successful and failed requests; and the amount of memory on the server and the percent of processor utilization.
- **Configuration:** Displays the gateway home directory being used by the WebAccess Agent, the current log settings, the performance settings (processing threads and maximum users), and the client/server settings (IP address, TCP port, and so forth).
- **Environment:** Displays server information such as name, operating system date, memory, processor utilization, and loaded modules.
- **Log Files:** Lets you view the contents of the WebAccess Agent's log files and the current log settings.

For detailed information about each field on the Status, Configuration, Environment, or Log Files page, select the page, then click Help.

You cannot use the Web console to change any of the WebAccess Agent's settings. Changes must be made through ConsoleOne, the WebAccess Agent console, or the startup file.

Refer to the following sections for information about enabling and using the Web console:

- "Enabling the WebAccess Agent Web Console" on page 879
- "Using the WebAccess Agent Web Console" on page 881

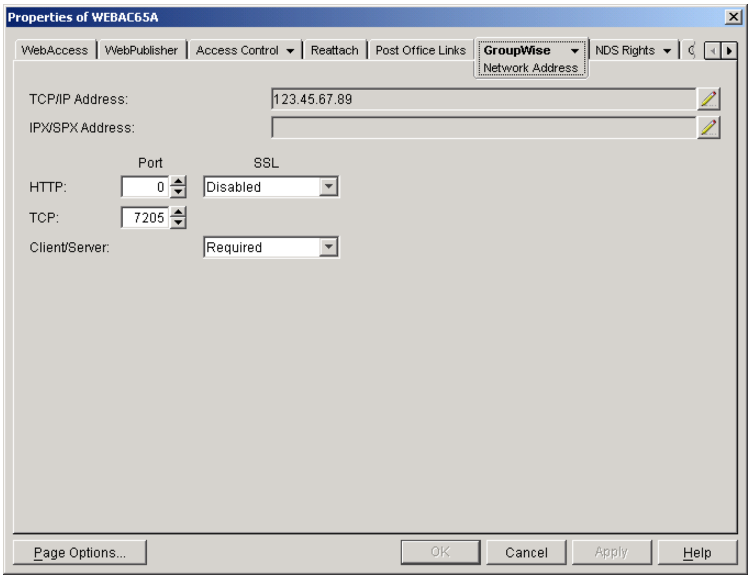
Enabling the WebAccess Agent Web Console

If, during, installation, you enabled the Web console, skip to Using the WebAccess Agent Web Console. On Linux, the Web Console is enabled by default. Skip to Using the WebAccess Agent Web Console.

If you want to enable the Web console, you need to complete the following steps.

1 In ConsoleOne, right-click the WebAccess Agent object, then click Properties.

2 Click GroupWise > Network Address to display the Network Address page.



Properties of WEBAC65A

TCP/IP Address: 123.45.67.89

IPX/SPX Address:

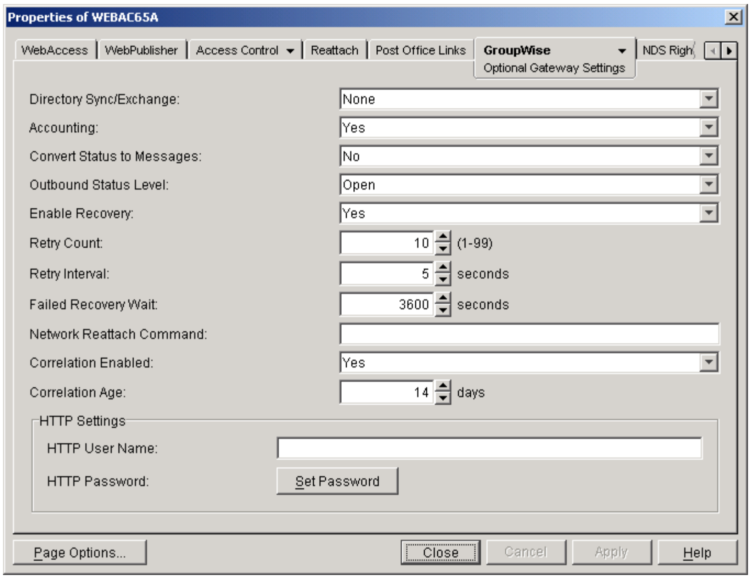
	Port	SSL
HTTP:	0	Disabled
TCP:	7205	
Client/Server:		Required

3 In the HTTP Port field, enter a port number. We recommend that you use port 7211 if it is not already in use on the WebAccess Agent's server.

Assigning a port number enables the Web console; assigning 0 as the port number disables the Web console.

Any user who knows the WebAccess Agent's IP address (or hostname) and the HTTP port number will be able to use the Web console. If you want to restrict Web console access, you can assign a username and password. To do so:

4 Click the GroupWise tab, then click Optional Gateway Settings to display the Optional Gateway Settings page.



Properties of WEBAC65A

Directory Sync/Exchange: None

Accounting: Yes

Convert Status to Messages: No

Outbound Status Level: Open

Enable Recovery: Yes

Retry Count: 10 (1-99)

Retry Interval: 5 seconds

Failed Recovery Wait: 3600 seconds

Network Reattach Command:

Correlation Enabled: Yes

Correlation Age: 14 days

HTTP Settings

HTTP User Name:

HTTP Password:

5 In the HTTP User Name field, enter an arbitrary username (for example, webcon).

6 Click Set Password to assign a password (for example, monitor).

7 Click OK to save your changes.

The Web console can only be enabled or disabled through the Network Address page (see Step 2 above). However, you can use the /httpuser and /httppassword startup switches to override the assigned username and password. For more information, see Appendix 64, "Using WebAccess Agent Startup Switches," on page 895.

Using the WebAccess Agent Web Console

1 In a Web browser, enter the following:

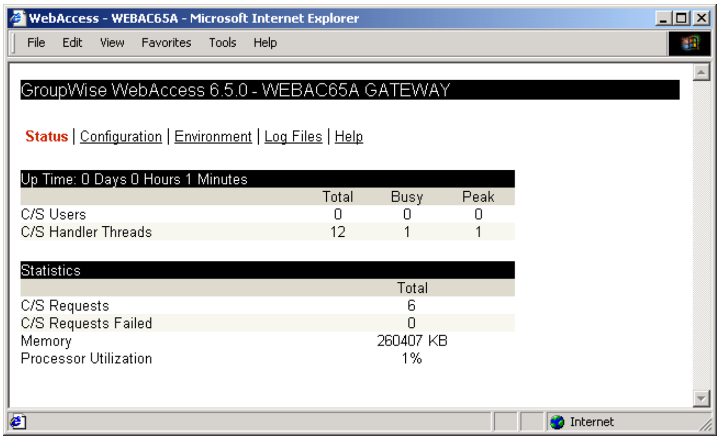
`http://`*`IP_address`*`:`*`agent_port`*

or

`https://`*`IP_address`*`:`*`agent_port`*

where *IP_address* is the IP address of the server where the WebAccess Agent is running, and *agent_port* is the port number assigned to the agent. If you used the default port during installation, the port number is 7211.

2 If prompted, enter the Web console username and password.



GroupWise WebAccess 6.5.0 - WEBAC65A GATEWAY

Up Time: 0 Days 0 Hours 1 Minutes	Total	Busy	Peak
C/S Users	0	0	0
C/S Handler Threads	12	1	1

Statistics	Total
C/S Requests	6
C/S Requests Failed	0
Memory	260407 KB
Processor Utilization	1%

3 Select Status, Configuration, Environment, or Log Files to view the desired information.

For detailed information about each field on the Status, Configuration, Environment, or Log Files page, select the page, then click Help.

Monitoring the WebAccess Agent through NetWare 6.5 Remote Manager

If the WebAccess Agent is running on a NetWare 6.5 server, you can use the IP Address Management feature in NetWare Remote Manager (NetWare Remote Manager > Manage Server > IP Address Management) to view the IP address and port configuration for the WebAccess Agent. This is also true for other GroupWise agents (MTA, POA, and Internet Agent) running on NetWare 6.5 servers.

IMPORTANT: If the WebAccess Agent is running in protected mode, it will not display in NetWare Remote Manager.

You access NetWare Remote Manager by entering the following URL in a Web browser:

`http://`*`server_address`*`:8008`

For example:

```
http://137.65.123.11:8008
```

For more information about using NetWare Remote Manager, see the NetWare 6.5 documentation (http://www.novell.com/documentation/nw65).

Monitoring the WebAccess Application

The WebAccess Application includes a Web console, similar to the WebAccess Agent's Web console, that you can use to monitor it. The Web console lets you see information about logged in users, such as their IP address, their GroupWise and Web browser versions, and the WebAccess Agent providing mailbox access. In addition, you can view the WebAccess Application's log files and configuration files, and view Java information such as the version and classpath settings.

The following sections provide information to help you use the Web console:

- "Enabling the WebAccess Application Web Console" on page 882
- "Using the WebAccess Application Web Console" on page 882
- "Understanding the WebAccess Application Web Console Information" on page 883

Enabling the WebAccess Application Web Console

1. Open the webacc.cfg file, located in the WebAccess Application's home directory (novell\webaccess on the Web server or /opt/novell/groupwise/webaccess on Linux).
2. Locate the following lines in the file:

   ```
   Admin.WebConsole.enable=false
   Admin.WebConsole.username=admin
   Admin.WebConsole.password=admin
   ```

3. Enable the Web console by changing the FALSE entry to TRUE:

   ```
   Admin.WebConsole.enable=true
   ```

4. If desired, change the default username and password. A username and password is required.
5. Save the file.
6. Restart the Java servlet engine.

Using the WebAccess Application Web Console

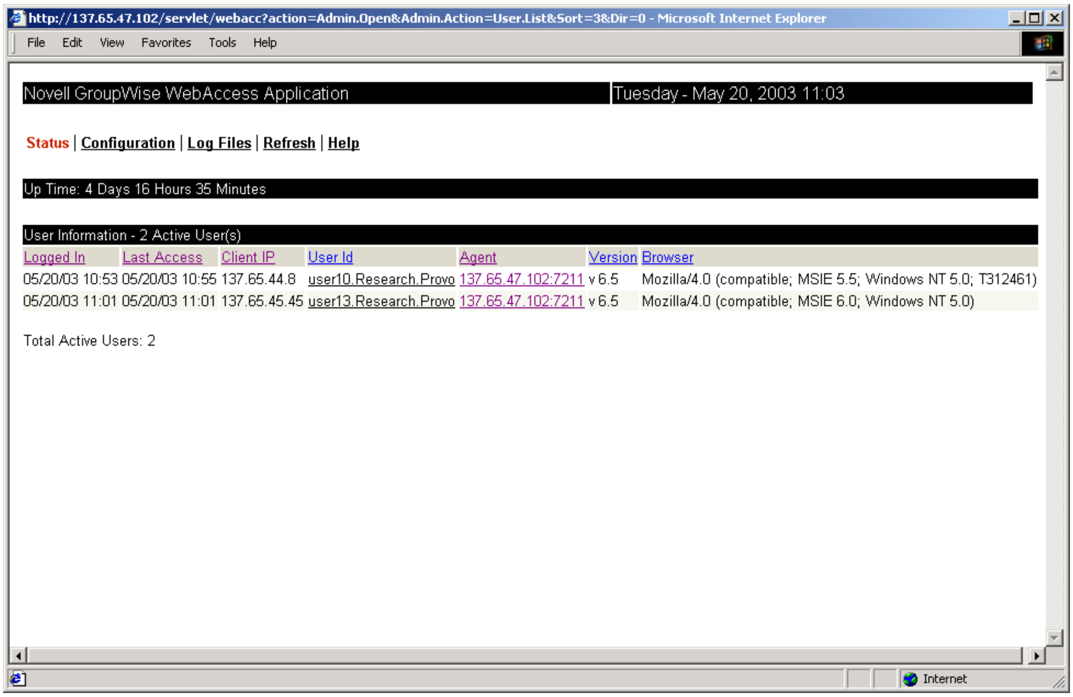
1. In a Web browser, enter the following URL:

   NetWare or Windows: http://*server_address*/servlet/webacc?action=Admin.Open
   Linux: http://*server_address*/gw/webacc?action=Admin.Open

   where *server_address* is the Web server's IP address or DNS hostname.

2. When prompted, enter the username and password.

The Web console is displayed.



http://137.65.47.102/servlet/webacc?action=Admin.Open&Admin.Action=User.List&Sort=3&Dir=0 - Microsoft Internet Explorer

Novell GroupWise WebAccess Application

Tuesday - May 20, 2003 11:03

Status | Configuration | Log Files | Refresh | Help

Up Time: 4 Days 16 Hours 35 Minutes

User Information - 2 Active User(s)

Logged In	Last Access	Client IP	User Id	Agent	Version	Browser
05/20/03 10:53	05/20/03 10:55	137.65.44.8	user10.Research.Provo	137.65.47.102:7211	v 6.5	Mozilla/4.0 (compatible; MSIE 5.5; Windows NT 5.0; T312461)
05/20/03 11:01	05/20/03 11:01	137.65.45.45	user13.Research.Provo	137.65.47.102:7211	v 6.5	Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.0)

Total Active Users: 2

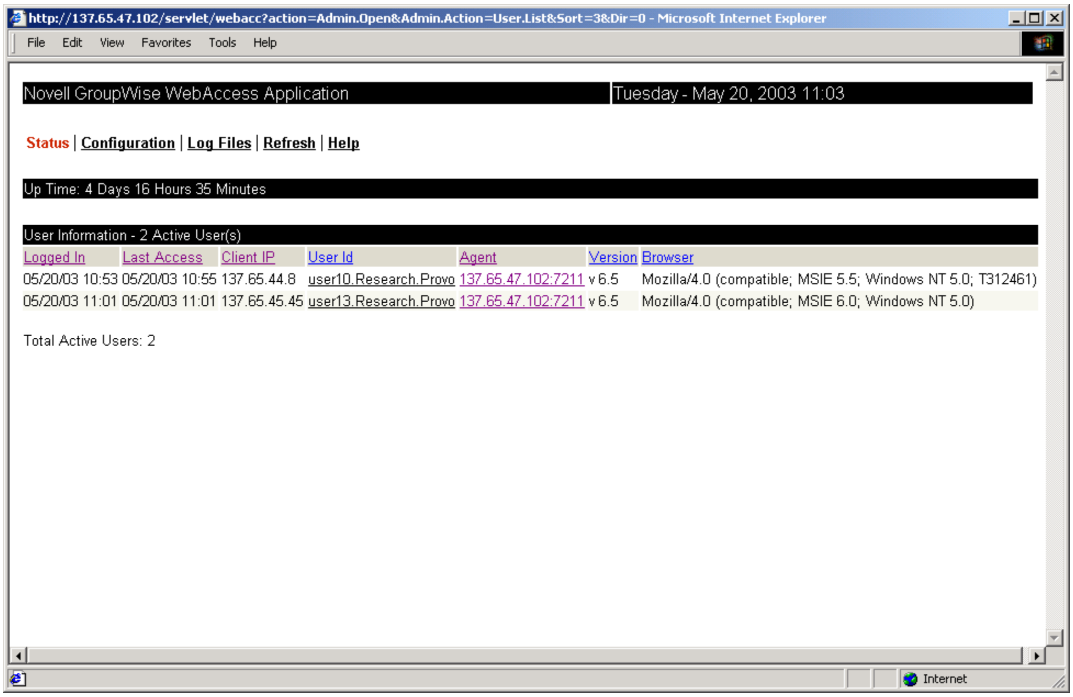
Understanding the WebAccess Application Web Console Information

The Web console information is organized into three main pages:

- "Status" on page 883
- "Configuration" on page 885
- "Log Files" on page 887

Status

The Status page, shown below, is the initial page that is displayed when you log in to the Web console. It provides information about the users who are currently logged in to GroupWise WebAccess.


http://137.65.47.102/servlet/webacc?action=Admin.Open&Admin.Action=User.List&Sort=3&Dir=0 - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Internet

Novell GroupWise WebAccess Application | Tuesday - May 20, 2003 11:03

Status | Configuration | Log Files | Refresh | Help

Up Time: 4 Days 16 Hours 35 Minutes

User Information - 2 Active User(s)

Logged In	Last Access	Client IP	User Id	Agent	Version	Browser
05/20/03 10:53	05/20/03 10:55	137.65.44.8	user10.Research.Provo	137.65.47.102:7211	v 6.5	Mozilla/4.0 (compatible; MSIE 5.5; Windows NT 5.0; T312461)
05/20/03 11:01	05/20/03 11:01	137.65.45.45	user13.Research.Provo	137.65.47.102:7211	v 6.5	Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.0)

Total Active Users: 2

Refresh: Click the Refresh option to refresh the status information.

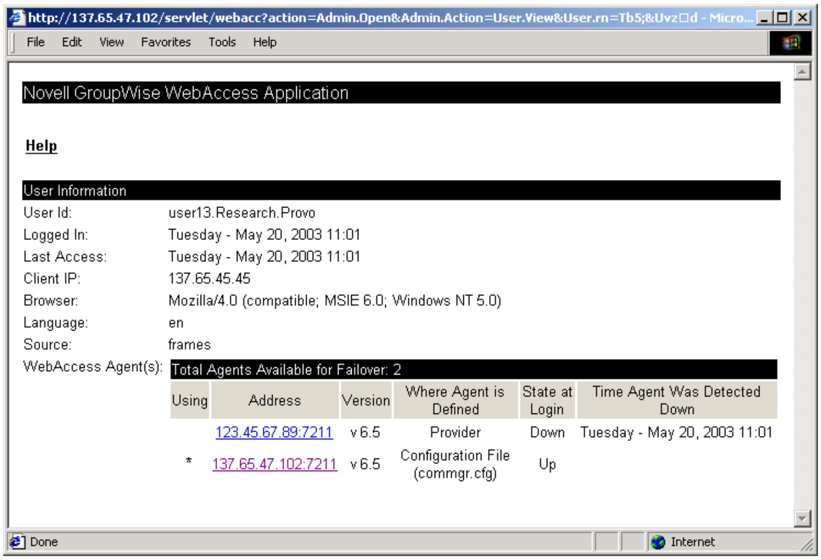
Up Time: Displays the number of days, hours, and minutes since the WebAccess Application started.

User Information: Displays information for the users currently logged in to GroupWise WebAccess. Each column is described below. Click a column heading to sort on that column.

- Logged In: Displays the date and time the user logged in.
- Last Access: Displays the date and time the user last performed a WebAccess operation that generated a request for the WebAccess Application.
- Client IP: Displays the IP address for the user’s session. If you are using a proxy server, the proxy server’s IP address is displayed.
- User ID: Displays the user’s name, post office, and domain (userID.po.domain). You can click a user’s ID to display expanded information about the user. This information is described in Expanded User Information below.
- Agent: Displays the IP address of the WebAccess Agent that is providing the user’s mailbox access. You can click the agent’s address to log into the agent’s Web console.
- Version: Displays the version of the WebAccess Agent that is providing the user’s mailbox access.
- Browser: Displays the user’s Web browser version.

Expanded User Information

When you click a user’s address in the User ID column, the following expanded User Information page is displayed.



http://137.65.47.102/servlet/webacc?action=Admin.Open&Admin.Action=User.View&User.rn=Tb5;&Uvz□d - Micro...
File
Edit
View
Favorites
Tools
Help
Done
Internet

Novell GroupWise WebAccess Application

Help

User Information

User Id:	user13.Research.Provo
Logged In:	Tuesday - May 20, 2003 11:01
Last Access:	Tuesday - May 20, 2003 11:01
Client IP:	137.65.45.45
Browser:	Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.0)
Language:	en
Source:	frames
WebAccess Agent(s):	Total Agents Available for Failover: 2

Using	Address	Version	Where Agent is Defined	State at Login	Time Agent Was Detected Down
	123.45.67.89:7211	v6.5	Provider	Down	Tuesday - May 20, 2003 11:01
*	137.65.47.102:7211	v6.5	Configuration File (commgr.cfg)	Up	

The User ID, Logged In, Last Access, Client IP, and Browser fields contain the same information that is displayed on the Status page. The following fields contain additional information not provided on the Status page.

Language: Displays the WebAccess language used.

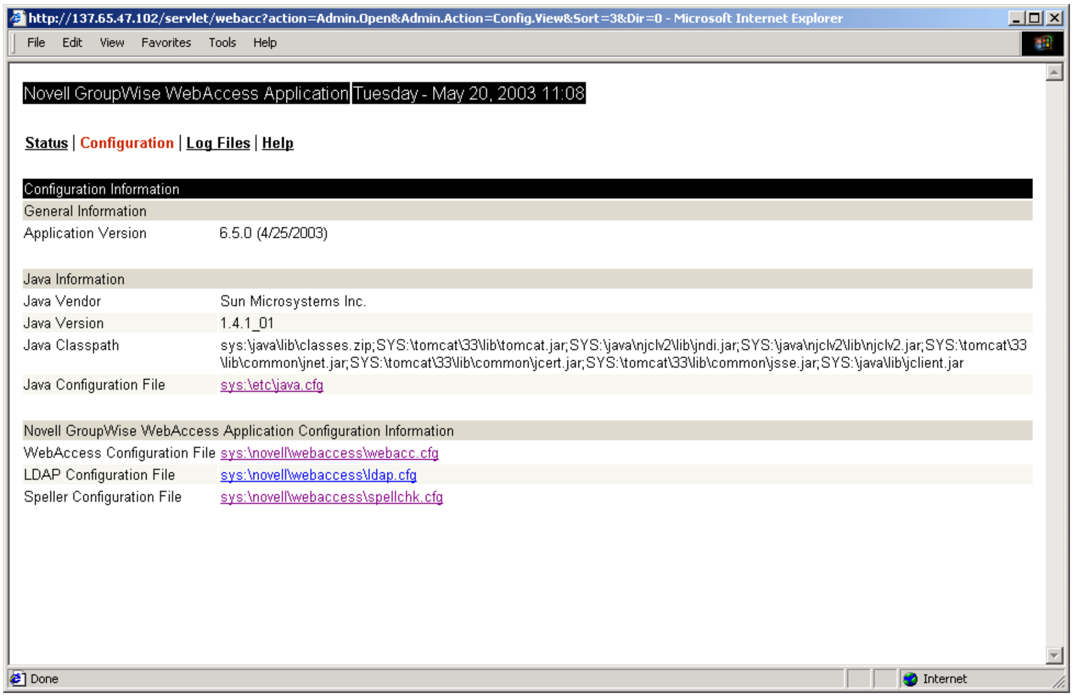
Source: Displays the WebAccess templates used.

WebAccess Agents: Displays the WebAccess Agents available to the user and each agent's status.

- Using: An asterisk (*) indicates that the WebAccess Agent is being used to provide access to the user's mailbox.
- Address: Displays the WebAccess Agent's address. You can click the agent's address to log into the agent's Web console.
- Version: Displays the WebAccess Agent's version.
- Where Agent Is Defined: Displays the location where the agent is defined. There are four possible locations: the user's post office, the user's domain, the GroupWiseWebAccess Provider, and the WebAccess commgr.cfg. For more information about where agents are defined and the order in which they are used, see "Configuring Redirection and Failover Support" on page 810.
- State at Login: Displays the state (UP or DOWN) that the WebAccess Agent was in when the user logged in to GroupWise WebAccess.
- Time Agent Was Detected Down: If the WebAccess Agent's state is DOWN, displays the time that the DOWN state was detected.

Configuration

The Configuration page, shown below, displays the WebAccess Application's version and Java information and lets you view the WebAccess Application configuration files.


http://137.65.47.102/servlet/webacc?action=Admin.Open&Admin.Action=Config.View&Sort=3&Dir=0 - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Done
Internet

Novell GroupWise WebAccess Application Tuesday - May 20, 2003 11:08

Status | Configuration | Log Files | Help

Configuration Information	
General Information	
Application Version	6.5.0 (4/25/2003)
Java Information	
Java Vendor	Sun Microsystems Inc.
Java Version	1.4.1_01
Java Classpath	sys:\java\lib\classes.zip;SYS:\tomcat\33\lib\tomcat.jar;SYS:\java\njclv2\lib\jndi.jar;SYS:\java\njclv2\lib\njclv2.jar;SYS:\tomcat\33\lib\common\jnet.jar;SYS:\tomcat\33\lib\common\jcert.jar;SYS:\tomcat\33\lib\common\jsse.jar;SYS:\java\lib\jclient.jar
Java Configuration File	sys:\etc\java.cfg
Novell GroupWise WebAccess Application Configuration Information	
WebAccess Configuration File	sys:\novell\webaccess\webacc.cfg
LDAP Configuration File	sys:\novell\webaccess\ldap.cfg
Speller Configuration File	sys:\novell\webaccess\spellchk.cfg

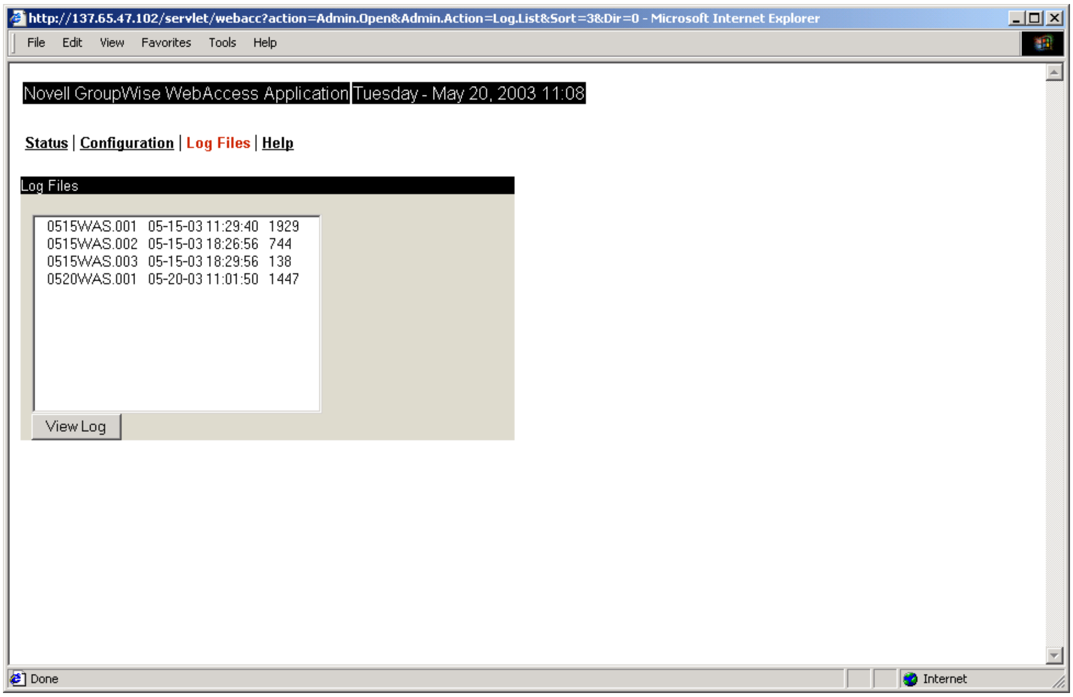
General Information: Displays the WebAccess Application's version number and date.

Java Information: Displays the Java vendor, version, and classpath information. You can click the link in the Java Configuration File field to open the configuration file for viewing. This is a view only option; you cannot make changes to the file.

Novell GroupWise WebAccess Application Configuration Information: Displays the configuration files used by the WebAccess Application. You can click the link for a file to open it for viewing. This is a view only option; you cannot make changes to the file.

Log Files

The Log Files page, shown below, lists the WebAccess Application's log files. To view a log file, select the file in the list, then click View Log.



http://137.65.47.102/servlet/webacc?action=Admin.Open&Admin.Action=Log.List&Sort=3&Dir=0 - Microsoft Internet Explorer

Novell GroupWise WebAccess Application Tuesday - May 20, 2003 11:08

Status | Configuration | Log Files | Help

Log Files

```
0515WAS.001  05-15-03 11:29:40  1929
0515WAS.002  05-15-03 18:26:56  744
0515WAS.003  05-15-03 18:29:56  138
0520WAS.001  05-20-03 11:01:50  1447
```

62 Securing WebAccess Agent Connections Via SSL

The GroupWise® WebAccess Agent can use the SSL (Secure Socket Layer) protocol to enable secure connections to Post Office Agents (POAs) and the WebAccess Agent Web console. For it to do so, you must ensure that the WebAccess Agent has access to a server certificate file and that you've specified which connections types you want secured through SSL. The following sections provide instructions:

- "Defining the Certificate File" on page 889
- "Defining Which Connections Will Use SSL" on page 890

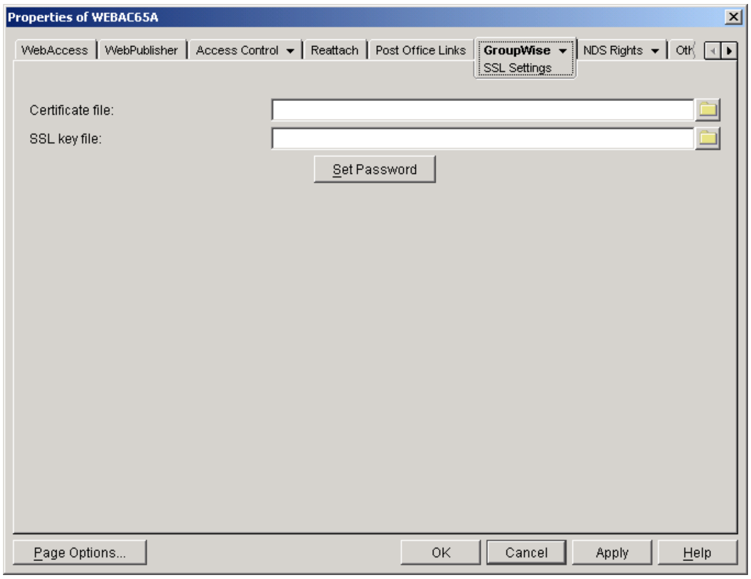
Defining the Certificate File

To use SSL, the WebAccess Agent requires access to a server certificate file and key file. The WebAccess Agent can use any Base64/PEM or PFX formatted certificate file located on its server. If the WebAccess Agent's server does not have a server certificate file, you can use the GroupWise Generate CSR utility to help you obtain one. For information, see "GroupWise Generate CSR Utility (GWCSRGEN)" on page 79.

To define the certificate file and key file that the WebAccess Agent will use:

1 In ConsoleOne®, right-click the WebAccess Agent object, then click Properties.

2 Click GroupWise > SSL Settings to display the SSL Settings page.



3 Fill in the Certificate File, SSL Key File, and Set Password fields:

Certificate File: Select the server certificate file that the WebAccess Agent will use. The certificate file must be in Base64/PEM or PFX format. If you type the filename rather than using the Browse button to select it, use the full path if the file is not in the same directory as the WebAccess Agent program.

SSL Key File: Select the key file associated with the certificate. If the private key is included in the certificate file rather than in a separate key file, leave this field blank. If you type the filename rather than using the Browse button to select it, use the full path if the file is not in the same directory as the WebAccess Agent program.

Set Password: Click Set Password to specify the password for the key. If the key does not require a password, do not use this option.

4 If you want to define which connections will use SSL, click Apply to save your changes, then continue with the next section, Defining Which Connections Will Use SSL.

or

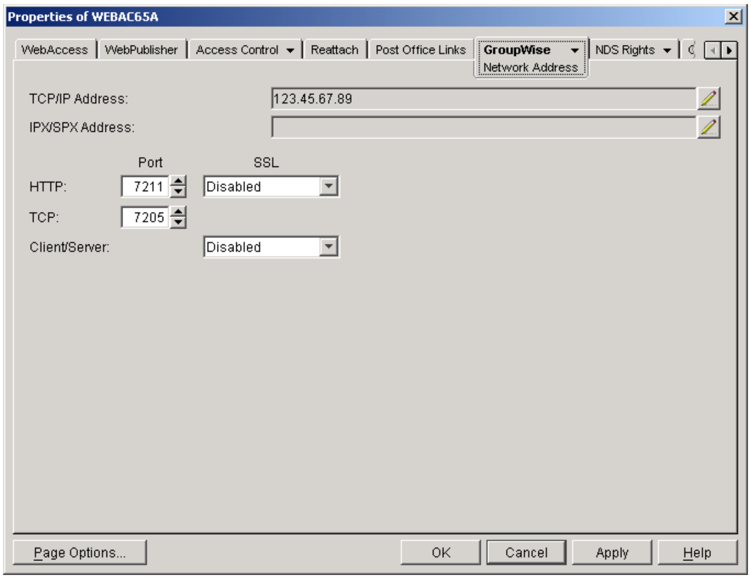
Click OK to save your changes.

Defining Which Connections Will Use SSL

After you've defined the WebAccess Agent's certificate and key file (see "Defining the Certificate File" on page 889), you can configure which connections you want to use SSL.

1 In ConsoleOne, if the WebAccess Agent object's property pages are not already displayed, right-click the WebAccess Agent object, then click Properties.

2 Click GroupWise > Network Address to display the Network Address page.



Properties of WEBAC65A

TCP/IP Address: 123.45.67.89

IPX/SPX Address:

	Port	SSL
HTTP:	7211	Disabled
TCP:	7205	
Client/Server:		Disabled

3 Configure the SSL settings for the following connections:

HTTP: Select Enabled to enable the WebAccess Agent to use a secure connection when passing information to the WebAccess Agent Web console. The Web browser must also be enabled to use SSL; if it is not, a non-secure connection will be used.

Client/Server: Select from the following options to configure the WebAccess Agent's use of secure connections to POAs:

- Disabled: The WebAccess Agent will not support SSL connections. All connections will be non-SSL.
- Enabled: The POA determines whether an SSL connection or non-SSL connection is used.
- Required: The WebAccess Agent will force SSL connections. Non-SSL connections will be denied.

63 Creating a PQA File for the WebAccess Client

You can use the GroupWise® WebAccess Installation program on Windows to create a Web Clipping Application (PQA), also referred to as a Palm Query Application, to enable Palm OS* device users to access their mailboxes through WebAccess.

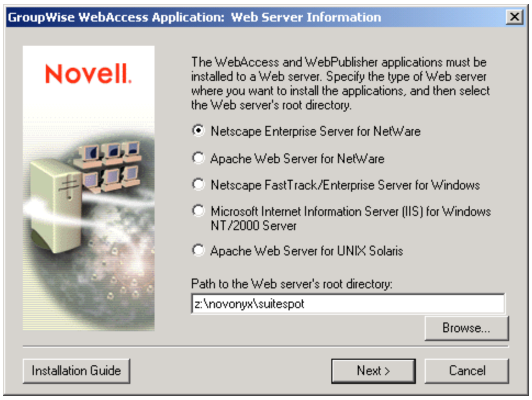
The Web Clipping Application, named groupwise.pqa, includes the URL required to connect to your GroupWise WebAccess installation, a Login page, an About Novell GroupWise page, and the images used when displaying GroupWise WebAccess on the Palm OS device.

To create a groupwise.pqa file:

1 If you've already created another groupwise.pqa file that you want to keep, make sure it is not in the Web server's *doc_root_directory*\com\novell\webacces\palm\en directory. The Installation program overwrites any groupwise.pqa file in the directory.

2 At a Windows workstation, run setup.exe /pqa from the \internet\webacces directory on the *GroupWise 6.5 Administrator* CD or the GroupWise software distribution directory

3 Select the language for the Installation program, then accept the License Agreement to display the following dialog box.



GroupWise WebAccess Application: Web Server Information

The WebAccess and WebPublisher applications must be installed to a Web server. Specify the type of Web server where you want to install the applications, and then select the Web server's root directory.

● Netscape Enterprise Server for NetWare

○ Apache Web Server for NetWare

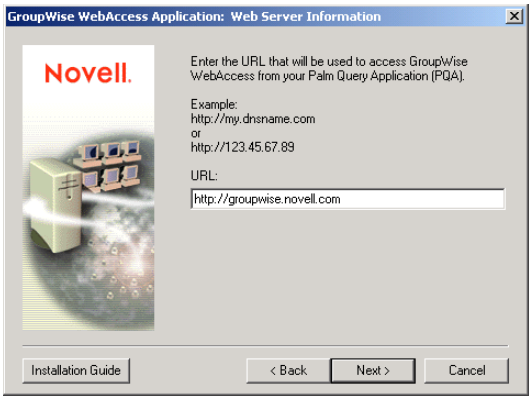
○ Netscape FastTrack/Enterprise Server for Windows

○ Microsoft Internet Information Server (IIS) for Windows NT/2000 Server

○ Apache Web Server for UNIX Solaris

Path to the Web server's root directory:
z:\novonyx\suitespot

4 Select the type of Web server where WebAccess is installed, make sure the path to the Web server's root directory is correct, then click Next.



Enter the URL that will be used to access GroupWise WebAccess from your Palm Query Application (PQA).

Example:
http://my.dnsname.com
or
http://123.45.67.89

URL: http://groupwise.novell.com

5 Specify the URL you want included in the .pqa file. For example:

```
http://groupwise.novell.com
```

The Installation program will automatically append /servlet/webacc to the URL so that users will be directed to the WebAccess login page. For example, using the URL above, the Installation program would create the following URL in the groupwise.pqa file:

```
http://groupwise.novell.com/servlet/webacc
```

As you determine the URL, keep in mind the following:

- If the Web server uses SSL, you should change http to https.
- If you are using a proxy server, you need to enter the proxy server's address.
- The Web clipping proxy server (gateway) does not currently support challenge and response authentication. Therefore, you need to ensure that the Web server is not configured to require basic challenge and response authentication, or at least is configured not to require this authentication for the URL defined in the groupwise.pqa file.

6 Click Next to create the .pqa file, then click Finish.

The groupwise.pqa file is created in the Web server's *doc_root_directory*\com\novell\webacces\palm\en directory. You can distribute it to your Palm OS device users just as you would any other .pqa file.

64 Using WebAccess Agent Startup Switches

You can use the switches listed below when starting the GroupWise® WebAccess Agent. The switches override any configuration settings you specified through ConsoleOne®.

To use a switch, you can:

- Add the switch to the command line. For example, load gwinter.nlm /ph-j:\domain\wpgate\webac65a.
- Include the switch in the strtweb.ncf or the strtweb.bat file. The strtweb.ncf file is located in the same directory as the NetWare® WebAccess Agent (typically sys:system) and the strtweb.bat file is located in the same directory as the Windows WebAccess Agent (by default, c:\webacc).
- Include the switch in a startup file and then reference the file when starting the WebAccess Agent. For example:

  ```
  load sys:system\gwinter @webac65a.waa
  ```

  During installation of the WebAccess Agent, the installation program creates a default startup file, *agent_name*.waa, where *agent_name* is the name assigned to the WebAccess Agent (for example, webac65a.waa). The startup file is referenced from the strtweb.ncf and strtweb.bat files and is created in the same directory as the WebAccess Agent.

The following switches are available:

NetWare WebAccess Agent	Linux WebAccess Agent	Windows WebAccess Agent	ConsoleOne Settings
@filename	@filename	@filename	N/A
/cluster	N/A	N/A	N/A
/help	--help	/help	N/A
/home	--home	/home	N/A
/http	--http	/http	GroupWise > Network Address
/httpuser	--httpuser	/httpuser	GroupWise > Optional Gateway Settings > HTTP User Name
/httppassword	--httppassword	/httppassword	GroupWise > Optional Gateway Settings > HTTP Password
/ip	--ip	/ip	GroupWise > Network Address
/log	--log	/log	GroupWise > Log Files > Log File Path
/logdays	--logdays	/logdays	GroupWise > Log Files > Max Log File Age

NetWare WebAccess Agent	Linux WebAccess Agent	Windows WebAccess Agent	ConsoleOne Settings
/logdiskon	--logdiskon	/logdiskon	N/A
/loglevel	--loglevel	/loglevel	GroupWise > Log Settings > Logging Level
/logmax	--logmax	/logmax	GroupWise > Log Settings > Max Log Disk Space
/maxusers	--maxusers	/maxusers	N/A
/password	N/A	N/A	N/A
/port	--port	/port	GroupWise > Network Address
N/A	--show	N/A	N/A
/threads	--threads	/threads	WebAccess > Settings > Maximum Threads
/user	N/A	N/A	N/A
/work	--work	/work	

@filename

Specifies a startup file to use. You can add any of the WebAccess Agent startup switches to the startup file and then reference the file when starting the WebAccess Agent. For example:

```
load sys:system\gwinter @webac65a.waa
```

During installation of the WebAccess Agent, the Installation program creates a default startup file, *agent_name*.waa, where *agent_name* is the name assigned to the WebAccess Agent (for example, webac65a.waa). On NetWare and Windows, the *agent_name*.waa file is created in the same directory as the WebAccess Agent program. On Linux, it is created in the /opt/novell/groupwise/agents/share directory

The startup file is referenced from the strtweb.ncf file on NetWare, the grpwise-wa script on Linux, and the strtweb.bat file on Windows, which enables you to run strtweb.ncf, grpwise-wa, or strtweb.bat to start the WebAccess Agent.

/cluster

Enables the WebAccess Agent to run in a clustered environment (using Novell® Cluster Services™).

For detailed information about running the WebAccess Agent in a clustered environment, see "Implementing WebAccess in a Novell Cluster" in "Novell Cluster Services" in the *GroupWise 6.5 Interoperability Guide*.

/help

Displays a listing and description of the startup switches.

/home (Required)

Specifies the path to the WebAccess Agent's gateway directory under the domain directory. If you use the default WebAccess Agent gateway directory name, the path is *x*:\domain\wpgate\webac65a. This switch is required.

/http

If the WebAccess Agent's Web console is disabled in ConsoleOne, this switch enables the Web console and assigns the port you specify. The default port is 7211.

/httpuser

Specifies the username that must be entered when logging in to the WebAccess Agent's Web console.

/httppassword

Specifies the password that must be entered when logging in to the WebAccess Agent's Web console.

/ip

Specifies the IP address of the WebAccess Agent's server.

/log

Specifies the path to the log file directory. On NetWare and Windows, the default log file directory is the *domain*\wpgate\webac65a\000.prc directory. On Linux, the default directory is /var/log/novell/groupwise/*domain.gateway*/000.prc.

Log files are named *mmdd.nnn*, where *mm* is the month, *dd* is the day, and *nnn* is a sequenced number starting with 001. For example, the first log file used on March 28 is named 0328.001, and the second log file used is named 0328.002.

For more information about the WebAccess Agent's logging, see "Controlling the Agent's Logging" on page 832.

/logdays

Specifies the maximum number of days to keep log files. This setting works in combination with the /logmax setting. Log files are deleted when the maximum number of days or disk space size is reached, whichever comes first. The default is 7 days.

For more information about the WebAccess Agent's logging, see "Controlling the Agent's Logging" on page 832.

/logdiskon

Turns disk logging on. By default, the log file is not written to disk on NetWare and Windows. On Linux, the log file is written to disk by default.

For more information about the WebAccess Agent’s logging, see “Controlling the Agent’s Logging” on page 832.

/loglevel

Specifies the level of information to write to the screen and to disk. There are three levels: Normal, Verbose, and Diagnostic. The default level is Normal. You can use Verbose to receive more information. You should use Diagnostic only if you are having problems with the WebAccess Agent. The verbose and diagnostic logging levels do not degrade Internet Agent performance, but log files saved to disk consume more disk space when verbose or diagnostic logging is in use. For more information about the logging levels, see “Controlling the Agent’s Logging” on page 832.

/logmax

Specifies the maximum disk space to use for logging. This setting works in combination with the /logdays setting. Log files are deleted when the maximum disk space or number of days is reached, whichever comes first. The default is 1024 KB.

For more information about the WebAccess Agent’s logging, see “Controlling the Agent’s Logging” on page 832.

/maxusers

Specifies the maximum number of users that the WebAccess Agent will allow to log in at one time. The default is 250.

/password (NetWare Only)

Used by the NetWare WebAccess Agent only. Specifies the Novell eDirectory™ password to use to access the network servers where the GroupWise domain directory and post office directories reside. Must be used with “/user (NetWare Only)” on page 899.

/port-*number*

Specifies the port number the WebAccess Agent listens to. The default is 7205.

--show

Used by the Linux WebAccess Agent only. Running the WebAccess Agent with this option disabled (the default) causes the WebAccess Agent to run as a daemon without a user interface. Enabling this option causes the logging UI to appear in a terminal window.

/threads-*number*

Specifies the number of threads the WebAccess Agent uses to process user requests. The default is 12, which means the WebAccess Agent can process 12 user requests at one time. For more information, see "Configuring the WebAccess Agent" on page 829.

/user (NetWare Only)

Used by the NetWare WebAccess Agent only. Specifies the eDirectory username to use to access the network servers where the GroupWise domain directory and post office directories reside. Must be used with /password.

/work

Specifies the path to the WebAccess Agent's work directory. By default, the work directory is the same as the WebAccess Agent's gateway directory (*x*:\domain\wpgate\webac65a).