Administration Guide

Netscape Application Server

Version 4.0

Recycled and Recyclable Paper

Contents

Preface

This chapter contains the following topics:

- Using the Documentation

- About This Guide

- How This Guide Is Organized

- Documentation Conventions

Using the Documentation

The following table lists the tasks and concepts that are described in the Netscape Application Server (NAS) and Netscape Application Builder (NAB) printed manuals and online read-me file. If you are trying to accomplish a specific task or learn more about a specific concept, refer to the appropriate manual.

Note that the printed manuals are also available as online files in PDF and HTML format.

For information about	See the following	Shipped with
Late-breaking information about the software and the documentation	`readme.htm`	NAS 4.0 Developer Edition (Solaris), NAS 4.0, NAB 4.0
Installing Netscape Application Server and its various components (Web Connector plug-in, Netscape Application Server Administrator), and configuring the sample applications	*Installation Guide*	NAS 4.0 Developer Edition (Solaris), NAS 4.0
Installing Netscape Application Builder.	`install.htm`	NAB 4.0

For information about	See the following	Shipped with
Basic features of NAS, such as its software components, general capabilities, and system architecture.	*Overview*	NAS 4.0 Developer Edition (Solaris), NAS 4.0, NAB 4.0
Deploying Netscape Application Server at your site, by performing the following tasks: • Planning your Netscape Application Server environment • Integrating the product within your existing enterprise and network topology • Developing server capacity and performance goals • Running stress tests to measure server performance • Fine-tuning the server to improve performance	*Deployment Guide*	NAS 4.0
Administering one or more application servers using the Netscape Application Server Administrator tool to perform the following tasks: • Deploying applications with the Deployment Manager tool • Monitoring and logging server activity • Setting up users and groups • Administering database connectivity • Administering transactions • Load balancing servers • Managing distributed data synchronization	*Administration Guide*	NAS 4.0

For information about	See the following	Shipped with
Migrating your applications to the new Netscape Application Server 4.0 programming model from version 2.1, including a sample migration of an Online Bank application provided with Netscape Application Server	*Migration Guide*	NAS 4.0 Developer Edition (Solaris), NAS 4.0, NAB 4.0
Creating NAS 4.0 applications within an integrated development environment by performing the following tasks: • Creating and managing projects • Using wizards • Creating data-access logic • Creating presentation logic and layout • Creating business logic • Compiling, testing, and debugging applications • Deploying and downloading applications • Working with source control • Using third-party tools	*User's Guide*	NAB 4.0

For information about	See the following	Shipped with
Creating NAS 4.0 applications that follow the new open Java standards model (Servlets, EJBs, JSPs, and JDBC), by performing the following tasks: • Creating the presentation and execution layers of an application • Placing discrete pieces of business logic and entities into Enterprise Java Bean (EJB) components • Using JDBC to communicate with databases • Using iterative testing, debugging, and application fine-tuning procedures to generate applications that execute correctly and quickly	*Programmer's Guide (Java)*	NAS 4.0 Developer Edition (Solaris), NAB 4.0
Using the public classes and interfaces, and their methods in the Netscape Application Server class library to write Java applications	*Server Foundation Class Reference (Java)*	NAS 4.0 Developer Edition (Solaris), NAB 4.0
Creating NAS C++ applications using the NAS class library by performing the following tasks: • Designing applications • Writing AppLogics • Creating HTML templates • Creating queries • Running and debugging applications	*Programmer's Guide (C++)*	Order separately
Using the public classes and interfaces, and their methods in the Netscape Application Server class library to write C++ applications	*Server Foundation Class Reference (C++)*	Order separately

About This Guide

The *Administration Guide* guide leads you through the tasks that you perform as the administrator of one or more Netscape Application Server (NAS) machines. This guide assumes you have installed NAS on at least one machine. For information about installing NAS, refer to the *Installation Guide*.

You perform most of the administration tasks with NAS Administrator, a GUI-based tool for server and application administration. This tool is described in "About Netscape Application Server Administrator" on page 20.

How This Guide Is Organized

This guide is divided into three parts. If you are new to administering a Netscape Application Server (NAS) machine, begin with Part I, "Getting Started" for an overview of how to start the server and administrator tool. If you are already familiar with administering application servers, skim the material in Part I, "Getting Started" before going on to Part II, "Administering a Single Netscape Application Server."

If you are administering more than one application server, continue to Part III, "Administering Multiple Netscape Application Servers," for additional information specific to a multiple-server enterprise.

Part I: Getting Started

The first part of the *Administration Guide* describes the environment of NAS.

The following chapter is included in this part:

*   Chapter 1, "Performing Basic Tasks with the Administrator Tool," describes how to get started with NAS Administrator, as well as the basic NAS configuration tasks you can perform to begin working with NAS.

Part II: Administering a Single Netscape Application Server Machine

The second part of the *Administration Guide* describes server and application administration procedures for a single NAS machine. The procedures included in this part are those that you are most likely to do right away.

The following chapters are included in this part:

- Chapter 2, "Deploying and Upgrading Applications," describes how to deploy applications from one machine to another using the Deployment Manager, as well as how to upgrade applications already installed on a NAS machine

- Chapter 3, "Monitoring Server Activity," describes the monitoring service provided by NAS Administrator that allows you to chart various attributes of the Executive, Java, and C++ server processes

- Chapter 4, "Monitoring NAS with Third-Party Tools," describes how to monitor NAS using the Simple Network Management Protocol (SNMP).

- Chapter 5, "Logging Server Messages," describes the message-logging service provided by NAS.

- Chapter 6, "Securing Applications," describes how to set up users and groups to provide security for your applications.

- Chapter 7, "Increasing Fault Tolerance and Server Resources," describes how you can increase application performance.

- Chapter 8, "Configuring the Web Connector Plug-In," describes the web connector plug-in, which sends users' requests to applications residing on NAS.

- Chapter 9, "Administering Database Connectivity," describes how to configure data access drivers and apply settings to database connectivity parameters.

- Chapter 10, "Administering Transactions," describes the tasks and conceptual information necessary for administering transactions using NAS Administrator.

Part III: Administering Multiple Netscape Application Server Machines

The third part of the *Administration Guide* describes how to administer multiple NAS machines. Included are more in-depth administration procedures and concepts that apply to a multiple-server enterprise. These procedures focus solely on multiple-server administration, and are used with the single-server procedures described in Part II.

The following chapters are included in this part:

- Chapter 11, "Configuring Multiple Servers," describes how to configure the web connector plug-in, distributed data synchronization, and multicast communication for multiple NAS machines using NAS Administrator.

- Chapter 12, "Administering Multi-Server Applications," describes how to maintain multiple NAS machines at the same time using NAS Administrator

- Chapter 13, "Balancing User-Request Loads," describes load balancing, which optimizes the ability of each NAS machine to process users' requests by keeping those requests balanced among several application servers.

- Chapter 14, "Managing Distributed Data Synchronization," describes how to group NAS machines into data synchronization clusters.

- Appendix A, "Troubleshooting," contains troubleshooting information about your NAS machine.

Documentation Conventions

File and directory paths are given in Windows format (with backslashes separating directory names). For Unix versions, the directory paths are the same, except slashes are used instead of backslashes to separate directories.

This guide uses URLs of the form:

http://*server.domain*/*path*/*file*.html

In these URLs, *server* is the name of server on which you run your application; *domain* is your Internet domain name; *path* is the directory structure on the server; and *file* is an individual filename. Italic items in URLs are placeholders.

This guide uses the following font conventions:

- The `monospace` font is used for sample code and code listings, API and language elements (such as function names and class names), file names, path names, directory names, and HTML tags.

- *Italic* type is used for book titles, emphasis, variables and placeholders, and words used in the literal sense.

Getting Started

- Performing Basic Tasks with the Administrator Tool

Performing Basic Tasks with the Administrator Tool

This chapter describes how to get started with Netscape Application Server (NAS) Administrator, as well as the basic NAS configuration tasks you can perform using either NAS Administrator or at the command line.

The following topics are included in this chapter:

- About Netscape Application Server Administrator

- Starting the Administrator

- Starting Netscape Application Server

- Registering a Netscape Application Server Machine

- Unregistering a Server

- Setting EJB Container Parameters for Run Time

- Using the Netscape Registry Editor

- Updating the Installation Key

- Changing the IP Address

About Netscape Application Server Administrator

Netscape Application Server (NAS) Administrator is a Java application with a graphical user interface that you use to administer one or more NAS machines. NAS administration involves such performance-related tasks as adjusting database connection threads and load-balancing parameters. Server administrators must also configure devices the application server uses, including the web server.

In addition to these administrative duties, keep in mind that NAS requires not only administration of the server itself, but of application components as well. Application administration involves managing application components by grouping, enabling, and partitioning them to achieve better application performance. Application components, the core of a NAS application, are stored on the application server and contain code written by the application developer. Enterprise Java Beans (EJBs), servlets, JavaServer Pages (JSPs), and AppLogic objects are all application components. For more information about each of these, refer to the *Programmer's Guide*.

Administrative tasks are all performed using NAS Administrator. The left panel of the NAS Administrator's main window displays all NAS machines recognized by the administrator tool. The right panel of that window displays individual tool features.

When NAS Administrator is open to the default General window, the toolbar, main window with left and right panels, and the menu bar are shown as illustrated in the following figure:
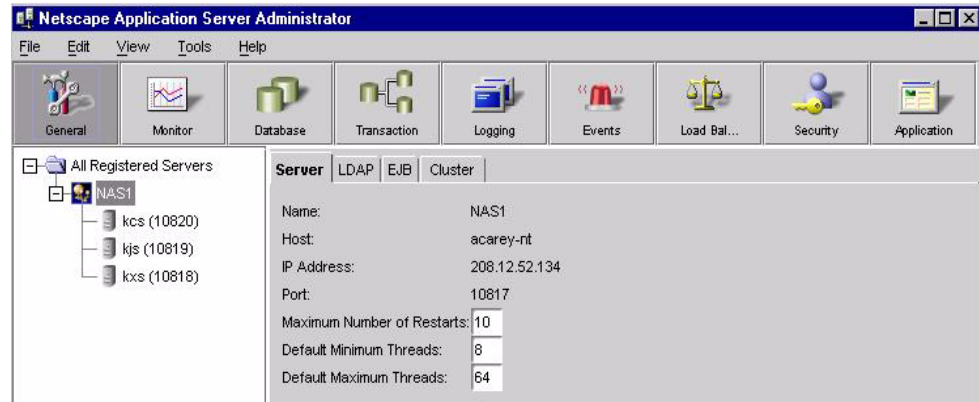
Starting the Administrator

To administer one or more Netscape Application Server (NAS) machines, start the NAS Administrator by performing one of the following tasks:

- On a Windows NT system: from the Start menu, choose Programs, then choose Netscape Application Server 4.0. Finally, choose NAS Administrator.

- On a UNIX system: make sure your PATH variable contains the absolute path to the NAS bin directory by typing the following from a command prompt:

```
ksvradmin &
```

Starting Netscape Application Server

Manually starting Netscape Application Server (NAS) is not usually necessary on a UNIX system; you can choose automatic server start-up when you install NAS. Thereafter, NAS starts automatically on system start-up. However, if you manually stop NAS or if the server crashes, you can manually start the server by performing the following steps:

1. Click the General button on the NAS Administrator toolbar to open the General window.

2. In the left pane of the General window, select the server you want to start.

3. In the right pane of the General window, click Start Server. Note that servers do not appear in the hierarchical tree when they are not running.

Registering a Netscape Application Server Machine

Registering a Netscape Application Server (NAS) machine adds that server to the scope of the administrator tool. This is best done after you add a server or a group of servers to the enterprise.

You must register a NAS machine before you can use NAS Administrator to manage it and the applications stored on it.

To register a NAS machine, perform the following steps:

1. On the NAS Administrator toolbar, click the General button to open the General window.

2. From the File menu, choose New, then Server.

The New Netscape Application Server dialog box appears.
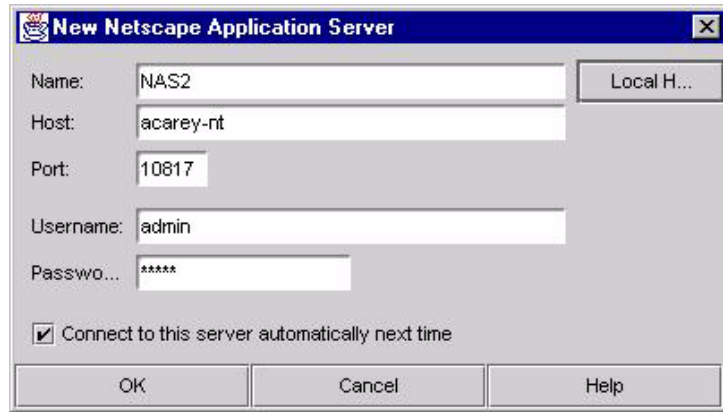
3. In the Name text box, specify the name of the server.

This is an arbitrary name you use to distinguish one server from another. For instance, you might name the servers in your enterprise NAS1, NAS2, NAS3, and so on.

4. In the Host text box, specify the host name of the server.

This is the name of your server machine.

5. In the Port text box, specify the port number for the Administrative Server.

6. In the User Name and Password text box, specify the user name and password you entered during installation of the server or when modifying the Users and Groups.

7. (Optional) To always connect to this server and display it in the Enterprise window, select the "Connect to this server" checkbox. This is the default.

8. Click OK.

The server is registered.

Unregistering a Server

You can remove a server from the scope of the enterprise when that server is no longer available.

To unregister or delete a NAS machine, perform the following steps:

1. On the NAS Administrator toolbar, click the General button to open the General window.

2. In the left pane of the General window, double-click All Registered Servers.

   A list of all registered servers in the enterprise appears.

3. Select the server or servers you want to delete.

4. From the Edit menu, choose Delete.

   The selected server is removed from the scope of NAS Administrator.

Setting EJB Container Parameters for Run Time

Netscape Application Server (NAS) provides an Enterprise Java Bean (EJB) container that enables you to build distributed applications using your own components and components from other suppliers. When you configure NAS for your enterprise, you must set the EJB container's declarative parameters. These parameters determine, for example, when an EJB is removed after being inactive for a specified number of seconds. Set these parameters using the editor in NAS Administrator.

To access the editor, perform the following steps:

1. On the NAS Administrator toolbar, click the General button to open the General window.

2. In the right pane of the General window, click the EJB tab to open the EJB container declarative parameters editor.

The following window appears:

The editor allows you to set the following values:

- default session timeout: if an EJB is unaccessed for the specified number of seconds, it is removed. Applies to stateful session EJBs.

- default passivation timeout: time in seconds that elapses before the state of the EJB, which is currently in memory, is written to disk. This value must be less than session timeout.

- metadata cache size: refers to the metadata cache for EJBs. Value is in number of EJBs.

- implementation cache size: maximum cache size in number of EJBs.

- timer interval: how frequently (in seconds) the EJB pool checks to see if it should passivate or remove an EJB.

You must restart the server before changes take effect.

Using the Netscape Registry Editor

The Netscape Registry Editor is a stand-alone GUI tool that displays registry information for Netscape products. The editor is installed with each instance of Netscape Application Server (NAS) and is similar in appearance and function to the registry editor installed on Windows machines. You should always use the

Netscape Registry Editor to manage registry entries for NAS as it displays values stored not only in your local machine's registry, but in your Directory Server as well.

You can launch the Netscape Registry Editor by typing `kregedit` at the command line on Solaris machines.

For Windows NT machines, click the Windows NT Start button and choose Run. Type `kregedit` and click OK.

The following window appears:
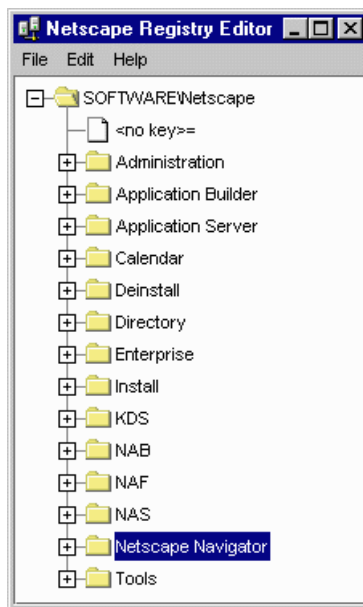
To modify a value in the registry, double-click the entry. The following dialog box appears:
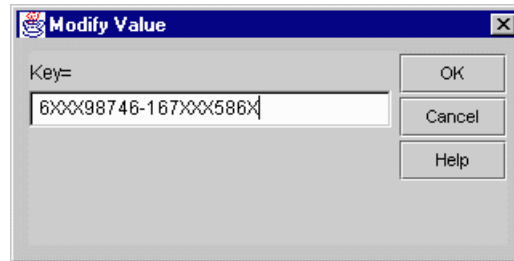
Updating the Installation Key

If you installed Netscape Application Server (NAS) under an evaluation license, the server stops running at the end of the evaluation period. If you have extended the evaluation period or purchased the server, you will need to update the installation key. Updating the installation key saves you from having to reinstall the server software and reconfigure the environment.

To reset the installation key, perform the following steps:

1. Shutdown NAS.

2. Open the Netscape Registry Editor by typing `kregedit` at the command line.

   See "Using the Netscape Registry Editor" on page 25.

The following window appears:
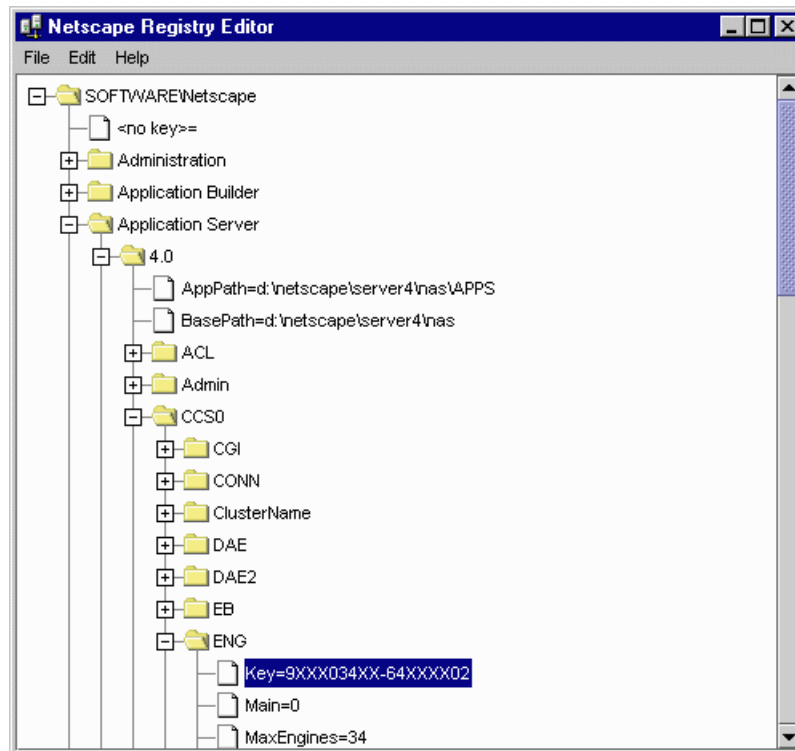
3. Open the following key:

    ```
    SOFTWARE\NETSCAPE\APPLICATION SERVER\4.0\CCS0\ENG
    ```

4. Double-click the `Key` String value and enter the new Key value.
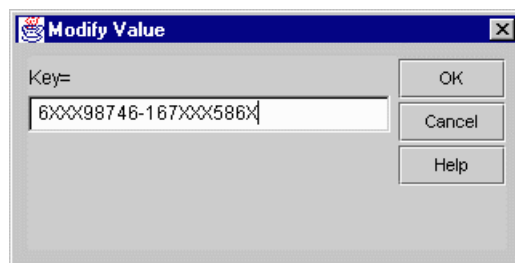
5. Click OK.

6. Close the registry editor.

7. Restart NAS.

Changing the IP Address

When a Netscape Application Server (NAS) machine address changes, such as when the machine is moved, you must update the registry of that machine with the new address. If the machine participates in data synchronization, you must also update the registry of the other machines in the same cluster. Rather than locate every instance of the IP address in the registry and change each instance manually, you can use kregedit to update the entire registry with the new IP address.

To change the IP address, perform the following steps:

1. Open the Netscape Registry Editor by typing `kregedit` at the command line.

See "Using the Netscape Registry Editor" on page 25.

2. Open the following key:

```
SOFTWARE\NETSCAPE\APPLICATION SERVER
```
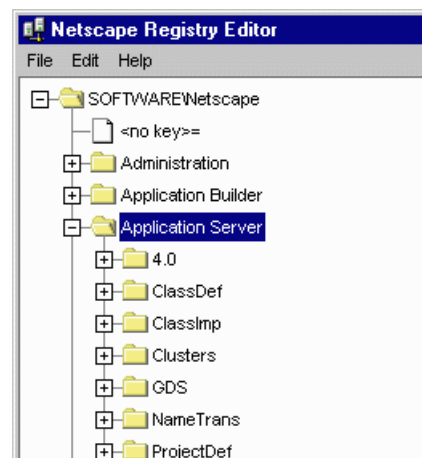
3. From the Edit menu, choose Change IP Address.

The following dialog box appears:
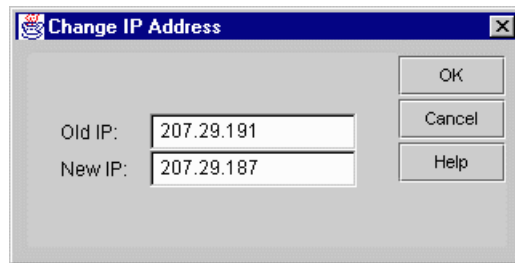
4. Enter the old and new IP address.

5. Click OK to save your changes.

Administering a Single Netscape Application Server

2

- Deploying and Upgrading Applications

- Monitoring Server Activity

- Monitoring NAS with Third-Party Tools

- Logging Server Messages

- Securing Applications

- Increasing Fault Tolerance and Server Resources

- Configuring the Web Connector Plug-In

- Administering Database Connectivity

- Administering Transactions

Chapter

2

Deploying and Upgrading Applications

This chapter describes how to deploy and upgrade applications on a Netscape Application Server machine.

The following topics are included in this chapter:

* Deploying an Application Using Deployment Manager

* Deploying Application Files Manually

* Upgrading an Application

* Editing EJB Properties at Run Time

Deploying an Application Using Deployment Manager

You can deploy an application using the Deployment Manager, a separate tool accessible from Netscape Application Server (NAS) Administrator or from Netscape Application Builder (NAB). When you deploy an application, the Deployment Manager installs all the application's files and registers all its components on the destination server, a NAS machine. An application must be deployed before it can be used.

Generally, a developer creates an application on a development machine using tools such as NAB, then deploys that application from the development machine to an application server using the Deployment Manager. NAS administrators, on the other hand, might use the Deployment Manager to both download an application from a NAS machine and subsequently redeploy that application to one or more applications servers.

Specifying Application Directories

Before you deploy an application, you can change the application root directories that specify where the Java Server or C++ Server processes should look for application component files such as query files and template files. By referencing application root directories, you can move these components around without having to rewrite application code. If you do not specify particular root directories, application files are deployed to default directories.

The Java Server and C++ Server processes use root directories to find application components when those component are needed. For example, after a result set is returned from the database, the application most likely uses a template to format the data. The process, whether Java Server or C++ Server, scans the template root directory or directories to find the specified template file referenced by the application or query file.

To specify application root directories, perform the following steps:

1. From the NAS Administrator toolbar, click the Application button to open the Application window.

2. In the left pane of the Application window, select the NAS machine whose application root directories you want to change.

3. In the right pane of the Application window, use the text boxes to modify root directories for the specified application components as shown in the following figure:

Use a semicolon-delimited list when specifying more than one directory for an application component.

4. Click Apply Changes to save your changes to your NAS machine.

Packaging Application Files for Deployment

Before you deploy an application, you must bundle its files into a JAR file or "package." A package contains information about the application files, such as their type and destination directory. You can select specific application files to include in the package or bundle an entire application into a package ready for deployment.

When you later deploy the package, the bundled application files are automatically distributed to their appropriate directories on one or more servers. For example, Java files and query files are automatically sent to the applications directory of your application server. At the same time, web server files, such as HTML templates, are automatically deployed to a user-defined directory on the web server. In addition, the HTML template files that reside on the application server, rather than on the web server, are deployed to their proper directory.

Creating a Package

To create a package, perform the following steps:

1. Open the Deployment Manager in one of two ways:

   - From the NAS Administrator Application window, click the Launch NASDM button.

   - From the Windows Start menu, under Programs, choose Netscape Application Server 4.0, then NAS Deployment Manager.

2. From the Deployment Manager's File menu, choose New Project.

   The following dialog box appears:
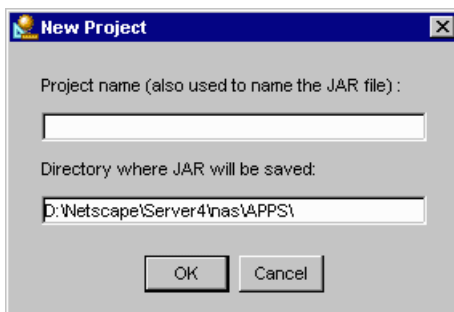
3. Type a name for your project.

   This name is also used to identify your JAR file or package. The project name and directory name must be the same.

4. If necessary, edit the path to the directory where your JAR file is stored.

5. Click OK.

6. From the Edit menu, choose Insert.

   You can insert files one at a time, insert all files in a directory, or insert all files from the subtree of a selected directory.

7. Navigate to the directory where application files are stored and select the files to include in the package.

The following dialog box appears:
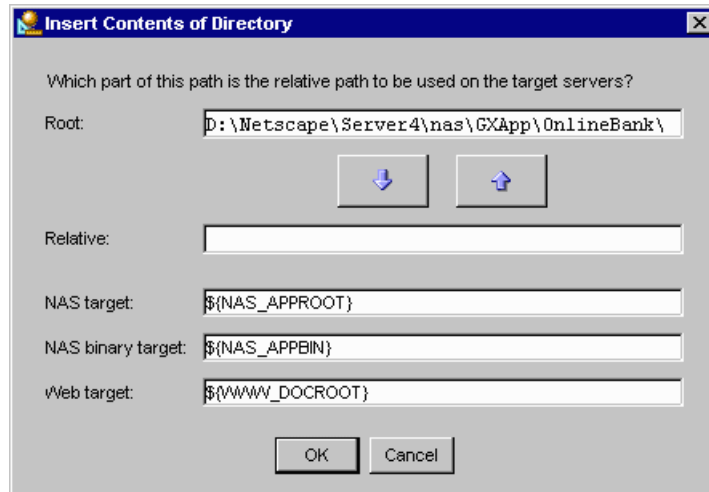
8. If necessary, edit the application and web file root directories for your application server using the arrow buttons.

On a Windows machine, the default root directory for application files is similar to the following:

```
C:\Netscape\Server4\nas\APPS\
```

When you deploy an application using the Deployment Manager, the destination server organizes the application files relative to the root directory on your development machine. For instance, application files stored on your development machine in the location

```
C:\Netscape\Server4\nas\APPS\myapps\
```

are organized on the destination server in the following location:

destination server application root directory/myapps/

In the previous figure, pressing the down arrow button would add the `OnlineBank` portion of the path to the target directories as shown in the following figure:
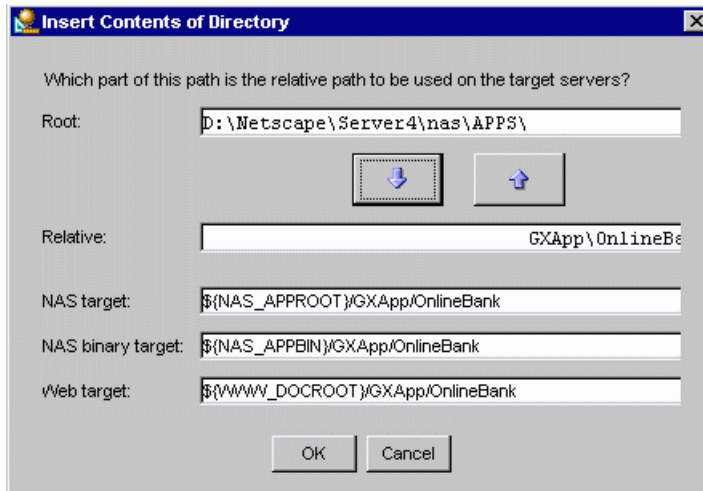
9. Click OK.

The following dialog box appears:
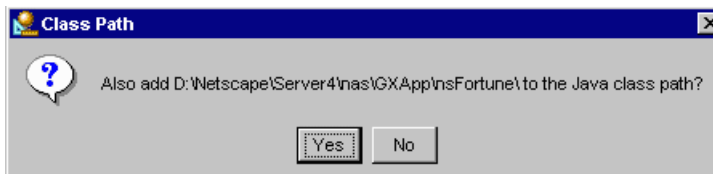
10. Click Yes if the directory contains `.class` files that will be used in the application.

If the directory does not contain `.class` files that will be used in the application, there is no need to add to the class path. If the directory does contain `.class` files and you click No, the Deployment Manager will not be able to load these `.class` files or deploy them correctly.

The Deployment Manager must have a class path in order to load `.class` files. Class files must be loaded to edit an EJB's deployment descriptor or to determine if a `.class` file is a servlet.

11. Once the application files appear in the JAR window as shown in the following figure, create meta-info files such as:

   • EJB descriptors: see "Preparing an Enterprise Java Bean for Deployment" on page 41.

   • `servletInfo` files: see "Editing a Servlet" on page 44.
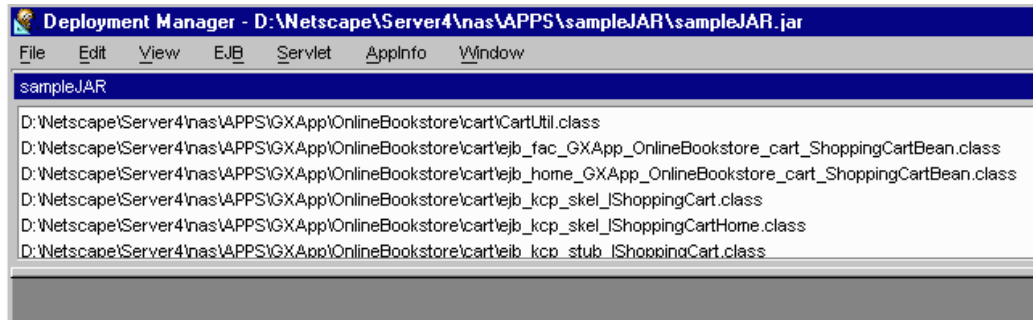
12. From the File menu, choose Save JAR when all files you want to include in the package appear in the list.

   If the list of files in your project shows files marked in red with an asterisk, these files require additional action. Skip to step 13.

   If your project files require no further action, the following dialog box appears:
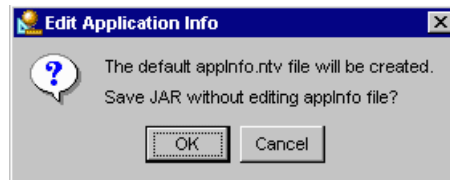
   A Name Type Value (NTV) file called `appinfo.ntv` is created by default if such a file doesn't already exist. This file contains the name of the JAR file, associated session information, and a list of servlets included in the JAR file.

13. Choose one of the following options:

- Click Cancel to open the appInfo editor.

  If you choose to edit the appInfo file, see "Editing a Servlet" on page 44 for more information about this editor.

- Click OK to save the JAR file without editing the appInfo file.

14. If your project files require additional action (indicated by red type and an asterisk next to the files in the JAR window), the following dialog box appears:
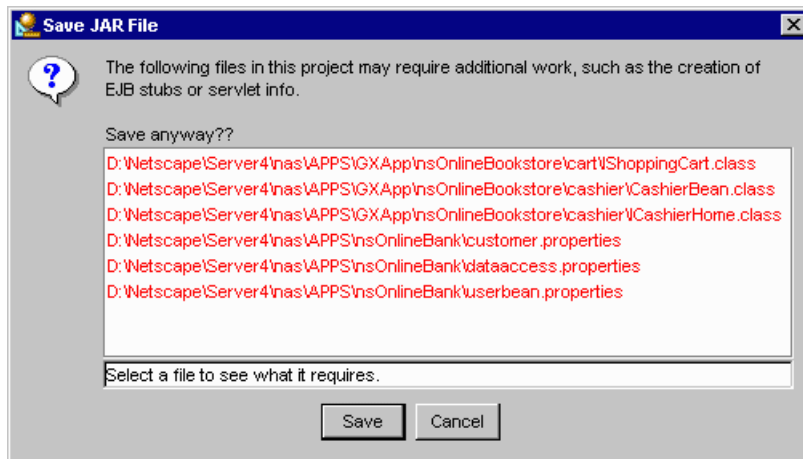
You can click a file to see what action is required. For instance, an EJB may be missing stubs. Click Cancel to dismiss the dialog box and correct these problems or click Save to save the JAR as-is.

15. When saving a JAR file, you can filter `.java` files out of the package by selecting the corresponding checkbox in the Save JAR File dialog box as shown in the following figure:
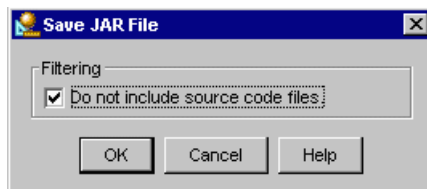
16. Click OK.

If you want to save an "in-progress" list of files without completing the package, you can save the list of files as a `.dxm` file by choosing Save List from the File menu. You can later open the file and modify it.

The package is ready for deployment.

Preparing an Enterprise Java Bean for Deployment

You can prepare an Enterprise Java Bean (EJB) to include in your JAR file using the packaging tool. To prepare an EJB, perform the following steps:

1. From the list of files in the JAR window, select the following three types of classes:

   • implementation class

   • remote interface class

   • home interface class

   These three classes are required to create an EJB. You can add other classes in addition to these three.

2. From the EJB menu, choose Create Descriptor.

   A dialog appears showing which classes are used in which roles.

   If you selected only one `.class` file, that file appears next to the appropriate class type (Remote Interface, for example). You can then select the remaining two required files from drop-down boxes that appear next to the appropriate class type as shown in the following figure:
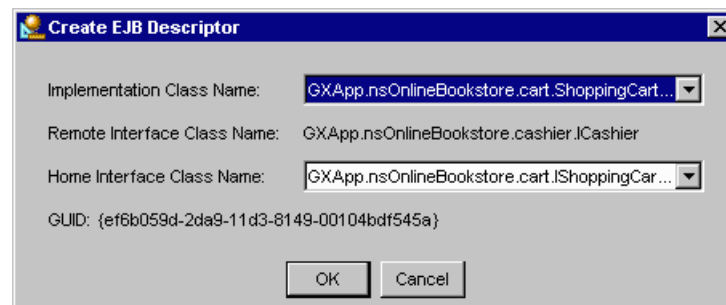
The drop-down boxes list only those `.class` files that satisfy the requirements of the EJB.

3. Click OK when each field is satisfied.

The EJB deployment descriptor editor appears as a separate tool in the Deployment Manager window:
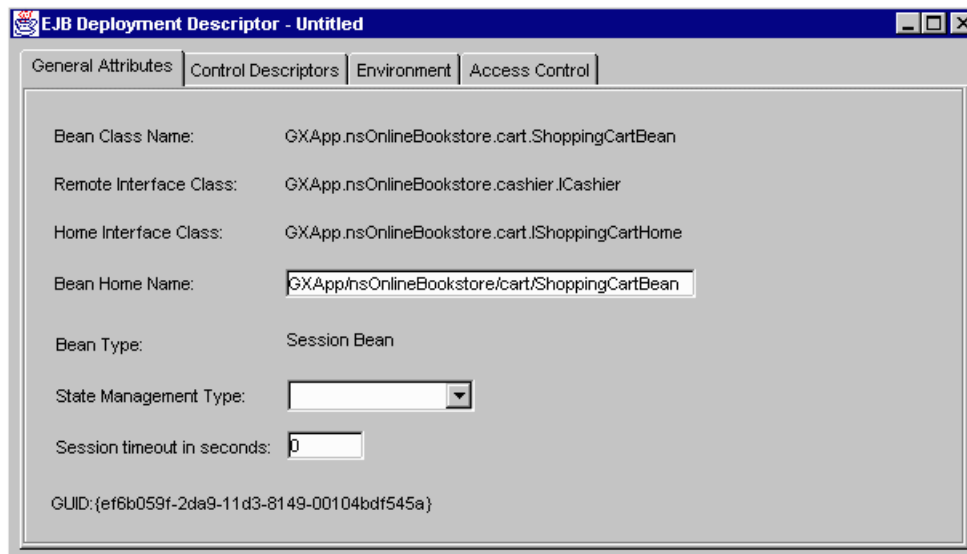
4. Edit EJB deployment information if necessary using this editor.

See "Editing an EJB's Deployment Descriptor" on page 45.

5. From the Deployment Manager's EJB menu, choose Save Descriptor.

The file is saved as a `.properties` file.

If your EJB is a session bean rather than an entity bean, and you have neglected to specify the state management type in the bean's deployment descriptor, the following dialog box appears:
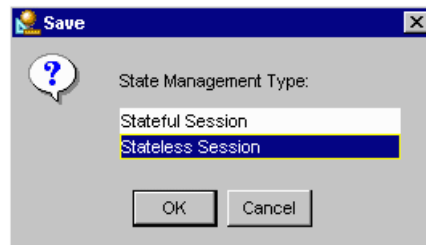
6. Select the type of your EJB: Stateful Session or Stateless Session.

   A stateless session bean is completely transient and encapsulates a temporary piece of business logic needed by a specific client for a limited time span.

   A stateful session bean is also transient, and uses a "conversational state" to preserve information about its contents and values between client calls. The conversational state enables the container to maintain information about the state of the session bean and to recreate that state at a later point in program execution when needed.

   The following dialog box appears:
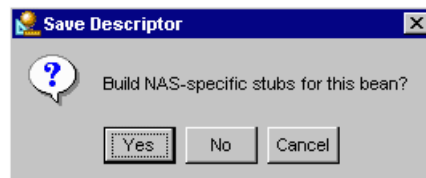
7. Decide whether to build stubs for the EJB or not.

   Stubs and skeletons are required by the EJB container and must be deployed with the application files. These stubs and skeletons enable remote communication and allow the container to intercept all bean requests. When you create stubs and skeletons, the Deployment Manager automatically adds them to the list of application files.

After the Deployment Manager builds the stubs, the following status window appears (this may take a moment or two):
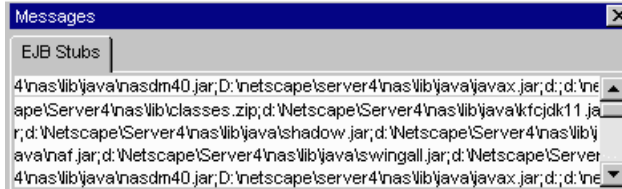
If you choose to build stubs later, you can select Build Stubs from the EJB menu while the EJB deployment descriptor editor is open and displaying that EJB. You can choose Build All Stubs at anytime to build stubs for all EJBs whose `.properties` files are in the list of files appearing in the JAR window.

Editing a Servlet

You can edit any class file in a package as a servlet. The file is then marked as a servlet and an `.ntv` file is created for it by default.

To edit a servlet, perform the following steps:

1. From the Servlet menu, choose Edit Info.

   The NTV editor appears.
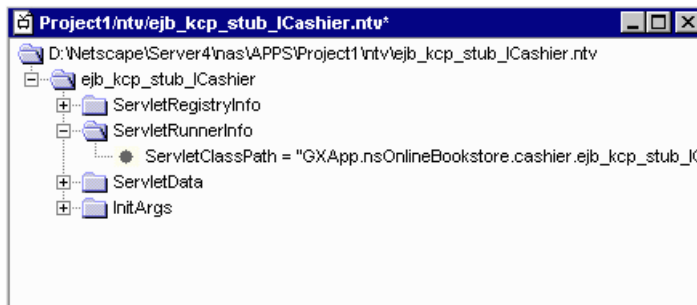
2. Right-click a branch in the tree to edit that branch or remove it.

See "Creating a Package" on page 36 for more information about `.ntv` files.

Editing an EJB's Deployment Descriptor

Deployment descriptors include the declarative attributes associated with an EJB (or bean). These attributes tell an EJB's container how to manage the bean. A container is where an EJB "lives" from its creation to its destruction. The container manages the EJB's life cycle and support services while providing services that allow clients to look up the interfaces of installed EJB classes.

A developer might edit an EJB's deployment descriptor as part of a cycle of developing and testing an application. The editor allows either administrators or developers to easily modify the attributes of EJBs to better work within an application.

You can edit an EJB's deployment descriptor using the editor shown in the following figure:
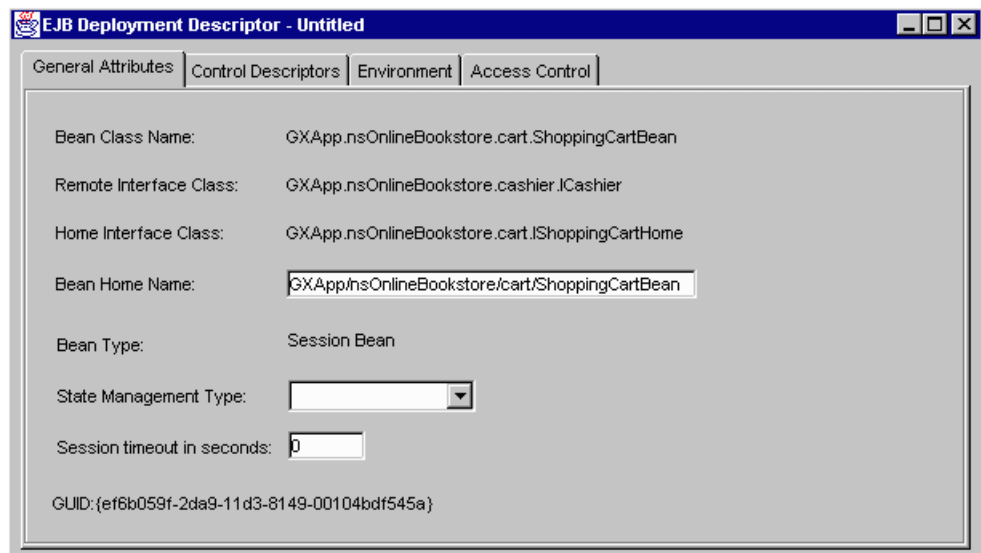
Editing General Attributes

To edit the General Attributes of an EJB, perform the following steps:

1. If the editor is not already open, you can open it by right-clicking on a
   `.properties` file displayed in the JAR window and choosing Edit
   Descriptor.

   You can edit the Bean Home Name as necessary, change the State
   Management Type of the bean, or edit the bean's session timeout value.

   See the *Programmer's Guide* for more information about these values.

2. From the File menu, choose Save Descriptor to save your changes.

Editing Control Descriptors

The Control Descriptor tab allows you to configure meta-data for an EJB at
deployment time. You can edit transaction isolation levels, transactional
attributes, and the mode entry for the bean.
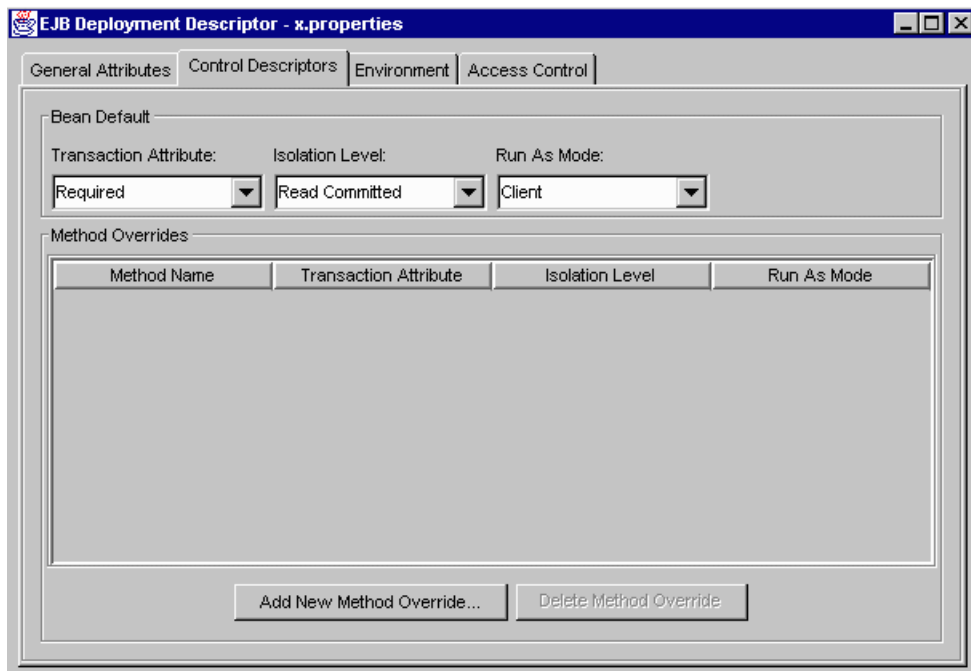
To edit the Control Descriptors for an EJB, perform the following steps:

1. Click the Control Descriptors tab to display the window as shown in the previous figure.

   The EJB's meta-data appears in the Bean Default area of the window. You can edit this information for the bean as a whole or choose a particular method within the bean and change that method's meta-data in the Method Overrides area of the window.

   The Run As Mode entry defines the identity a bean uses during execution and how a bean identifies its user to other beans or resources. Most commonly, this value is Client, which means that a bean or its methods are executed using the client's identity. See Chapter 12, "Writing Secure Applications," in the *Programmer's Guide* for more information.

   Transaction Attributes specify when a bean needs to begin a transaction. See the Chapter 8, "Handling Transactions with EJBs," in the *Programmer's Guide* for more information.

   An isolation level specifies how much or how little a transaction can see of other, simultaneous interactions with a database. For more information, see Chapter 8, "Handling Transactions with EJBs," in the *Programmer's Guide*.

Editing Environment Properties

See "Editing Environment Properties" on page 66.
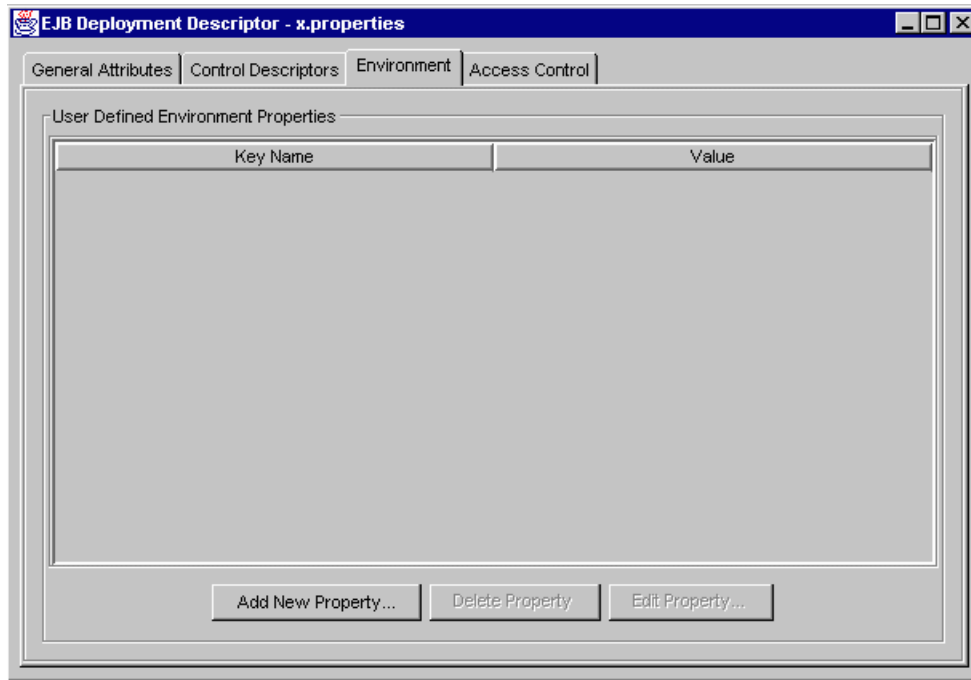
Editing Access Control

See "Editing EJB Access Control" on page 62.
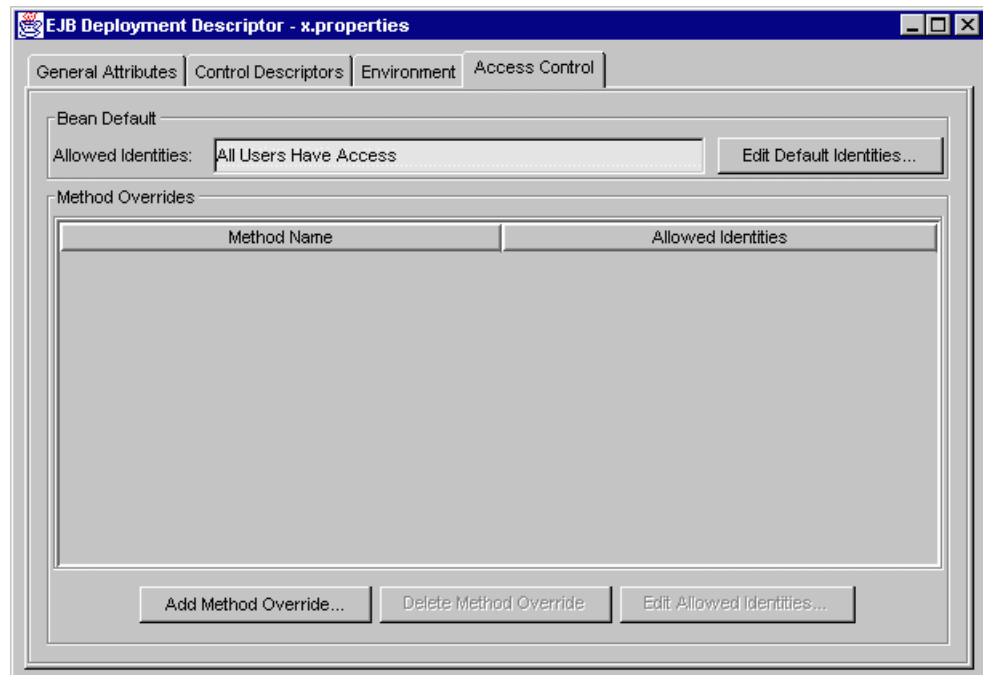
Deploying an Application

After you have created a package using the packaging tool, you use the Deployment Manager to send the package to a NAS machine.

If there is more than one JAR file in a single directory, you can deploy multiple JAR files at the same time from that directory.

To deploy a JAR using the Deployment Manager, perform the following steps:

1. From the File menu, choose Deploy.

   You can deploy the currently open package, or you can choose a directory and deploy one or more packages stored therein.

You are prompted to save the JAR file as well as the `appInfo` file if either has changed since you last saved it.

2. From the list of registered servers, choose a NAS machine to deploy to.

To register additional servers, click Register and enter the necessary server information.
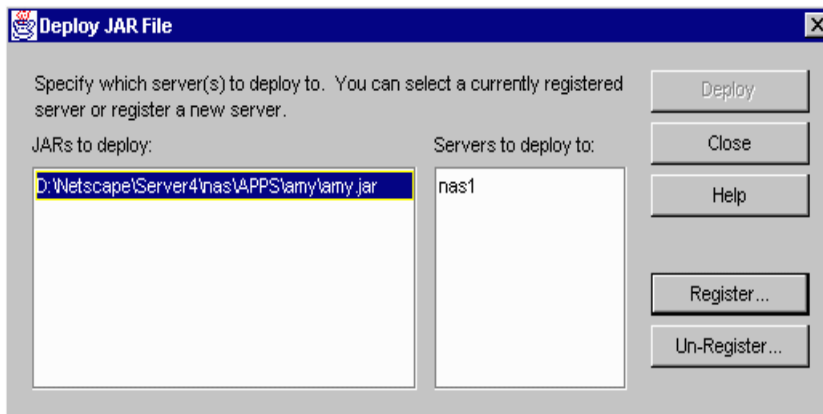
3. Click Deploy.

The status of the deployment process appears in a separate window.

Downloading a Package

Once an application has been installed on an application server, you can download that application to your machine using the Deployment Manager. When you download an application, you save that application's JAR file to your local machine or another machine on your network.

To download an application, perform the following steps:

1. From the File menu of the Deployment Manager, choose Download.

   The following dialog box appears:
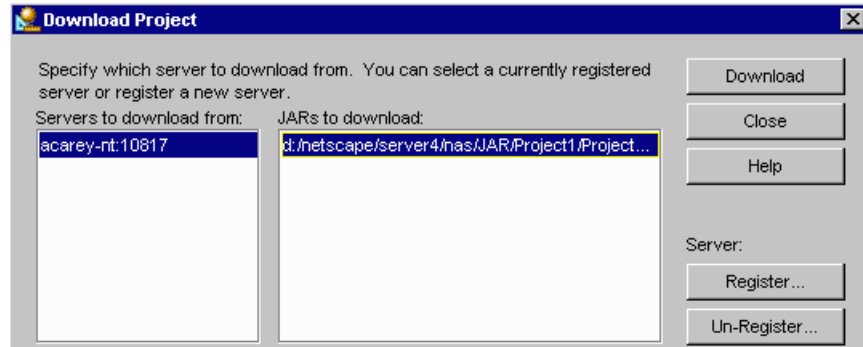
2. Select a server from which to download an application.

   You can register additional servers by clicking the Register button. See "Deploying an Application" on page 49 for details.

3. From the list of JAR files found on the selected server, select a JAR file to download.

4. Click Download.

5. Choose the directory where you will save the JAR file on your local machine or the network.

   The message window displays the status of the file you are downloading.

Deploying Application Files Manually

The Deployment Manager does not allow you to deploy individual application components that are not part of an application. Instead, you can deploy individual application files manual.

This section contains separate procedures for deploying Enterprise Java Beans, servlets and JavaServer Pages (JSPs), and data sources:

- Manually Deploying EJBs

- Manually Deploying Servlets and JSPs

- Manually Deploying Data Sources

Manually Deploying EJBs

To deploy an EJB manually, perform the following steps:

1. Compile the EJB.

2. Generate stubs and skeletons.

3. Create a deployment descriptor.

4. Copy files to NAS.

5. Register the EJB.

Compile the EJB

Using `javac`, compile all EJB files including the bean, home interface, and remote interface.

The following example shows what you might type at the command line on Solaris operating systems:

```
% $GX_ROOTDIR/usr/java/bin/javac ShoppingCartBean.java
IShoppingCart.java IShoppingCartHome.java
```

The following example shows what you might type at the command line on Windows NT, using `javac` in `%NASPATH%`:

```
Z:\> C:\Netscape\server4\nas\bin\javac ShoppingCartBean.java
IShoppingCart.java IShoppingCartHome.java
```

In these examples, `ShoppingCartBean.java` is the bean implementation file, `IShoppingCart.java` is the remote interface, and `IShoppingCartHome.java` is the home interface.

Generate Stubs and Skeletons

Stubs and skeletons are required by the EJB container and must be deployed with the application files. These stubs and skeletons enable remote communication and allow the container to intercept all bean requests.

Using `ejbc`, generate stubs and skeletons for your EJB following these examples:

```
$GX_ROOTDIR/bin/ejbc -sf cart.ShoppingCartBean cart.IShoppingCart
cart.IShoppingCartHome

Z:\>ejbc -sf cart.ShoppingCartBean cart.IShoppingCart
cart.IShoppingCartHome
```

`-sf` is used for stateful session beans.

Create a Deployment Descriptor

Deployment descriptors include the declarative attributes associated with an EJB (or bean). These attributes tell an EJB's container how to manage the bean. A container is where an EJB "lives" from its creation to its destruction. The container manages the EJB's life cycle and support services while providing services that allow clients to look up the interfaces of installed EJB classes.

For more information, see "Creating EJB Property Files" in Chapter 10, "Creating Configuration Files," in the *Programmer's Guide*.

Copy the Files to NAS

Copy the class files from your development machine to the destination server's `APPS` directory. You must preserve the directory structure used on the development machine.

Register the EJB

Once you have copied the files onto the destination machine, you must register the EJB using `beanreg` on the NAS host machine. BeanReg registers a bean locally using the `.properties` file that describes the bean. Registering your bean requires the deployment descriptor file you created in "Create a Deployment Descriptor."

For example, on Solaris machines, type the following at the command line;

```
% $GX_ROOTDIR/bin/beanreg filename.properties
```

On a Windows NT machine, type the following at the command line, using `beanreg` in `%NASPATH%`:

```
Z:\>beanreg filename.properties
```

Manually Deploying Servlets and JSPs

To deploy a servlet or JSP manually, perform the following steps:

1. Create servlet configuration files.

2. Copy files to NAS.

3. Register the servlet.

Create Servlet Configuration Files

You must create a configuration file for each servlet, and one for the application as a whole. These configuration files must be referenced in an application-wide configuration file called `appInfo.ntv`. Servlets that are not part of a specific application are part of the "generic" application, which also must have a configuration file `appInfo.ntv`.

Servlet configuration files can be named anything as long as the name contains the `.ntv` suffix and the filename is referenced in `appInfo.ntv`. They must be placed in a certain directory, which is specified in the next section, "Copy the Files to NAS."

The following is an example of an `appInfo.ntv` file:

```
NTV-ASCII
{
   "SessionInfo"      NTV      {
    "timeout"      Int     "400",
    "flags"          StrArr     ["SESSION DISTRIB"],
    "sessionManagement"    Int     "0"
   },
```

```
   "ServletFiles"     StrArr     ["servInfo"],
   "AppName"     Str     "nsOnlineBookstore",
}
```

`AppName` refers to the name of the application, which, in this case, is nsOnlineBookstore. `servInfo` refers to the servletInfo file, `servInfo.ntv`. You can find this file in the following location:

NAS install directory/APPS/NSOnlineBookstore/ntv

For more information, see Chapter 10, "Creating Configuration Files," in the *Programmer's Guide*.

Copy the Files to NAS

Servlets and JSPs can be part of an application or exist outside of an application, depending upon how the web client invokes the servlet.

To copy these files to NAS, first follow the directions that apply to your application component:

- Copy Servlets as Part of an Application
- Copy JSPs as Part of an Application
- Copy Servlets Not Part of an Application
- Copy JSPs Not Part of an Application

Next, copy the class files into the NAS class path, which is usually `$GX_ROOTDIR/APPS`. You must preserve the directory structure when copying the class files. See the examples in "Sample Directory Structures for Servlets and JSPs Not Part of an Application" on page 57 and "Sample Directory Structures for Servlets and JSPs as Part of an Application" on page 57 for details.

Finally, copy static content like HTML and GIF files to the web server documentation root.

Copy Servlets as Part of an Application

For a servlet that's part of an application, the following URL would appear in the web client:

`http://$HOSTNAME:$PORT/NASApp/$AppName/$ServletName`

In this URL, *$HOSTNAME* is the DNS name of the host machine, *$PORT* is the TCP/IP port, NASApp is a key defined the registry to signal to NAS that the URL references a servlet, *$AppName* is the AppName variable in the `appInfo.ntv` files, and *ServletName* is the name of the servlet.

To mirror this URL on your own machine, copy the application's NTV files to the `$GX_ROOTDIR/APPS/`*$AppName*`/ntv` directory.

Note   The `appInfo.ntv` file must point to all relevant `servletInfo.ntv` files. If you change the names of the `servletInfo.ntv` files, you must reflect those changes in the `appInfo.ntv` file.

Copy JSPs as Part of an Application

For a JSP that's part of an application, copy the JSP into the `$GX_ROOTDIR/APPS/`*$AppName* directory, then refer to them using *$AppName*`/`*jspnam*`e.jsp`.

Copy Servlets Not Part of an Application

For a servlet that's not part of an application, the following URL would appear in the web client:

`http://`*$HOSTNAME:$PORT*`/servlets/`*$ServletName*

In this URL, *$HOSTNAME* is the DNS name of the host machine, *$PORT* is the TCP/IP port, `servlets` is a non-changable string, and *ServletName* is the name of the servlet.

To mirror this URL on your own machine, copy the `servletInfo.ntv` files to the `$GX_ROOTDIR/APPS/ntv` directory. You must update the `appInfo.ntv` file with the new `servletInfo.ntv` files.

Copy JSPs Not Part of an Application

For a JSP that's not part of an application, copy the JSP into the `$GX_ROOTDIR/APPS/` directory, then refer to them using *jspnam*`e.jsp`.

Sample Directory Structures for Servlets and JSPs Not Part of an Application

For servlets and JSPs not in an application, use the following directory structures as examples:

NAS install directory (`APPS` is in the NAS class path):

```
/disk2/ns-home/nas/APPS/
```

Servlet with class name `javaSpec`:

```
/disk2/ns-home/nas/APPS/javaSpec.jsp
```

JSP referenced by `javaSpec.jsp`:

```
/disk2/ns-home/nas/APPS/javaSpec.jsp
```

Default NTV directory for all servlets and JSPs not in any application:

```
/disk2/ns-home/nas/APPS/ntv
```

Default `appInfo.ntv` file referencing all servlets not in an application:

```
/disk2/ns-home/nas/APPS/ntv/appInfo.ntv
```

Sample Directory Structures for Servlets and JSPs as Part of an Application

For servlets and JSPs in an application, use the following directory structures as examples. Here, the application name is Project1.

NAS install directory (`APPS` is in the NAS class path):

```
/disk2/ns-home/nas/APPS/Project1/
```

Location of all NTV files for Project1:

```
/disk2/ns-home/nas/APPS/Project1/ntv/
```

Location of `appInfo.ntv` for Project1:

```
/disk2/ns-home/nas/APPS/Project1/ntv/appInfo.ntv
```

`servletInfo.ntv` file for `Project1.webapp.NASservlet` servlet which is referenced in the `appInfo.ntv` file for this project:

```
/disk2/ns-home/nas/APPS/Project1/ntv/NASservlet.ntv
```

NASservlet servlet with a full class name of
`Project1.webapp.NASservlet`:

`/disk2/ns-home/nas/APPS/Project1/webapp/NASservlet.class`

NASservlet JSP file referencable with the name `Project1/webapp/`
`NASservlet.jsp`:

`/disk2/ns-home/nas/APPS/Poject1/webapp/NASservlet.jsp`

URL to access `NASservlet`:

`http://warp/NASApps/Project1/NASservlet`

Register the Servlet

On the NAS host machine, run one of the following scripts on the
`appInfo.ntv` file to register all servlets in the `appInfo.ntv` file:

On Solaris: `servletReg.sh`

On Windows NT: `servletReg.bat`

For example, on Solaris, you might type the following at the command line:

`%$GX_ROOTDIR/bin/servletReg.sh -i appInfo.ntv`

Manually Deploying Data Sources

To deploy a data source manually, perform the following steps:

1. Create the data source file.

2. Register the data source.

Create the Data Source File

The data source file contains information necessary for accessing the database such as a user name and password. Using a text editor, create a data source file (*filename*.props) using the following example as a guide:

```
DataBase=ksample
DataSource=ksample
UserName=kdemo
PassWord=kdemo
ResourceMgr=rm_orcl        #Optional
```

Data source files are not referenced at run time, so you need not place them in a specific directory on your NAS machine. However, it is good convention to place them in the APPS/*$AppName* directory.

For more information, see Chapter 10, "Creating Configuration Files," in the *Programmer's Guide*.

Register the Data Source

After you have created the data source file, you must run dsreg on the NAS host machine.

For example, on Solaris machines, you might type the following at the command prompt:

```
$GX_ROOTDIR/bin/dsreg "jdbc/BookstoreDS" BookstoreDS.props
```

where jdbc/BookstoreDS is the data source name.

On Windows NT machines, you might type the following at the command prompt:

```
Z:\>dsreg "jdbc/BookstoreDS" BookstoreDS.props
```

Upgrading an Application

Frequently, application developers must upgrade an application that is already installed on an application server. Upgrading an application involves deploying updated application files to your Netscape Application Server (NAS) machine through the Deployment Manager, often to fix bugs or to add features to

deployed applications. The application developer is ultimately responsible for this upgrade; however, you must work with the developer to prepare your NAS machine for the updated application files.

If you are upgrading an Enterprise Java Bean (EJB), you must prepare your NAS machine by stopping the server. You need not stop the server when upgrading JavaServer Pages (JSPs), servlets or AppLogics.

See the *Programmer's Guide* for specific information about application components and their requirements.

Dynamically Reloading Components

You can reload servlets and JavaServer Pages (JSPs) into NAS without restarting the server. Simply re-deploy (and thus overwrite) the servlet or JSP. The next time it is required by the application, NAS automatically reloads that component. An exception to this is that if you change the session method in the application's configuration file, you must restart the web server as well as NAS.

Enterprise JavaBeans (EJBs) can not be dynamically reloaded. You must restart the server after reloading an EJB.

For more information about dynamic reloading, see Appendix B, "Dynamic Loading," in the *Programmer's Guide*.

Editing EJB Properties at Run Time

An Enterprise Java Bean (EJB) instance is created and managed by a container class, but some properties of an EJB can be customized at run time with the declarative properties editor. The declarative properties editor, which is a part of Netscape Application Server (NAS) Administrator, allows you to view read-only attributes of an EJB's deployment descriptor as well as edit application environment properties and the default session timeout value.

To launch the editor, perform the following steps:

1. Click the Application button on the NAS Administrator toolbar to open the Application window.

2. From the tree in the left pane of the window, select the EJB whose properties you want to edit.

For example, `dataaccess` is selected in the following figure:
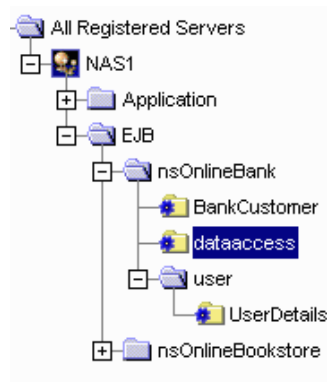
3. In the right pane of the Application window, click the Edit Deployment Descriptor button.

The following dialog box appears:

Editing General Attributes of an EJB

Once you have launched the declarative properties editor, the General Attributes tab appears by default. Information about the EJB you selected in the left pane of the Application window appears, including the EJB's class names and type.

You can edit the selected EJB's session timeout in seconds by entering a value in the corresponding text box. This value indicates how long the EJB can exist without being accessed. When the timeout value is reached, the EJB is removed.

Editing EJB Access Control

Click the Access Control tab to view and edit the access control lists that apply to the selected EJB and its methods. Access control lists specify security identities for a method or an entire EJB. These security identities are users or roles who are allowed to invoke the method.

The Allowed Identities field shows which users are currently able to access the selected bean. To modify Allowed Identities, perform the following steps:

1. Click Choose Default Identities.

   The following dialog box appears:
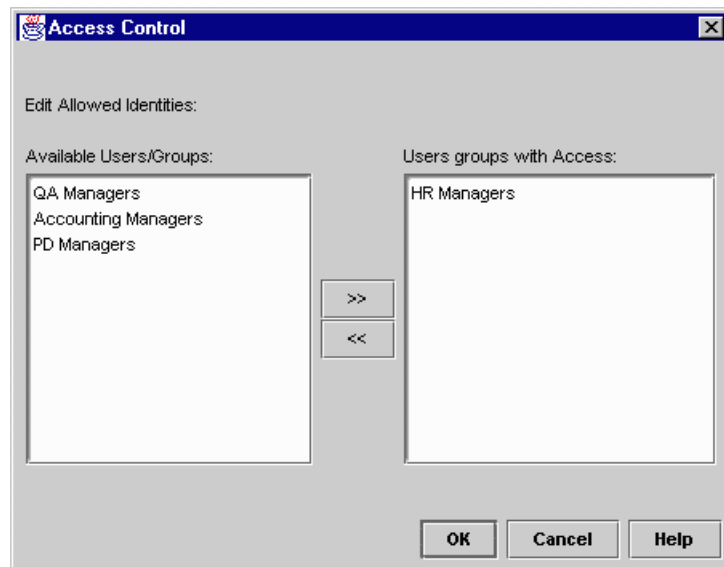
2. Add or remove allowed identities from the list boxes using the arrow buttons.

3. Click OK to dismiss the dialog box.

You can also modify access control at the method-level by applying an override to particular methods. For instance, if All Users have access to a DataAccess bean, you can limit the users who may access the `create()` method of that bean by setting a method override.

To add a method override, perform the following steps:

1. Click the Add Method Override button.

The following dialog box appears:
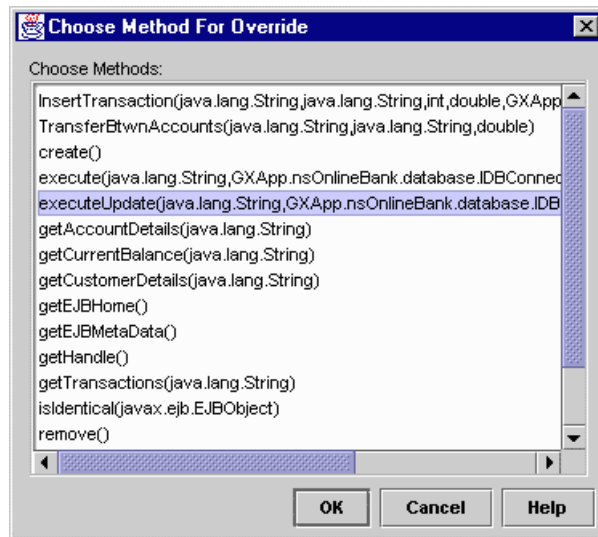
2. Select one or more methods.

3. Click OK.

The method appears in the Method Overrides sections of the Access Control tab as shown in the following figure:

| General Attributes | Environment | Access Control |

Bean Default

Allowed Identities: All Users Have Access   **Choose Default Identities...**

Method Overrides

Method Name	Allowed Identities
executeUpdate(java.lang.String,GXApp.nsOnlineBank.database.IDBC...	HR Managers
getNextTransID(GXApp.nsOnlineBank.database.IDBConnection)	HR Managers
getPrimaryKey()	PD Managers

Add Method Override...   **Delete Method Override**   **Edit Allowed Identities...**

4. Click a method to select it.

5. Click the Edit Allowed Identities button.

   The following dialog box appears:

Access Control

Edit Allowed Identities:

Available Users/Groups:	Users groups with Access:
QA Managers	HR Managers
Accounting Managers	
PD Managers	

>>

<<

OK   **Cancel**   **Help**
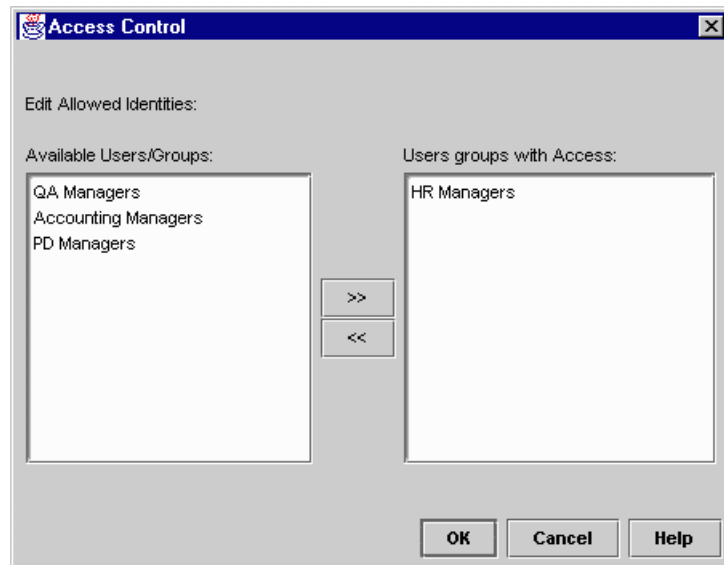
6. Add or remove Allowed Identities from the list boxes using the arrow buttons.

7. Click OK.
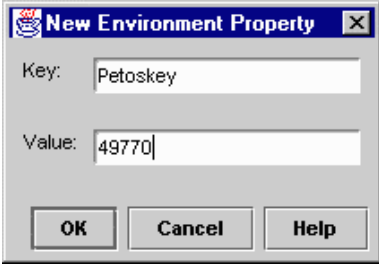
Editing Environment Properties

Click the Environment tab to add, edit and delete user-defined environment properties. Environment properties are elements specific to your application such as sales tax rates or zip codes. These properties are available when your EJB is running in a standard fashion. You can access an EJB's environment properties through the `setEnvironment()` method in `javax.ejb.EJBContext`.

To add a new environment property, perform the following steps:

1. Click the Add New Property button.

   The following dialog box appears:

2. Type the name of the property in the Key field.

   For example, if you are going enter a zip code, enter the city name in the Key column: `Petoskey`.

3. Type the value of the property in the Value field.

   Following the zip code example, enter the zip code corresponding to the city name in the Value column: `49770`.

4. Click OK to dismiss the dialog box.

To delete or edit a property, highlight the property in the table on the Environment tab and click either the Delete Property or Edit Property button.

Editing EJB Properties at Run Time

3

Monitoring Server Activity

This chapter describes the monitoring service provided by Netscape Application Server Administrator. This service allows you to chart various attributes of the Executive, Java, and C++ server processes.

The following topics are included in this chapter:

*   Monitoring Netscape Application Server

*   Receiving Event Notification

Monitoring Netscape Application Server

Netscape Application Server (NAS) Administrator provides a monitoring service that lets you chart the activity of the Executive, Java, and C++ Servers that make up NAS. You can also log the information to a file. By graphically representing this server activity or recording the data in a file, you can track and review the performance of an application server or group of servers and make adjustments to improve performance. For example, if you add more memory to the application server or deploy a new application, you may want to monitor the performance of the application server to see what impact these changes have on it.

Netscape Application Server's monitoring service polls the application server at designated intervals. This saves server resources because the server updates the information being monitored at the interval instead of updating it continuously. You can specify this time interval in the Monitoring window. For information about setting the interval time, see "Changing a Process Data Plot" on page 75.

The monitoring window "pops out" from the administrator tool when you click a process to monitor. This enables you to monitor server activity in a separate window while continuing to perform other administrative tasks using the administrator tool.

Monitoring Process Attributes

The server activity, or attributes, you can chart varies according to which server, or process, you are monitoring.

The Executive Server (KXS) process is responsible for managing and hosting the system-level services, such as the load-balancing service, and for delegating requests to one of the application processes, either the Java Server or the C++ Server, depending on the language in which the application is written.

You can chart the following attributes of the Executive Server process:

Executive Server Process Attribute	Description
CPU load	The amount of load on the CPU on which this Executive Server process is running, as calculated by the load balancing service.
Disk input and output	The rate of Read and Write operations issued by the system on which this Executive Server is running, as calculated by the load balancing service.
Memory thrash	The number of pages read from or written to the hard disk drive to resolve memory references to pages that were not in memory at the time of the reference.
Current requests	Number of requests currently waiting in the queue for processing.
Result cache entries	Number of entries stored in the result cache.
Average request time	Average amount of time for the Executive Server process to reply to a request.

Executive Server Process Attribute	Description
Requests/interval	Number of new requests received since the last polling.
Total requests	Total number of requests the process has received.
Threads	Number of threads being used by the process.
Bytes sent/interval	Number of new bytes sent since the last polling.
Bytes received/interval	Number of new bytes received since the last polling.

Note   If you monitor CPU load, disk input and output, or memory thrash, you must specify the intervals at which the statistics for these process attributes are updated. To set the intervals, select the Load Balancing tab, then click Advanced Settings.

The Java Server (KJS) and C++ Server (KCS) processes are responsible for hosting application elements, depending on the language in which the element is written. The Java Server hosts application components written in Java, and the C++ Server hosts components written in C++.

You can chart the following attributes of the Java and C++ Server processes:

C++/Java Server Process Attribute	Description
Average request time	Average amount of time for the Executive Server process to reply to a request.
Requests/interval	Number of new requests received since within the interval.
Total requests	Total number of requests the process has received.
Active data connections	Number of currently active data connections.
Cached data connections	Number of currently cached data connections.
Queries/interval	Number of queries executed within the interval.
Trans committed/interval	Number of transactions committed within the interval.
Trans rolledback/interval	Number of transactions rolled back within the interval.
Threads	Number of threads being used by the process.

C++/Java Server Process Attribute	Description
Bytes sent/interval	Number of new bytes sent since the last polling.
Bytes received/interval	Number of new bytes received since the last polling.
Session count	Number of existing sessions for a given application.

For each process, you can chart one or more attributes. You can also simultaneously chart the attributes of several application servers, if you have a multiple-server enterprise.

To monitor process attributes, perform the following steps:

1. On the NAS Administrator toolbar, click the Monitor tab to open the Monitor window.

2. In the left pane of the Monitor window, double-click the process whose attribute you want to chart.
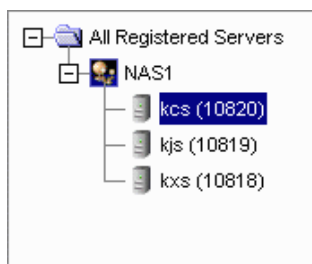
A separate monitoring panel pops out of the NAS Administrator tool:
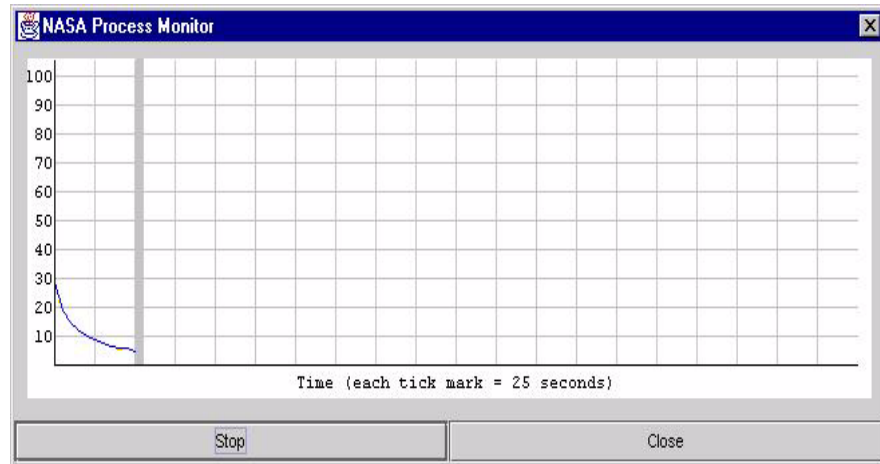
3. In the right pane of the monitoring window in NAS Administrator, click the Add Plot button located at the bottom of the window.

4. In the Attribute column, select the attribute to chart from the Attribute drop-down list.

5. From the Scale drop-down box, choose the scale at which to plot the attribute from the Scale drop-down list.

   Values range from 10:1 to 1:1,000,000.

6. From the Color drop-down box, choose a color to represent the process attribute on the chart from the Color drop-down list.

7. Repeat steps 2 through 6 for each process or attribute you want to chart.

Logging Process Data to a File

Once you begin monitoring a process attribute, you can send data collected by the monitoring service to a file.

To log process data to a file, perform the following steps:

1. On the NAS Administrator toolbar, click the Monitor tab to open the Monitor window.

2. Click the process whose data you want to log as shown in the following figure:

Processes to monitor:

Process	Attribute	Color	Scale
NAS1 - kcs (2:10820)	Average execution time	blue	1:1
NAS1 - kcs (2:10820)	Requests / Interval	cyan	1:1

Time Interval: 5 secs

3. Click the Options button at the bottom of the window.

   The following dialog box appears:
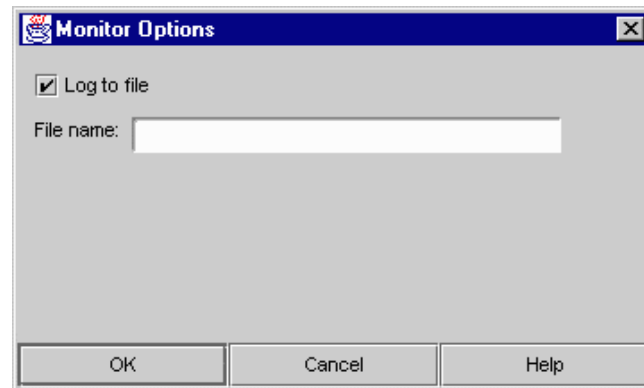
4. Click the Log to File checkbox to enable the logging service.

5. In the File Name text field, enter the name of the file where data is sent.

6. Click OK.

Changing a Process Data Plot

Once a process data plot is added to the Monitor window chart, you can adjust the plot using the Attribute, Color, and Scale drop-down boxes corresponding to each plot to adjust the plot.

To change the attribute of a plot, perform the following steps:

1. On the NAS Administrator toolbar, click the Monitor button to open the Monitor window.

2. Select the process you want to change.
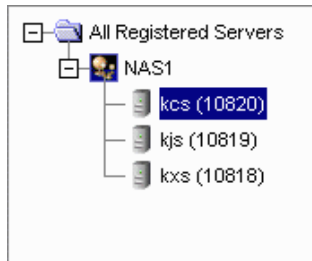
3. In the right pane of the Monitoring window, from the Attribute column, select the new attribute to chart.

4. To change the color or scale of the plot, repeat steps 2 and 3 using the Color and Scale drop-down boxes.

5. To change the interval at which the plot is updated, select a new time from the Interval drop-down box.

Removing a Process Data Plot

Remove a process plot if you no longer want to chart the process attribute.

To remove a process plot, perform the following steps:

1. Click the Monitor button on the NAS Administrator toolbar to open the Monitor window.

2. Select the process you want to remove.

3. Click Remove Plot.

Receiving Event Notification

Event notification is useful when you cannot actively monitor a Netscape Application Server (NAS) machine. This passive monitoring system is activated only in critical circumstances, such as when a process has failed.

You can set the system to alert one or more concerned parties via email when a critical situation arises by supplying the email address(es) of those you want to alert. In addition, you can specify a script that will run automatically when certain events occur.

About Events

You can specify an individual to notify or a script to run for the following critical events:

* Executive Sever (KXS) goes down

* Java Server (KJS) goes down

* C++ Server (KCS) goes down

* Process auto restarts exceeded

What Do I Do When a Server Goes Down?

If one or more of the Executive Server, Java Server, or C++ Server processes go down, the Administrative Server attempts to restart each process. If the process cannot be restarted by the Administrative Server process, the application stops running and can result in lost transactions.

Recurring failures are usually attributed to problems within the application code, but other failures can also happen. Regardless of what causes a process to fail, it is useful to be notified immediately.

If the process restarts, investigate the cause of the failure to determine whether adjustments can be made to prevent future failures. If the process does not restart, look at the log to find the cause of the failure.

What Do I Do When Restarts Are Exceeded?

In addition to process failure events, you can also be notified when the Administrative Server has exceeded the number of times it has attempted to restart a process.

Increase the Administrative Server restart option, if it is low, and determine the source of the process failure.

Configuring Email Notification for an Event

To send an email notification for an event, perform the following steps:

1. On the NAS Administrator toolbar, click the Events button to open the Events window.

2. From the left pane of the Events window, select the server for which you want to configure events.

3. From the right pane of the Events window, select the event or events for which you want to be notified by clicking the corresponding checkbox as shown in the following figure:

4. In the Email Addresses field, specify the email address or addresses of the persons you want to receive notification. Use the following format:

   `chewie@doghouse.com;gracie@meow.org`

5. In the Mail Server field, specify the mail server through which the notification is sent. Use the following format:

   `mail.company.com`

6. To see the most recent events that might have been sent out for this server, click Poll for Events.

   The Poll for Event dialog box appears displaying a list of the recent events for the selected server.

   Note that when you click the Poll for Events button, events are consumed (that is, the events you saw are no longer included in the next set of events that are displayed).

7. Click Apply Changes to save your changes to your application server.

Specifying an Event-Invoked Script

You can configure the event notification service to run a script. The script might page the system administrator, bringing the problem to the administrator's attention, or perform any other automated task that will help keep the system running smoothly when faced with a critical event.

When a script runs, it passes an argument to indicate what type of event has occurred. For instance, the following command indicates that a Java Server (KJS) process has crashed:

```
/script location/ crash kjs
```

To configure the event notification service to run a script in response to an event, perform the following steps:

1. On the NAS Administrator toolbar, click the Events button to open the Events window.

2. From the left pane of the Events window, select the server for which you want to configure events.

3. In the right pane of the Events window, select the event or events for which you want to enable a script by clicking the corresponding checkbox.

4. In the Script field, specify the path of the script to run. For example:

   ```
   /mydir/scripts/myscript.pl
   ```

5. Click Apply Changes to save your changes to your application server.

4

Monitoring NAS with Third-Party Tools

This chapter describes how to monitor Netscape Application Server using the Simple Network Management Protocol (SNMP).

The following topics are included in this chapter:

- About SNMP

- Working with the Master Agent and Subagent

- About the Management Information Base (MIB)

About SNMP

SNMP is a protocol used to exchange data about network activity. With SNMP, data travels between your application server and a workstation where network management software is installed. From this workstation, you can remotely monitor your network and exchange information about network activity between servers. For example, using an application like HP OpenView, you can monitor which Netscape Application Server (NAS) machines are running, as well as the number and type of error messages your application servers receive.

Your network management workstation exchanges information with the application servers in your enterprise through two types of agents: the subagent and the master agent. The subagent gathers information about an application server and passes that information to the master agent. The master agent exchanges information between the various subagents and the network management workstation. The master agent runs on the same host machine as the subagents with which it communicates.

Working with the Master Agent and Subagent

Master agent operation is defined in an agent configuration file called `CONFIG`. You can edit the `CONFIG` file manually.

To configure the master SNMP agent, perform the following steps:

1. Log in as root.

2. Check to see if there is a Solaris SNMP daemon (`snmpd`) running on port 161.

   If an SNMP daemon is running, make sure you know how to restart it and which MIB trees it supports. Then kill its process.

3. Edit the Solaris SNMP daemon start-up file `s76snmpdx` in `/etc/rc3.d` to modify the port to which the daemon listens.

   In the start section, replace the line

   `/usr/lib/snmp/snmpdx -y -c /etc/snmp/conf`

   with

   `/usr/lib/snmp/snmpdx -p 1161 -y -c /etc/snmp/conf`

4. Edit the `CONFIG` file located in `server4/nas/snmp` in the server root directory.

The CONFIG file defines the community and the manager that the master agent will work with. The manager value should be a valid system name or an IP address. The following is an example of a basic CONFIG file:

COMMUNITY   public
             ALLOW ALL OPERATIONS

MANAGER   *your_manager_station_name*
             SEND ALL TRAPS TO PORT 162
             WITH COMMUNITY public

5. (Optional) Define sysContact and SysLocation variables in the CONFIG file.

You can edit the CONFIG file to add initial values for sysContact and sysLocation which specify the sysContact and sysLocation MIB-II variables. Note that the strings for sysContact and sysLocation in this example are enclosed in quotes. Any string that contains spaces, line breaks, tabs, and so on must be in quotes. You can also specify the value in hexadecimal notation.

In this sample CONFIG file, sysContract and sysLocation variables are defined:

```
COMMUNITY    public
             ALLOW ALL OPERATIONS
MANAGER      nms2
             SEND ALL TRAPS TO PORT 162
             WITH COMMUNITY public
INITIAL      sysLocation "Server room 501 East
             Middlefield Road Mountain View,
             CA 94043 USA"
INITIAL      sysContact "John Doe email:
             <jdoe@netscape.com>"
```

The encapsulator forwards requests from the master agent to the Solaris agent that now listens on port 1161.

6. Edit the file CONFIG_SAGT, modifying the following lines:

```
Agent at 1161 with Community Public
```

This configures the subagent to serve the Solaris agent on port 1161.

```
Subtrees <list of oids>
```

This configures the SNMP subtrees served by the Solaris agent.

```
Forward All Traps
```

This ensures that all traps sent by the Solaris agent are forwarded to the master agent.

Starting the SNMP Master Agent

Once you have installed the SNMP master agent, you can start it manually or by using Netscape Console.

To start the master agent manually, enter the following at the command prompt:

```
# magt CONFIG INIT&
```

The `INIT` file is a nonvolatile file that contains information from the MIB-II system group, including system location and contact information. If `INIT` doesn't already exist, starting the master agent for the first time will create it. An invalid manager name in the `CONFIG` file will cause the master agent start up to fail.

Note `INIT` contains information about the local system. This file is created the first time you start the master agent. You should not copy this file across machines.

To automatically start the master agent when you start the server, perform the following steps:

1. Edit the files `nas/snmp/k75snmpmagt` and `nas/snmp/s75snmpmagt`.

2. Change `$GX_ROOTDIR` to the NAS installation directory path if this variable is not yet defined in the root's environment.

3. Copy `k75snmpmagt` in `/etc/rc2.d` and `s75snmpmagt` in `/etc/rc3.d`.

To start a master agent manually on a nonstandard port, use one of two methods:

- Method 1: In the CONFIG file, specify a transport mapping for each interface over which the master agent listens for SNMP requests from managers. Transport mappings allow the master agent to accept connections at the standard port and at a nonstandard port. The master agent can also accept SNMP traffic at a nonstandard port. The maximum number of concurrent SNMP is limited by your target system's limits on the number of open sockets or file descriptors per process. The following is an example of a transport mapping entry:

TRANSPORT	extraordinary SNMP OVER UDP SOCKET AT PORT 11161

After editing the CONFIG file manually, you should start the master agent manually by typing the following at the command prompt:

```
# magt CONFIG INIT&
```

- Method two: Edit the /etc/services file to allow the master agent to accept connections at the standard port as well as at a nonstandard port.

Enabling Statistics Collection

The subagent does not report SNMP statistics to the network management workstation unless you enable statistics collection on the SNMP Settings form, which is part of Netscape Console. If statistics collection is not enabled, the subagent cannot be started.

Note If the network management workstation experiences difficulty obtaining SNMP statistics, check the server log information:

NAS install directory/*mail-instanceName*/log/default

If the SNMP data collection process (snmpcoll), is not running, check the Administration Server Console to see whether the SNMP enable flag is on. For more information, see *Managing Servers with Netscape Console* on the Netscape documentation web site (http://home.netscape.com/eng/server/console/).

If you disable the start-up server, this collection process is also disabled.

To enable data collection, perform the following steps:

1. Check the Enable Statistics Collection box.

   If you remove the check, the subagent cannot be enabled.

2. Restart the subagent by clicking the Start button.

Your configuration information is stored in Directory Server, the subagent starts, and statistics collection begins.

About the Management Information Base (MIB)

Netscape Application Server (NAS) stores variables pertaining to network management in a tree-like hierarchy known as the server's management information base (MIB). NAS reports significant events to the network management workstation by sending messages containing these variables. The network management workstation can also query the server's MIB for data or can remotely change variables stored in the MIB.

You can find the NAS MIB in the following location:

NAS install directory\plugins\snmp\

Formatting MIB Entries

The MIB file contains the definitions for managed objects, or variables, that store network information for the server. Each variable definition includes the variable name, its data type and read/write access level, a brief description, and a permanent object identifier.

This sample entry shows the definition for the `nsmailEntityDescr` variable:

```
nasKesMaxThread    OBJECT-TYPE     / object type
               SYNTAX      INTEGER (SIZE (1..512))      / syntax
             ACCESS      read-write     / read/write access level
             STATUS      mandatory     / status
             DESCRIPTION               / description
             "The maximum number of threads used to serve requests."
             ::= { kes 4 }  / object identifier
```

This definition contains the following information:

- Object Type: gives the name of the variable, in this case, `nasKesMaxThread`.

- Syntax: gives the abstract data type of the variable object type in ASN.1 notation. For example, the Syntax of the `nasKesMaxThread` variable is `INTEGER (SIZE (1..512))`.

- Access: gives the read/write access level to the variable. Possible access levels are read-only, read-write, write-only, or not-accessible.

- Status: tells whether the element is mandatory, optional, or obsolete.

- Description: text description of the element, enclosed in quotes. For example, the description of the `nasKesMaxThread` variable is "The maximum number of threads used to serve requests."

- Object Identifier: assigned name that serves as a permanent identifier for each managed object in the MIB name tree in its name space. Objects in SNMP are hierarchical; the object identifier is a sequence of labels that represents the object in the hierarchy. For example, `nasKesMaxThread` is identified as `kes 4`. This means that it has the label 4 in the subtree `kes`.

  `kes`, in turn, has the label 4 in the `kesTable` subtree.

About the Management Information Base (MIB)

5

Logging Server Messages

This chapter describes the message-logging service provided by Netscape Application Server.

The following topics are included in this chapter:

*   About the Logging Service

*   About Web Server Requests

About the Logging Service

You can log server messages using the Netscape Application Server (NAS) message-logging service. The logging service is configured through the NAS Administrator Logging window. There you can specify the destination and types of messages logged.

When you enable logging, NAS records messages generated by NAS application-level and system-level services. These messages describe the events that occur while a service is running. For example, each time NAS communicates with the database, the logging service records the resulting messages generated by database access service.

Determining Types of Messages to Log

You can log any of the three types of messages generated by NAS services. Each type is described in the following table:

Message type	Description	When it might appear
Information message	Describes the processing of a request or normal service activity, such as a status update.	When no problems arise.
Warning message	Describes a noncritical problem that might be an indication to a larger problem.	When a service encounters a temporary problem, such as when it is unable to connect to a process.
Error message	Describes a critical failure of the service, from which recovery is not likely.	When a service encounters a critical problem, such as a pipe closure.

With the logging service, you can record error messages, error and warning messages, or all messages. To choose which type of messages to log, perform the following steps:

1. Click the Logging button on the NAS Administrator toolbar to open the Logging window.

2. Select the Enable Server Event Log checkbox as shown in the following figure:

3. In the General area, from the Message Type drop-down box, select Errors, Errors and Warnings, or All Messages.

4. In the Maximum Entries text field, enter the maximum number of entries that can exist before data is written to the log.

5. In the Write Interval text field, enter the amount of time (in seconds) that elapses before data is written to the log.

Logging Application Messages

Message logging is also useful for tracking and debugging application errors. By using the `log( )` method, application developers can send messages to the same log destination the server administrator configures for NAS services.

For example, if an application encounters a problem in a segment of code, you can log the associated error message. Informational messages about the application's status, rather than error messages, are also useful.

How Log Messages Are Formatted

Every log message has the following four components:

*   date and time the message was created

*   message type, such as information, warning, or error

*   service or application component ID

*   message text

Two of these components, the message type and the service or application component ID, are stored as numbers in a database table. The logging service maps those numbers to a string when sending the message to the other three destination logs.

A log message uses the following syntax when it is sent to a process console, an ASCII text file, or the application log:

```
[Date and time of message] Message type: Service ID: Message text
```

For example, the following messages sent to an ASCII text file illustrate message format:

```
[11/18/97 11:11:12:0] info (1): GMS-017: server shutdown (host
0xc0a801ae, port 10818, group 'NAS') - updated host database

[11/18/97 11:11:18:2] warning (1): GMS-019: duplicate server (host
0xc0a8017f, port 10818) recognized, please contact Netscape
Communications for additional licenses
```

Determining the Logging Destination

You can configure the logging service to record server and application
messages in any or all of the destinations described in the following table:

Log destination	Description	When to use
Process consoles	The NAS process consoles display log messages as they are generated.	This is the default. If logging is enabled and the server is enabled for automatic startup (UNIX) or interaction with the desktop (NT), the consoles open and display the log messages. You can disable this feature by deselecting the Log to Console checkbox.
Application log	The default application log file. For Windows NT, this is viewable through the Event Viewer.	This is the default. Provides a more comprehensive record of the server and application error messages. Warning and information messages are not logged to the application log. All messages are sorted by their timestamp.
ASCII text file	An ASCII text file, which you must create and specify.	Use when you want a more permanent record of the server and application messages. All messages are sorted by their timestamp.
Database table	A database table which you must create and specify.	This is the most versatile logging destination. Use when you want to sort, group, and create reports of the logged messages.

When you enable logging, the logging service automatically sends messages to the process consoles on Windows NT and Unix platforms, as long as those consoles are open and console logging in enabled. On Windows NT, the logging service also sends messages to the application log. Logging to a process console does not record the messages. You cannot retrieve the messages once they scroll off of the screen.

To enable the logging service and specify the destination of the log messages, perform the following steps:

1. Click the Logging button on the NAS Administrator toolbar to open the Logging window.

2. Select the Enable Server Event Log checkbox.

3. In the Log Target box, choose the type of logging to enable by clicking the Log to a Database, Log to Windows NT Application Log (Errors Only), and/or Log to File checkbox(es). You can disable console logging by deselecting the Log to Console checkbox.

See "Logging to a Database" in the following section and "Logging to a File" on page 97 for more information.

If you chose to log to a file, that file is created now. See "Rotating Log Files" on page 97 for information about managing log files.

NAS uses a log buffer to store messages before they are written to the application log, an ASCII file, and/or database logs. This buffer optimizes the performance of the application server by limiting the use of resources to continually update a log. The buffer is written to the destination when either the buffer interval times out or the number of entries in the buffer exceeds the maximum number allowed.

Logging to a Database

If you plan to log application server messages to a database, you need to create an event log database table. The following table describes the four field names and lists each field's data type.

Note On a UNIX system, you can use supplied scripts that automatically set up the eventlog and httplog tables. The scripts are located in the directory `$GX_ROOTDIR/APPS/GXApp/Logging/db`, and are named `Log_db2.sql`, `Log_ifmx.sql`, `Log_mssql.sql`, `Log_ora.sql`, and `Log_syb.sql`. Choose the script that is appropriate for the database you're using.

Database field name	Description	Data type
evttime	Date and time the message was created	Date/Time
evttype	Message type, such as information, warning, or error	Number
evtcategory	Service or application component ID	Number
evtstring	Message text	Text

The logging service maps the message components to the database fields listed in the table. You must use these exact field or column names in your database table.

To log to database, perform following steps:

1. Click the Logging button on the NAS Administrator toolbar to open the Logging window.

2. Select the Enable Server Event Log checkbox as shown in the following figure:

3. In the General area, from the Message Type drop-down box, select Errors, Errors and Warnings, or All Messages.

4. In the Log Target area, click the Log to Database checkbox.

   Enter the data source, the database name, the table name, and the user name and password necessary for accessing the database.

5. Click the Apply Changes button to save your changes to NAS Administrator.

Logging to a File

NAS Administrator's monitoring service allows you to log information about server activity to a file.

To log information to a file, perform the following steps:

1. Click the Logging tab on the NAS Administrator toolbar to open the Logging window.

2. Select the Enable Server Event Log checkbox.

3. In the General area, from the Message Type drop-down box, select Errors, Errors and Warnings, or All Messages.

4. In the Log Target area, select the Log to File checkbox.

5. In the Log to File text field, enter the name of the log file.

6. Click Apply Changes to save your changes to NAS Administrator.

Rotating Log Files

If you choose to record server messages in an ASCII file, you can enable log file rotation to regulate when log files are rotated. Since log files are stamped with the time and date they are created, log file rotation helps organize log files into manageable units.

To configure log file rotation, perform the following steps:

1. Click the Logging button on the NAS Administrator toolbar to open the Logging window.

2. Select the Enable Server Event Log checkbox.

3. In the General area, from the Message Type drop-down box, select Errors, Errors and Warnings, or All Messages.

4. Click the Log to File checkbox.

5. In the Enable File Rotation drop-down box, choose Yes.

6. From the Rotation Interval drop-down box, select the interval at which log files are rotated or enter a string to indicate when the log file is rotated.

   For instance, the following string indicates logging to a new file begins at 1:00 AM every Monday, as well as on the fifteenth of each month:

   ```
   1:0:0 1/15/*
   ```

The following string indicates logging to a new file begins at 2:00 AM, 5:00 AM, 6:00 AM, and 7 AM every Friday:

```
1, 5 – 7:0:05/*/*
```

7. Click Apply Changes to save your changes to the NAS Administrator.

About Web Server Requests

You can use the Netscape Application Server (NAS) logging service to log web server requests. Web server requests are monitored by the web connector plug-in. The plug-in sends requests to your NAS machine where they are processed. By logging web server requests, you can track request patterns and other important request information.

How Web Requests Are Logged

A web server request is divided into components. These components are standardized HTTP variables used by the web server to manage web requests. NAS includes a subset of these HTTP variables for you to log. You can add variables to the list if you need to log additional information.

Note   On a UNIX system, you can use supplied scripts that automatically set up the HTTP log and event log tables. See "Logging to a Database" on page 95 for more information.

Each HTTP variable must be mapped to a database field name within a table that you create. For instance, to log the length of the content of a web server request, map the CONTENT_LENGTH variable to a database field named, for example, content_length and defined as a text data type. The default HTTP variables used by NAS and their database data types are listed in the following table. Use this table to help you create the database table for logging web requests.

Default HTTP variables	Default database field name	Data type
Not applicable	logtime	Date/Time
CONTENT_LENGTH	content_length	Number

Default HTTP variables	Default database field name	Data type
CONTENT_TYPE	content_type	Text
HTTP_ACCEPT	accept	Text
HTTP_CONNECTION	connection	Text
HTTP_HOST	host	Text
HTTP_REFERER	referer	Text
HTTP_USER_AGENT	user_agent	Text
PATH_INFO	uri	Text
REMOTE_ADDR	ip	Text
REQUEST_METHOD	method	Text
SERVER_PROTOCOL	protocol	Text

You must have a field name called logtime in the database table. The time the message is created is assigned by the logging service. The logging service maps that time to the logtime database field. You can rename all of the other database field names.

The fields from the database table are automatically mapped to web server variables in the registry.

You must have a web server communication plug-in module such as NSAPI or ISAPI installed and properly configured. Even though this happens automatically during installation, there may be occasions when you must manually configure the web server.

Logging Web Server Requests

Before you can log web server requests, you must create a database table to hold the request messages. For more information about creating this table, see "How Web Requests Are Logged" on page 99.
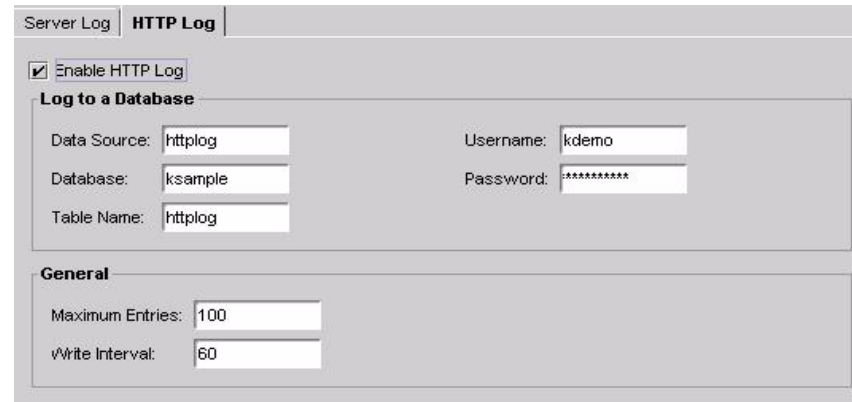
To log web server requests, perform the following steps:

1. Click the Logging button on the NAS Administrator toolbar to open the Logging window.

2. From the left pane of the Logging window, select the application server responsible for logging web server requests.

   If you have more than one application server, you can specify one server to log web server requests.

3. In the right pane of the logging window, click the HTTP Log tab.

   The following window appears:

4. Enter `httplog` in the Data Source field.

5. Enter the information you use to connect to the database in the Database field. For example, this would be the Oracle SID for an Oracle database.

6. In the Table Name field, enter `httplog`.

7. Enter the user name and passwords with which you connect to the database.Enter the maximum entries.

   This number represents the greatest number of entries that can exist before data is written to the log.

8. Enter the write interval.

   This number represents the amount of time that lapses before data is written to the log.

9. To enable database logging of web server requests, select Log to a Database.

10. Click Apply Settings to save your changes to your application server.

6

Securing Applications

This chapter describes how to implement Netscape Application Server security.

The following topics are included in this chapter:

*   About Security

*   Storing and Managing Users and Groups

*   Setting Access Control List Authorization

About Security

Implementing application security is a joint effort between the application developers and the server administrator: the application developers are responsible for determining what level of security to implement and implementing that level into their applications; the administrator is responsible for managing the users and groups who use the application, as well as access control lists.

This chapter explains how to set up users and groups, the type of security each provides, and how they are used with access control lists. It also describes how user entries are stored in Netscape Directory Server and managed using Netscape Console and LDIF. Access control lists are stored locally on each server machine and are managed using the NAS Administrator tool.

Limitations of This Document

This chapter does not explain Directory Server and Netscape Console in great detail. Rather, it provides descriptions of the basic start-up tasks you must perform when setting up Directory Server in association with your instance of NAS, as well as how to use Netscape Console to manage users and groups. See Netscape Directory Server and Netscape Console documentation for detailed instructions and descriptions of these products.

You can find Directory Server documentation installed with your instance of NAS in the following location:

`NAS install directory/manual/en/slapd/`

Netscape Console documentation is available on Netscape's web site in the following location:

`http://home.netscape.com/eng/server/console/`

What Is LDAP?

Every instance of Netscape Application Server (NAS) uses Directory Server to store shared server information, including information about users and groups. Directory Server supports Lightweight Directory Access Protocol (LDAP) versions 2 and 3. LDAP is an open directory access protocol that runs over TCP/IP. It is scalable to a global size and millions of entries. Using Directory Server, you can store all of your enterprise's information in a single, centralized repository of directory information that any application server can access via the network.

Netscape Directory Server is installed with each instance of NAS.

What Is Netscape Console?

Netscape Console is a stand-alone Java application. It finds all resources and applications registered in Directory Server, and displays them in a graphical interface. Netscape Console functions independently of any server, and you can use it from any computer or workstation connected to your enterprise.

Netscape Console is installed with each instance of NAS. You use Netscape Console to manage users and groups for NAS. You can also use Netscape Console to launch the NAS Administrator tool, but only for local instances of NAS -- that is, instances of NAS installed on the same machine as Netscape Console. You must launch remote instances of NAS from the command line or from the Windows NT start menu.

Storing and Managing Users and Groups

The information you specify for each entry you create is stored in the Directory Server used with your instance of Netscape Application Server (NAS). The information held in Directory Server is shared between all application servers when you have multiple servers supporting an application

Implementing User-Based Security

User-based security allows access to an application by authenticating a user's user name and password. The user name and password of any user who requires access to the application must be stored in Directory Server.

An application starts the user authentication process by calling the application component—usually a servlet—responsible for user authentication. The user's login privileges are then verified against the list of users stored in Directory Server.

Once a user is successfully authenticated, access to specific application components is managed programmatically using access control lists and application components responsible for application security.

User security verifies access to an application based on a user's name and password. To implement user security, you must create a user profile, which holds the user name and password, for all users of an application. This procedure is described in "Using Netscape Console to Add Entries to Directory Server" in the next section.

Using Netscape Console to Add Entries to Directory Server

You can use Netscape Console to create user entries and group entries. A user entry contains information about an individual person or object in the directory. A group consists of all users who share a common attribute. For example, all users in a particular department might belong to the same group.

What Is a Distinguished Name (DN)?

Each of the users and groups in your enterprise is represented in Directory Server by a distinguished name (DN). A DN is a text string that contains identifying attributes. You use DNs whenever you make changes in the directory's users and groups database. For example, you need to specify DN information each time you create or modify directory entries, set up access controls, and set up user accounts for applications such as mail or publishing. The users and groups interface of Netscape Console helps you create or modify DNs.

For example, this might be a typical DN for an employee of Netscape Communications Corporation:

```
uid=doe,e=doe@netscape.com,cn=John Doe,o=Netscape Communications
Corp.,c=US
```

The abbreviations before each equal sign in this example have the following meanings:

- `uid`: user ID

- `e`: email address

- `cn`: the user's common name

- `o`: organization

- `c`: country

DNs may include a variety of name-value pairs. They are used to identify both certificate subjects and entries in directories that support LDAP.

Creating User Entries Using Netscape Console

User security is best suited for applications that have a small number of known users. You must create a user profile for each user who accesses the application.

You must be a Directory Server administrator or a user with the necessary permissions to create a user.

To create a new user entry in the directory using Netscape Console, perform the following steps:

1. From the Windows Start menu, under Programs, choose Netscape Server Family, then Netscape Console 4.0 to open Netscape Console.

   For Unix, in the server root, enter `./startconsole`.

   The Netscape Console login dialog box appears:
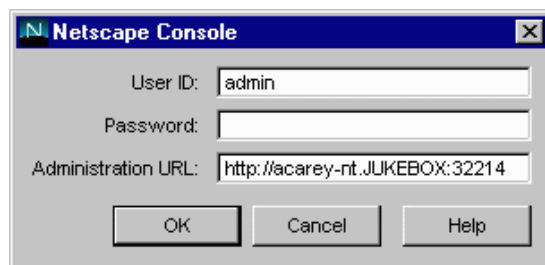
2. Enter a valid user name and password and click OK.

Netscape Console's main window appears:
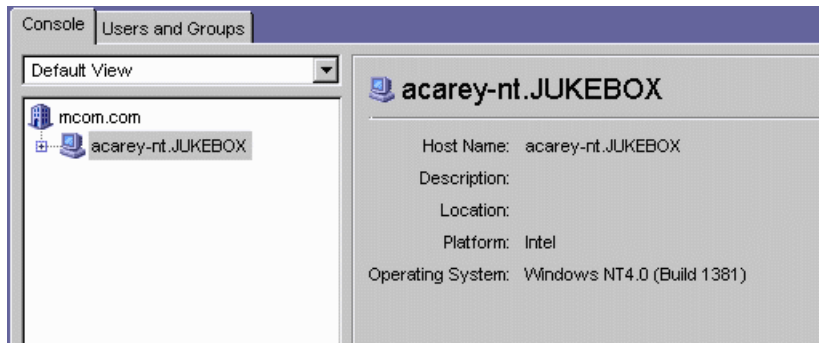
3. Click the Users and Groups tab.

The following window appears:
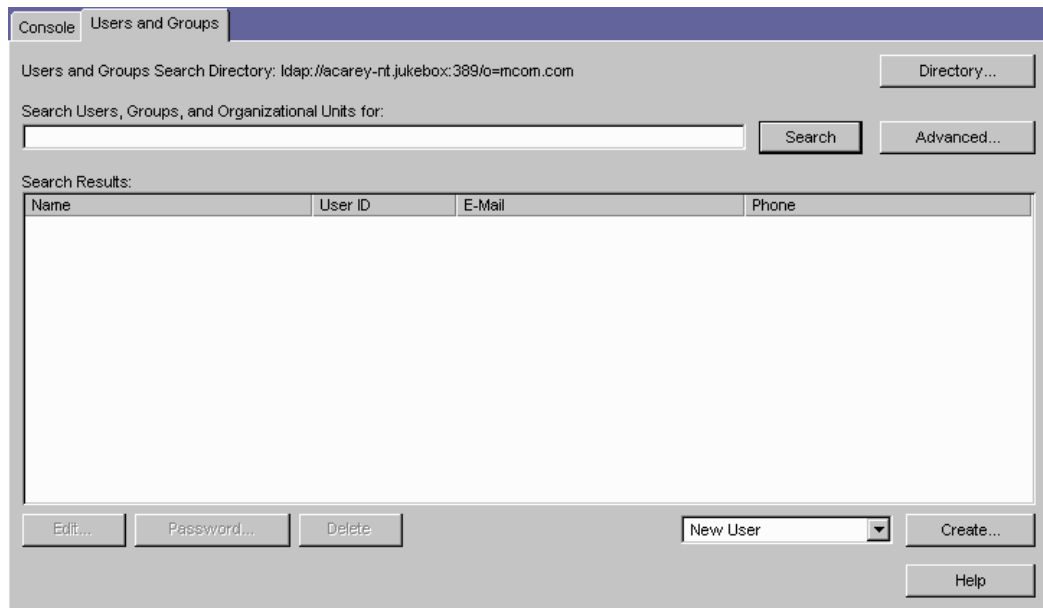
4. Use the drop-down list in the lower-right corner of the window to choose New User, then click Create.

The Select Organizational Unit dialog box appears:

5. In Select Organizational Unit, click the directory subtree (ou) to which the user will belong, then click OK.

The Create User window appears:

6. In the Create User window, enter user information.

- Full Name(s) is equivalent to the common name (cn) in the directory and is automatically generated based on the First Name and Last Name entered above. You can edit this name as necessary.

- A user ID is automatically generated from the first and last names you enter. You can replace this user ID with one of your choosing. The user ID must be unique from all other user IDs in the directory.

7. Click the Licenses tab.

The following window appears:
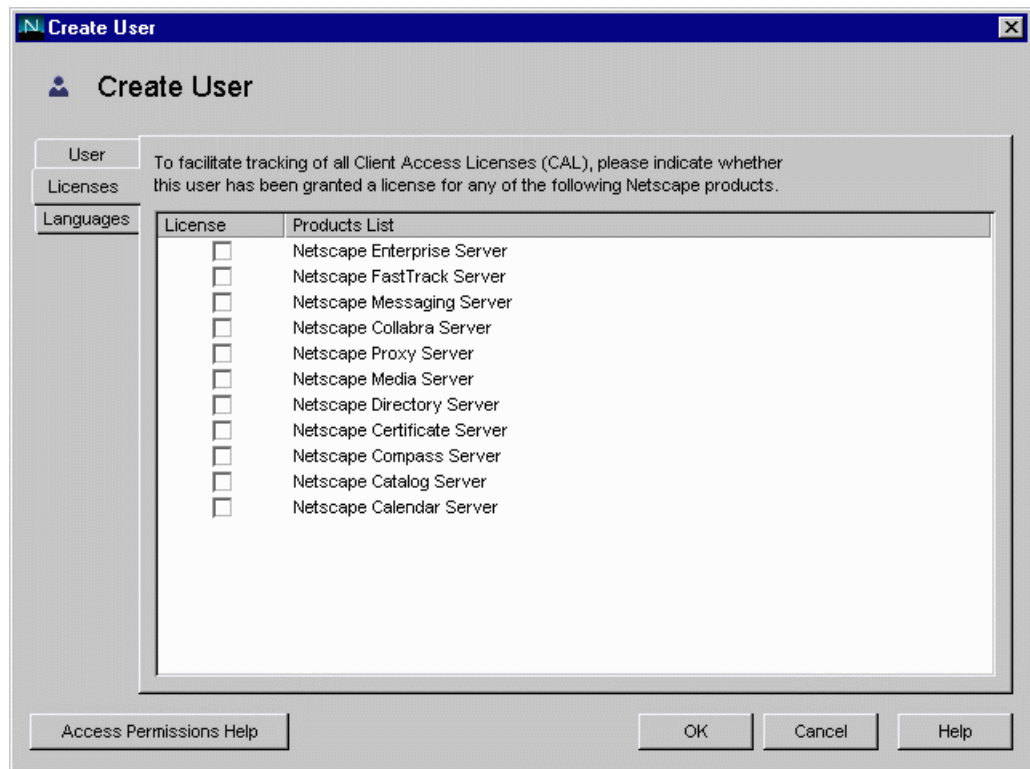
8. Select the servers this user is licensed to use, then click OK.

9. (Optional) Click the Languages tab.

The following window appears:
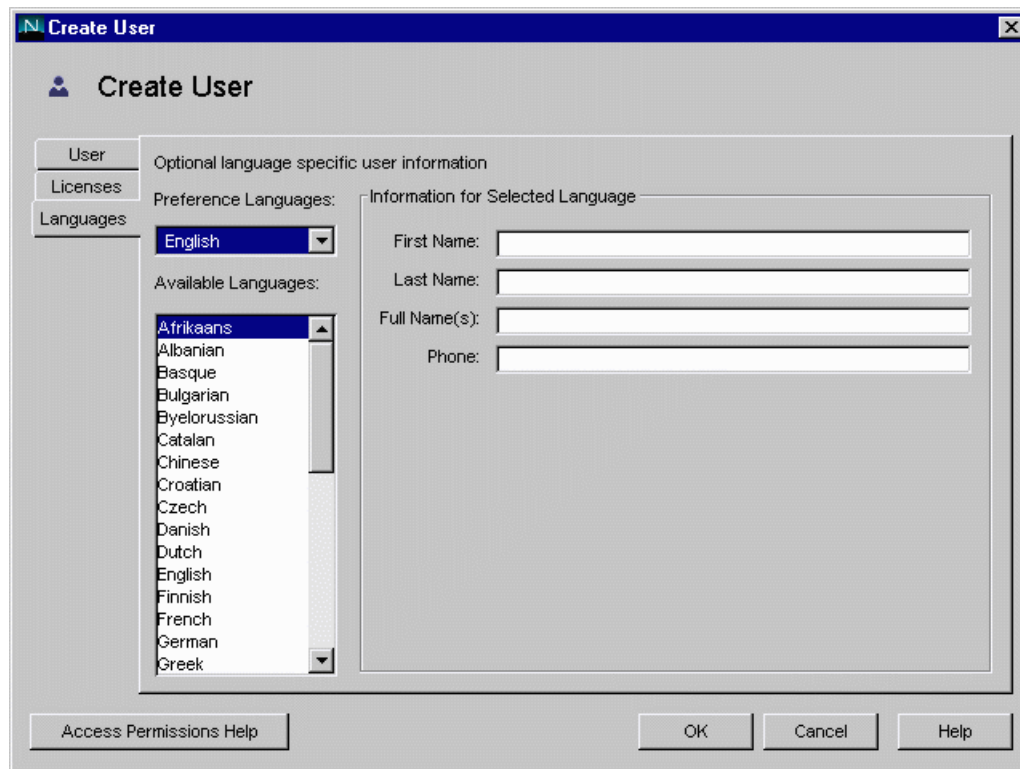
- Use the Preference Languages drop-down list to select the user's preferred language. Select a language to see the Pronunciation field when appropriate.

- Enter language-related information.

Creating Group Entries Using Netscape Console

A group consists of all users who share a common attribute. For example, all users with DNs containing the attribute `ou=Sales` belong to the Sales group. Once you create a new group, you add users, or members, to it. You can use three types of groups in your directory: static, dynamic, and certificate groups.

Creating a Static Group

Create a static group by specifying the same group attribute in the DNs of any number of users. A static group doesn't change unless you add a user to it or delete a user from it. For example, a number of users have the attribute `department=marketing` in their DN. None of those users are members of the Marketing group until you explicitly add each one to the group.

To create a static group in the directory, perform the following steps:

1. In Netscape Console, click the Users and Groups tab to display the following window:

2. Use the drop-down list in the lower-right corner of the window to choose New Group, then click Create.

The following dialog box appears:

3. In the Select Organizational Unit window, select the directory subtree (ou) to which the group will belong, then click OK.
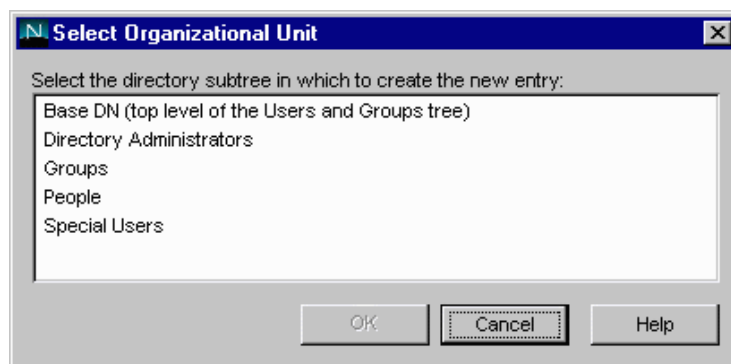
The Create Group window appears:

4. In the Create Group window, enter group information, then click the Members tab.

The following window appears:

5. If you only want to create the group now and plan to add group members later, click OK and skip the rest of this procedure.

To immediately add members to the group, continue to the next step.

6. In the Members window, click Add or Edit as appropriate.

   The following dialog box appears:
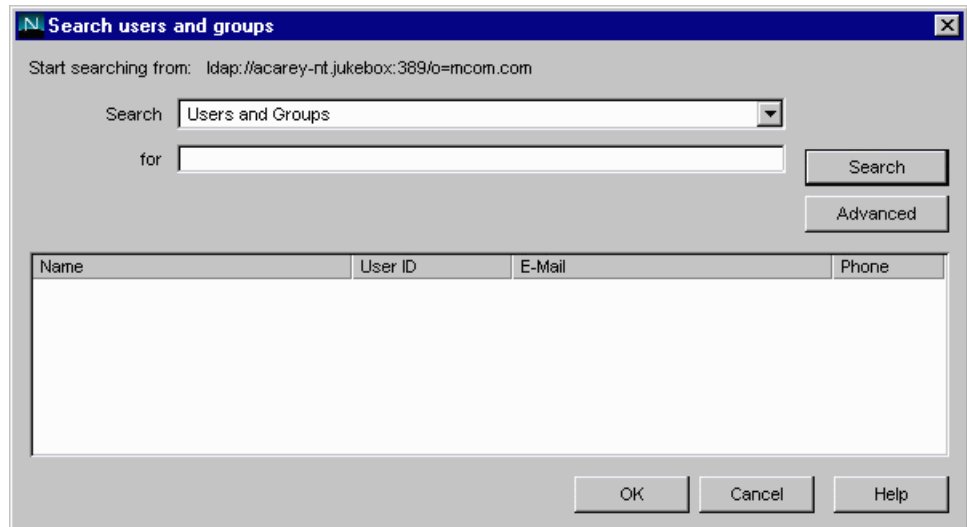
   Use the Search dialog box to locate a user you want to add to the Members
   User ID list. Repeat this step until all the users you want to add to the group
   are displayed in the Member User ID list.

Creating a Dynamic Group

Create a dynamic group when you want users to be added automatically to a
group based on their DN attributes. For example, you can create a group that
automatically includes any DN that contains the attribute
`department=marketing`. Whenever you apply a search filter for
`department=marketing`, the search returns a group including all DNs
containing that attribute. The DNs are included automatically; you do not have
to add each individual to the group.

To create a dynamic group in the directory, perform the following steps:

1. In Netscape Console, click the Users and Groups tab to display the following window:

2. Use the drop-down list in the lower-right corner of the window to choose New Group, then click Create.

   The following dialog box appears:

3. In the Select Organizational Unit window, select the directory subtree (ou) to which the group will belong, then click OK.

The Create Group window appears:

4. In the Create Group window, enter group information, then click the Members tab.

The following window appears:

5. Click Dynamic Group, then click Add.

The following dialog box appears:
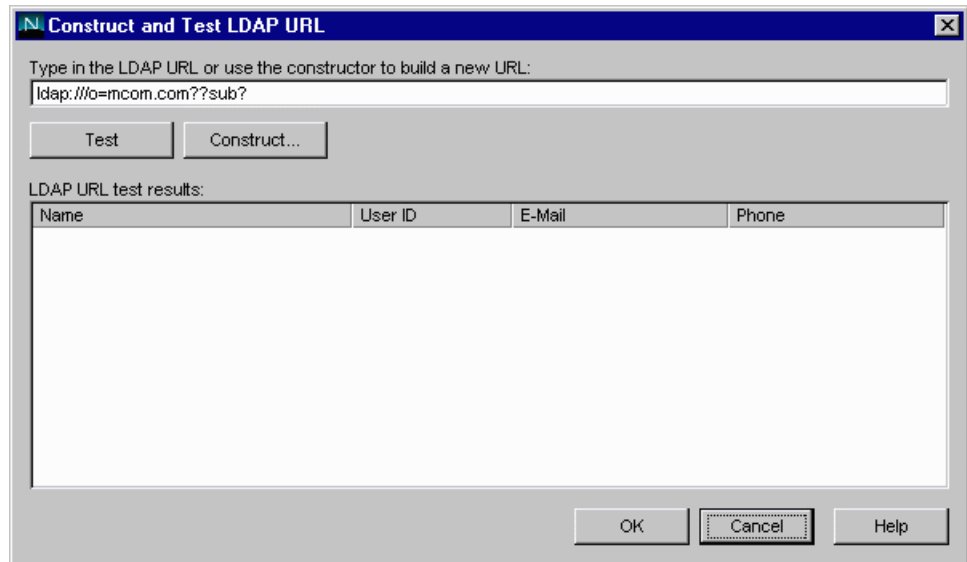
6. Use the Construct and Test LDAP URL dialog box to specify the criteria for including users in the dynamic group.

Choose one of the following:

- Enter an LDAP URL and skip to step 7. The LDAP URL will take the form:

  ```
  ldap:///o=mcom.com??sub?(department=marketing)
  ```

- Click Construct to build a new URL.

The following dialog box appears:
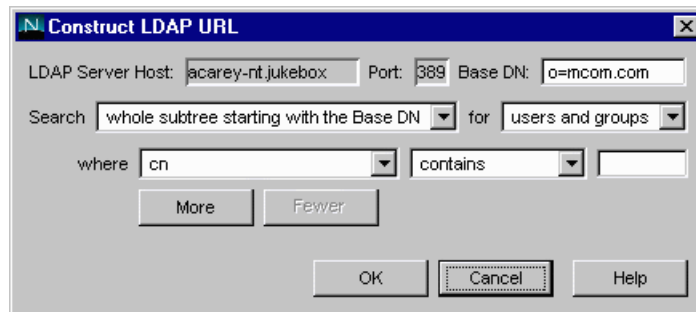
- In the Construct LDAP URL dialog box, provide search criteria:

LDAP Server Host: Enter the fully qualified host name of the user directory you want to search. For example:

    <host>:<domain>

Port: Enter port number for the Directory Server instance that contains the specified user directory.

Base DN: Enter the base DN from which to begin the search. For example: `ou=Marketing, o=Klondike Corp, c=US`

Search: Indicate the user directory subtree you want to search against.

7. Click OK.

8. (Optional) In the Construct and Test LDAP URL dialog box, to see a list of users and groups included in the dynamic group, click Test.

To accept the URL and add it to the list of dynamic group members, click OK.

9. Click OK.

Modifying Database Entries Using Netscape Console

Before you can modify user or group data, you must first use the Users and Groups Search function to locate the user or group entry in the user directory. Then you can select operations from the menu bar to change the entry. The operations you perform apply to all in the Search list.

See Netscape Console documentation for more information.

Using LDIF to Add Entries to Directory Server

You can add entries to Directory Server using LDIF or Netscape Console. Netscape Console is described "Using Netscape Console to Add Entries to Directory Server" on page 106.

Directory Server uses the LDAP Data Interchange Format (LDIF) to describe a directory and directory entries in text format. LDIF is commonly used to initially build a directory database or to add large numbers of entries to the directory all at once. You can also add or edit entries using the ldapmodify command along with the appropriate LDIF update statements.

To add entries to the database using LDIF, first define the entries in an LDIF file, then import the LDIF file from Directory Server.

Formatting LDIF Entries

LDIF consists of one or more directory entries separated by a blank line. Each LDIF entry consists of an optional entry ID, a required distinguished name, one or more object classes, and multiple attribute definitions.

The basic form of a directory entry represented in LDIF is:

```
dn: distinguished name
objectClass: object class
objectClass: object class
...
attribute type[;subtype]:attribute value
attribute type[;subtype]:attribute value
...
```

You must supply the DN and at least one object class definition. In addition, you must include any attributes required by the object classes that you define for the entry. All other attributes and object classes are optional. You can specify object classes and attributes in any order. The space after the colon is also optional. For information on standard object classes and attributes, refer to the *Netscape Directory Server Schema Reference Guide*.

Modifying Database Entries Using ldapmodify

You use the `ldapmodify` command-line utility to modify entries in an existing Directory Server database. `ldapmodify` opens a connection to the specified server using the distinguished name and password you supply, and modifies the entries based on LDIF update statements contained in a specified file. Because `ldapmodify` uses LDIF update statements, `ldapmodify` can do everything that `ldapdelete` can do. Most of Directory Server's command-line utilities are stored in a single location. You can find them in the following directory:

NAS install directory/bin/slapd/server

The remaining three—`ldapdelete`, `ldapmodify`, and `ldapsearch`—are stored in the following directory:

NAS install directory/shared/bin

The following is an example of the command used to add a user to an LDIF file:

```
ldapmodify -h myserverhost -p 389 -D "Directory Manager" -w admin -a -f
MyUsersFile
```

Creating Entries Programmatically

You can also create entries progammatically within an application using the LDAP JDK included with each installation of NAS. See the *Programmer's Guide* for more information.

Setting Access Control List Authorization

Access control lists (ACLs) allow you to set permissions for users and groups. A permission relates to an action the user is allowed to perform, such as read or write.

Netscape Application Server (NAS) comes with default permissions, but you can also create your own application-specific permissions and ACLs. The information in an ACL is used by the application to verify the permissions of the current user or group for an action the user attempts.

If a user does not have a certain permission, the application can direct the user to re-login, prompt him to exit the application, or direct him to a different part of the application.

Creating an Access Control List

You use NAS Administrator to create and manage access control lists (ACLs). When creating an ACL, you can create groups to which users belong and add only groups to the ACL rather than adding individual users as members to the ACL. This is useful if you are using individual user-based security; you save the administration maintenance of updating users in the ACL when users change.

For example, if you have created users for an intranet application and a user leaves the company, you need to remove that user only from the appropriate group or groups, as opposed to removing the user from the groups and any ACLs.

To create an access control list, perform the following steps:

1. On the NAS Administrator toolbar, click the Security button to open the Security window of NAS Administrator.

The following window appears:
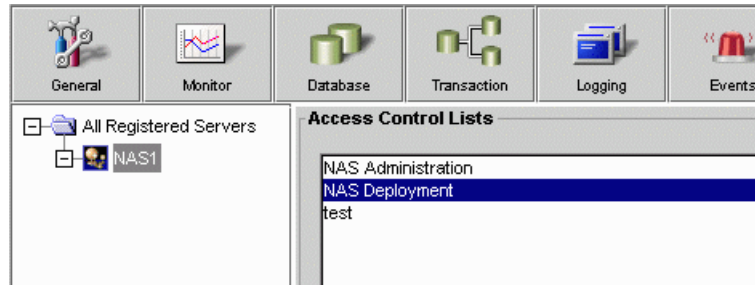
2. Click the New button located at the bottom of the window.

    The New Access Control List dialog box appears.
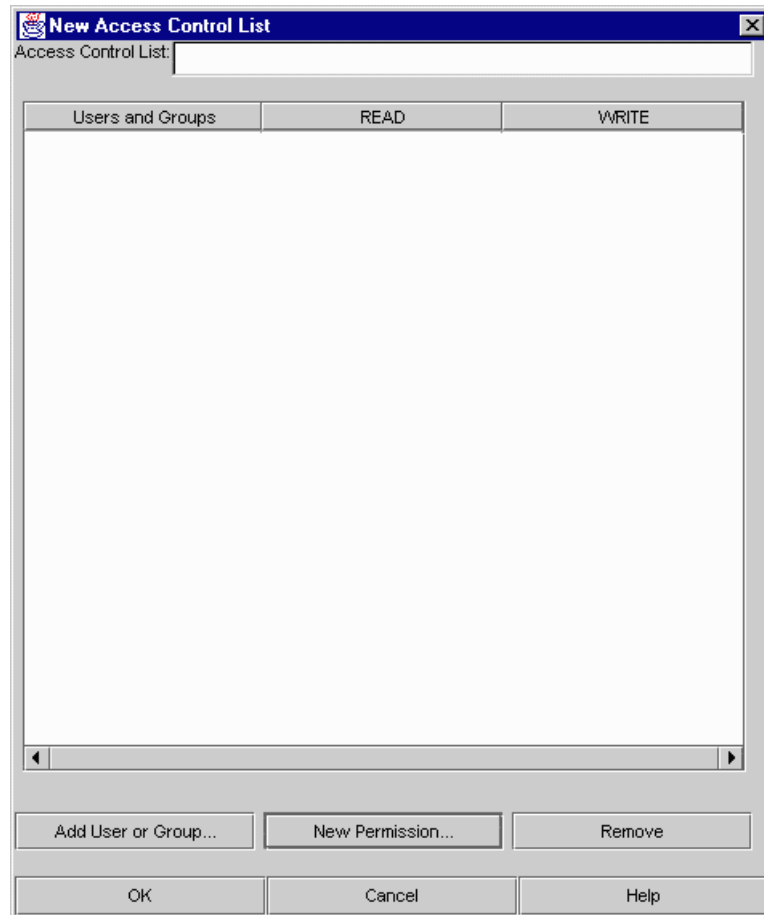
3. In the Access Control List field, enter a name for the ACL.

   The name can be any word or words you choose to distinguish one ACL from another.

4. To add a user or group to the ACL, click the Add User or Group button at the bottom of the dialog box.

   The Add User or Group dialog box appears.

5. Select the users and/or groups you want to add to the ACL.

   You can filter the list of users that appears in the result set by entering a string in the User Filter text box. For instance, to show only user IDs that begin with "F," enter F* in the User Filter text box, then click the User Filter button. The user IDs matching your filter criteria appear in the list box below. The User Filter applies only to users, not to groups.

6. Click OK.

7. To add a new permission to the ACL, click New Permission.

   The New Permission dialog box appears.
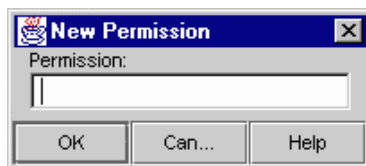
8. Enter the new permission action word.

   A permission defines the level of access a user or group has to a particular application or part of an application.

9. Click OK.

10. To set the appropriate permissions for the groups in the ACL, check each permission for that group.

Modifying an Access Control List

You can modify the following ACL properties:

- add groups

- create new permissions

- edit permissions

You can also remove groups from the system.

To modify an access control list, perform the following steps:

1. On the NAS Administrator toolbar, click the Security button to open the Security window of NAS Administrator.

   The following window appears:
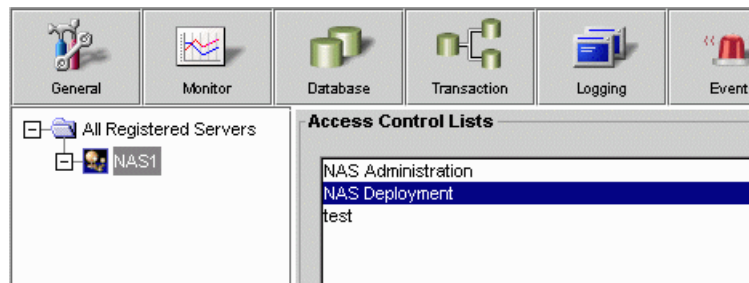
2. Click the Modify button located at the bottom of the window.

   The Modify Access Control List dialog box appears.
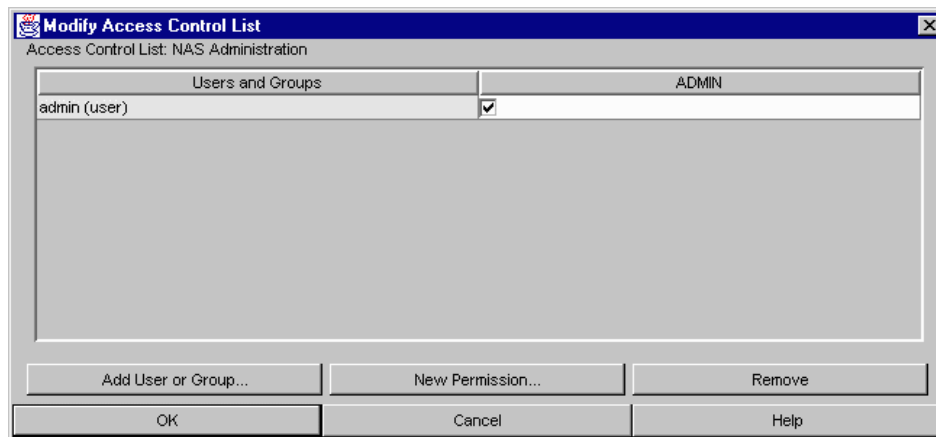
3. To add a new user or group, click Add User or Group.

   The Add User or Group dialog box appears.
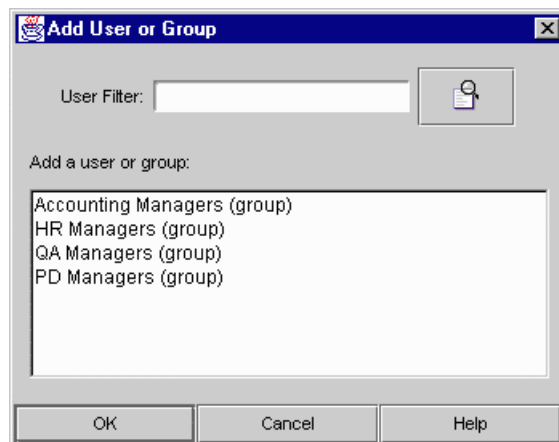
4. Select the group or groups you want to add to the ACL.

You can filter the list of users that appear in the list by entering a string in the User Filter text box. For instance, to show only user IDs that begin with "F," enter F* in the User Filter text box, then click the User Filter button. The user IDs matching your filter criteria appear in the list box below. The User Filter applies only to users, not to groups.

5. Click OK.

6. To create a new permission, click New Permission.

The New Permission dialog box appears.
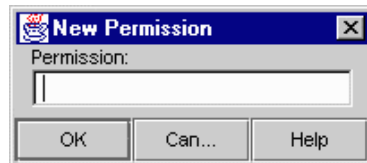
7. To edit the permissions of a group, select or deselect the appropriate permissions for that group.

8. To remove a group, select that group and click Remove.

Setting Access Control List Authorization

Increasing Fault Tolerance and Server Resources

This chapter describes increasing Netscape Application Server resources, which can increase application performance.

The following topics are included in this chapter:

- Adding and Tuning Java Server and C++ Server Processes

- Adjusting the Number of Threads for a Process

- Adjusting the Restart Option of the Administrative Server

- Implementing a Multi-Process, Single-Threaded Environment

- Configuring Directory Server Failover

Increasing Netscape Application Server (NAS) resources, such as number of threads, number of processes, and number of restart attempts can increase the performance of the applications running on the server and reduce the likelihood of application downtime.

When planning how to increase server resources, you must take into account the resources of the NAS machine. For instance, if the machine is not capable of handling additional processes, you can negatively affect the performance of an application by increasing the number of processes running on that machine. Likewise, assigning additional threads to a process removes available threads from the system-wide thread pool, limiting the system's ability to process other thread-utilizing requests, such as database access.

Adding and Tuning Java Server and C++ Server Processes

You can add a Java Server (KJS) or C++ Server (KCS) process to increase fault tolerance. By having one or two additional processes, an application is more likely to respond to users' requests. If one process fails, for instance, the second or third process can take its place, decreasing the amount of time an application is unavailable. This is particularly useful for applications that have known problems that can cause a process to fail.

Add more KJS processes for Java applications and add more KCS processes for C++ applications. It is usually not necessary to add more than two processes for each type of application, Java and C++. If an application cannot run on one or two processes, there are most likely errors in the code that are causing the processes to fail. Those errors should be addressed by the application developer.

To add a Java Server or C++ process, perform the following steps:

1. On the NAS Administrator toolbar, click the General button to open the General window.

2. In the left pane of the General window, select the NAS machine where you want to add the KJS process.

3. From the File menu, choose New, then Process.

   The Add Process dialog box appears.
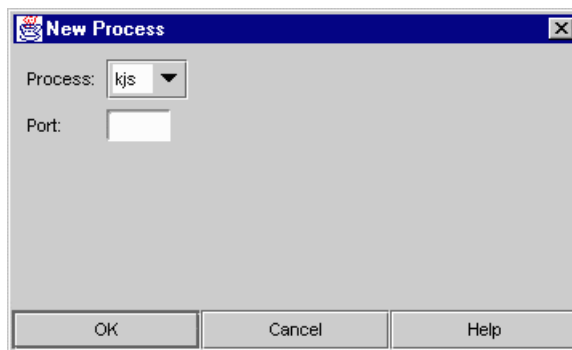
4. In the Process drop-down box, choose KJS or KCS.

5. In the Port Number text box, specify an unused port number where the additional process will run.

6. Click OK to dismiss the dialog box.

7. If this process is to be used in a single-threaded environment, perform the following steps:

   1. Click the process in the left pane of the General window.

   2. In the right pane of the window, set the Default Minimum and Default Maximum Threads to 1.

8. Click the Apply Changes button to save your changes.

Adjusting the Number of Threads for a Process

Request threads handle users' requests for application components. When NAS receives a request, the application server assigns the request to a free thread. The thread manages the system needs of the request. For example, if the request needs to use a system resource that is currently busy, the thread waits until that resource is free. When the resource is free, the thread allows the request to use that resource.

You can specify the minimum and maximum number of threads that are reserved for requests from applications. The thread pool is dynamically adjusted between those two values. The minimum thread value you specify holds at least that many threads in reserve for application requests. That number is increased up to the maximum thread value you specify on an as-needed basis.
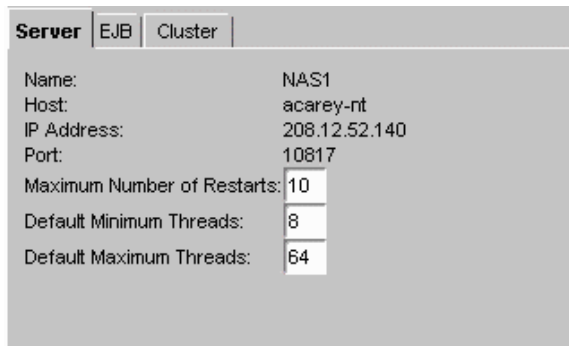
Increasing the number of threads available to a process to allow that process to respond to more application requests simultaneously. Threads can be added to a process at the process level, or globally at the NAS level.

By default, each process uses the threads assigned to NAS. For example, if NAS uses a minimum of 8 threads and a maximum of 64 threads, each individual process uses a minimum of 8 threads and a maximum of 64 threads.

To adjust the number of request threads for all (KJS/KCS/KXS) processes, perform the following steps:

1. On the NAS Administrator toolbar, click the General button to open the General window.

2. In the left pane of the General window, select the server whose number of threads you want to adjust.

3. In the Default Minimum Threads text box, enter the minimum number of threads available for the Java Server (KJS), C++ Server (KCS), and Executive Server (KXS) processes of the selected NAS machine.

4. In the Default Maximum Threads text box, enter the maximum number of threads available for the KJS, KCS, and KXS processes of the selected NAS machine.

5. Click Apply Changes to save your changes.

You can also customize the usage of threads for each process. Once you do this, however, the number you set globally at the NAS level is overridden by the number you set at the process level.

To adjust the number of threads available for individual processes (KJS, KCS, and KXS), perform the following steps:

1. On the NAS Administrator toolbar, click the General button to open the General window.

2. In the left pane of the General window, select the process whose number of threads you want to adjust.

3. In the Default Minimum Threads text box, enter the minimum number of threads available for that process.

These settings override the default settings set at the server level.

4. In the Default Maximum Threads text box, enter the maximum number of threads available for that process.

5. Click Apply Changes to save your changes.

Adjusting the Restart Option of the Administrative Server
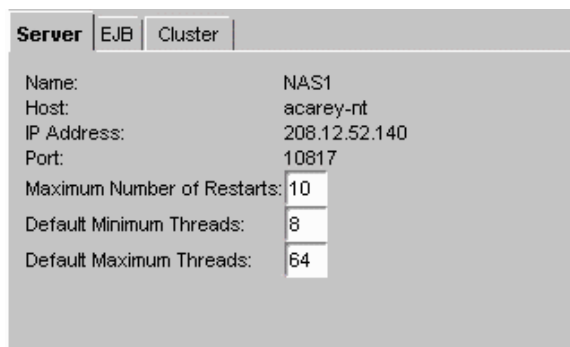
Adjust the restart option of the Administrative Server to increase or decrease the number of times the Administrative Server attempts to restart an Executive Server (KXS), Java Server (KJS), or C++ Server (KCS) process that has failed. This option increases fault tolerance and application availability by attempting to ensure that all processes are running.

The Administrative Server is the administrative process within NAS through which administrative tasks are processed.

To adjust the restart option of the Administrative Server, perform the following tasks:

1. On the NAS Administrator toolbar, click the General button to open the General window.

2. In the left pane of the General window, select the NAS machine whose you Administrative Server restart option you want to adjust.

3. In the right pane of the General window, in the Maximum Number of Restarts text field, enter the new value.

4. Click Apply Changes to save your changes.

Implementing a Multi-Process, Single-Threaded Environment

You can add a Java Server (KJS) or C++ Server (KCS) process to implement a multi-process, single-threaded environment. Running multiple KJS processes, all in single-threaded request mode, effectively creates a "multi-threaded" environment, which allows simultaneous processing of users' requests.

Implementing a multi-process, single-threaded environment allows each process to accept only one request at a time. This is useful when you are integrating third-party utilities. Running third-party utilities in the NAS multi-

threaded request environment can cause errors beyond the control of the application server, including thread safety issues. To work around this type of problem and still allow the Netscape Application Server (NAS) to scale, you can implement a multi-process, single-threaded environment.

For example, if a third-party utility runs within the KJS process, but this utility is not thread safe, you can adjust the request threads of the KJS to 1 and eliminate the utilitiy's safety issues. However, this creates a request backlog as requests wait for the KJS to process a single request at a time. To alleviate that problem, you can run multiple KJS processes, all running in single-threaded request mode, and effectively create a "multi-threaded" environment allowing simultaneous processing of users' requests.

You do need to maintain multiple request threads for the Executive Server (KXS) process, as it distributes all requests that come into NAS.

To implement a multi-process, single-threaded environment, perform the following tasks:

1. Add KJS or KCS processes.

   See "Adding and Tuning Java Server and C++ Server Processes" on page 134.

2. Adjust the request threads allocated for those processes to 1.

   See "Adjusting the Number of Threads for a Process" on page 135.

Configuring Directory Server Failover

The Directory Server connected to your Netscape Application Server (NAS) machine contains global information shared by all application servers in a Directory Server cluster. A Directory Server cluster is simply one or more NAS machines that share a single Directory Server. To protect this globally shared information, you must configure a second Directory Server to act as a backup if the primary server fails.

Before adding a backup Directory Server to your Directory Server cluster, you must replicate the NAS subtree of the primary Directory Server using supplier initiated replication (SIR). SIR is a replication configuration where servers containing master copies of directory trees and subtrees replicate directory data to servers containing replicated directory trees and subtrees.

The two copies of the NAS subtree must always be in sync with each other.

The NAS subtree is

`cn=Global, cn=nasconfig, cn=NAScluster, o=NetscapeRoot`
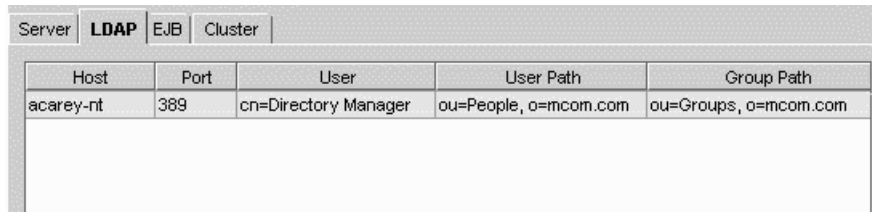
where `nasconfig` is specified during installation.

For details and replication procedures, see "Managing Replication," a chapter in *Netscape Directory Server Administrator's Guide*. This document is installed with your installation of Directory Server in the following location:

`NAS install directory/manual/en/slapd/ag/replicat.htm`

Now add a backup Directory Server using the NAS Administrator tool by performing the following steps:

1. On the NAS Administrator toolbar, click the General button to open the General window.

2. In the General window, click the LDAP tab to display the following screen:

Server	**LDAP**	EJB	Cluster		
Host	Port	User	User Path	Group Path
acarey-nt	389	cn=Directory Manager	ou=People, o=mcom.com	ou=Groups, o=mcom.com

Each Directory Server associated with your NAS machine appears in the window.

3. To add a secondary Directory Server, click the Add button.

   The following dialog box appears:

4. Enter the new server's information.

5. Click OK.

To remove a Directory Server, click Remove.

You must always have at least one Directory Server configured to work with NAS.

Configuring Directory Server Failover

Configuring the Web Connector Plug-In

This chapter describes the web connector plug-in which sends users' requests to applications residing on Netscape Application Server.

The following topics are included in this chapter:

- About the Web Connector Plug-In

- Configuring the Web Connector for Web Server Logging

- Configuring Cookie and Hidden Field Usage

- Configuring a CGI Flag for CGI Requests

- Changing the Web Connector Port Number

- Specifying HTTP Variables for Input to Application Components

About the Web Connector Plug-In

The web connector plug-in is installed on your web server at the time you install Netscape Application Server (NAS).

If you install NAS on the same machine where a web server is installed, the web connector is simultaneously installed and the web server configured automatically.

If you install NAS on a machine where a web server is not installed, you must manually install the web connector on that web server machine. For more information about manually installing the web connector, see the *Installation Guide*.

You can configure the following web connector functions:

Connector functionality	Description	More information
Web server request logging	Mapping web server request components to database fields and adding HTTP variables to the log.	"Configuring the Web Connector for Web Server Logging" on page 148
Cookie and hidden field security	Enable or disable cookies and hidden fields during web server to NAS communication.	"Configuring Cookie and Hidden Field Usage" on page 150
CGI flag for CGI request processing	Set a flag to process requests in CGI mode when that is necessary.	"Configuring a CGI Flag for CGI Requests" on page 151
The plug-in port number	Reconfigure the port number used by the plug-in.	"Changing the Web Connector Port Number" on page 152
Configuring HTTP variables as input for application components	Determine which HTTP variables can be accessed by application components.	"Specifying HTTP Variables for Input to Application Components" on page 153

Manually Configuring a Web Server

When you install Netscape Application Server (NAS), your web server is automatically configured for the web connector plug-in, meaning that all the necessary directories and settings on the web server are updated. However, there may be occasions, when, after you've installed the web connector plug-in, you must manually re-configure the web server. This procedure is recommended only if you are having problems with the connection between NAS and your web server.

The following steps explain how to manually configure a web server to use the web connector plug-in, whether your web server resides on the same or a different machine than where NAS is installed.

If you perform only step one of the following procedure (enabling CGI), the web connector will run as a CGI script. If you perform the entire procedure, the web connector will run as a plug-in, which is more efficient since a plug-in is faster than a CGI script.

You must be logged in as the same administrator user who installed the web server.

To reconfigure a Netscape web server, perform the following steps:

1. Enable CGI, if it is not already enabled:

    1. Go to the Netscape program group and click Administer Netscape Servers.

    2. Enter the administrator ID and password, and click OK.

    3. On the Netscape Server Selector screen, click on the web server instance you want to configure.

    4. On the main menu bar across the top of the page, click Programs.

    5. On the CGI directory screen under URL prefix, type `cgi-bin`.

    6. Under CGI directory, enter the cgi-bin path.

    For Netscape Enterprise Server 3.x and higher, Windows NT:

    drive letter`:/Netscape/SuiteSpot/docs/cgi-bin`

    Now you are ready to configure the web connector plug-in.

2. Edit the `obj.conf` file in the web server configuration directory.

    For Netscape Enterprise Server 3.x, Windows NT:

    drive letter`:\Netscape\SuiteSpot\https-`*machinename*`\config`

    For Netscape Enterprise Server 3.x, Unix:

    NAS install directory`/https-`*machinename*`/config`

Make a copy of the file before modifying it. At the end of the `Init` section of the `obj.conf` file, add the following as two lines:

- Windows NT:

```
Init fn="load-modules"
funcs=nas_name_trans,gxrequest,gxlog,gxinit,gxredirect,gxhtmlrequest
shlib="path to NAS bin dir/example: gxnsapi351.dll"

Init fn="gxinit"
```

- Unix:

```
Init fn="load-modules"
funcs=nas_name_trans,gxrequest,gxlog,gxinit,gxredirect,gxhtmlrequest
shlib="gxnsapi30.so"

Init fn="gxinit"
```

Specify the following for `shlib`, Netscape Enterprise Server 3.6:

- Windows NT:

```
NAS install directory\bin\gxnsapi351.dll
```

- Unix:

```
NAS install directory/gxlib/libgxnsapi30.so
```

3. In the `Object name=default` section, just after `type=text/plain` section, add the following line:

```
Service fn="gxredirect" fnname="imagemap" method="(GET|HEAD)"
```

4. In the `Object name=cgi` section(s), insert the following line immediately before the line `Service fn="send-cgi"`:

```
Service fn="gxrequest"
```

And then insert the following line immediately after the line `Service fn="send-cgi"`:

```
AddLog fn="gxlog"
```

5. Make a copy of the current version of the file `obj.conf` and copy it to the back up version (so that the backup is consistent with the current version) in the following directory:

   For Windows NT:

   `drive letter:\Netscape\SuiteSpot\https-machinename\conf_bk`

   For Unix:

   `Netscape install directory/https-machinename/conf_bk`

6. **Unix only**: Modify the web server's start and stop scripts as follows:

   In the start script:

   Set `GX_ROOTDIR` to the directory in which NAS is installed. For example:

   `GX_ROOTDIR=NAS install directory; export GX_ROOTDIR`

7. Restart the web server.

Reconfiguring the Microsoft Internet Information Server

Keep in mind the following information when reconfiguring Microsoft IIS:

- Rename the `gxisapi.dll` library to `gx.dll` and leave it in the cgi-bin directory of the IIS wwwroot (`inetput/wwwroot/cgi-bin/`).

- Configure the ISAPI filter file, `gx.dll`, in the following registry entry:

  ```
  My Computer\HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\
  W3SVC\Parameters\
  ```

  A string key, `Filter DLLs`, should be added under Parameters, with the following value:

  ```
  c:\inetpub\wwwroot\cgi-bin\gx.dll
  ```

Configuring the Web Connector for Web Server Logging

Web server requests are divided into components. Each component is represented by an HTTP variable. HTTP variables are standardized across all web servers, so the configurations you make with regard to their use are web-server independent.

Mapping HTTP Variables to Database Fields

To enable logging of a particular component of a web server request, you must map HTTP variables to specific database fields to ensure that web server requests are properly logged. Mapping HTTP variables to database fields is done in the web connector plug-in on the web server machine. The web server machine may or may not be the same machine where you installed Netscape Application Server (NAS).

To map HTTP variables to database fields, perform the following steps:

1. Open the Netscape Registry Editor by typing `kregedit` at the command line.

   The editor tool opens and displays the keys and values that apply to NAS. If the web server and NAS are installed on separate machines, the editor opens and displays the keys and values that apply to the web connector plug-in.

2. Open the following key:

   `Software\Netscape\Application Server\4.0\CCS0\HTTPLOG\INPUTVARS`

   Each value under this key represents an HTTP variable and the database field to which the variable is mapped.

   The ID of the value is the HTTP variable. The string value is the database field.

   The HTTP variable is in ALL CAPS, such as `HTTP_REFERER`, and the database field is exactly as it appears in the database table.

3. Double-click the HTTP variable you want to map to a database field.

   The String editor dialog box appears.

4. Enter the database field name as the value data and click OK.

5. Leave any HTTP variables you do not want to log blank.

6. Close the editor.

See your web server documentation for an explanation of the HTTP variables.

Use the Netscape Registry Editor to modify the web connector plug-in.

Adding HTTP Variables to the Log

You can also modify the list of available HTTP variables, adding variables to the list to expand your logging options.

To add HTTP variables to the log, perform the following steps:

1. Open the Netscape Registry Editor by typing `kregedit` at the command line.

   The editor opens and displays the keys and values that apply to NAS. If the web server and NAS are installed on separate machines, the editor opens and displays the keys and values that apply to the web connector plug-in.

2. Open the following key:

   `Software\Netscape\Application Server\4.0\CCS0\HTTPLOG\INPUTVARS`

   Each value under this key represents an HTTP variable and the database field to which the variable is mapped.

   The ID of the value is the HTTP variable. The string value is the database field.

   The HTTP variable is in ALL CAPS, such as `HTTP_REFERER`, and the database field is exactly how it appears in the database table.

3. Add a new String value with the new HTTP variable name.

4. Double-click the new HTTP variable and enter the database field name as the value data.

5. Click OK.

6. Repeat steps 3 through 5 for each new HTTP variable.

7. Close the editor.

See your web server documentation for a list and an explanation of all available HTTP variables.

Configuring Cookie and Hidden Field Usage

Netscape Application Server (NAS) is designed to work with web browsers in all modes of cookie and hidden-field security. There are three configurations you can set for the web connector plug-in to support the various security modes. These configurations are described in the following table:

Cookie setting	Description
0	Cookies and hidden fields are passed back to the requesting web browser. This is the default setting.
1	Only hidden fields are passed back to the requesting web browser.
2	Only cookies are passed back to the requesting web browser.

To configure cookie and hidden field usage, perform the following steps:

1. Open the Netscape Registry Editor by typing `kregedit` at the command line.

   The editor tool opens and displays the keys and values that apply to NAS. If the web server and NAS are installed on separate machines, the registry editor opens and displays the keys and values that apply to the web connector plug-in.

2. Open the following key:

   ```
   Software\Netscape\Application Server\4.0\CCSO\HTTPAPI
   ```

3. Double-click the `NoCookie` DWORD value.

    The DWORD editor dialog box appears.

4. To disable cookies being passed to the web browser, change the value data to 1.

5. To disable hidden fields being passed to the web browser, change the value data to 2.

6. To enable both cookie and hidden fields, change the value data to 0.

7. When finished, close the editor.

Configuring a CGI Flag for CGI Requests

Some requests must be processed in CGI mode. You can set a flag in the web connector plug-in to identify those requests.

To configure a CGI flag for CGI requests, perform the following steps:

1. Open the Netscape Registry Editor by typing `kregedit` at the command line.

    The editor opens and displays the keys and values that apply to Netscape Application Server (NAS). If the web server and NAS are installed on separate machines, the editor opens and displays the keys and values that apply to the web connector plug-in.

2. Open the following key:

    `Software\Netscape\Application Server\4.0\CCSO\HTTPAPI`

3. Double-click the `AgentToken` String value.

    The String Editor dialog box appears.

4. For the value data, enter the flag that marks requests for CGI mode processing.

5. Click OK.

6. Close the editor.

Changing the Web Connector Port Number

In certain configurations, the web connector port number might conflict with another software package. You can reconfigure the connector port number to resolve this conflict.

To change the web connector port number, perform the following steps:

1. Open the Netscape Registry Editor. by typing `kregedit` at the command line.

   The editor opens and displays the keys and values that apply to Netscape Application Server (NAS). If the web server and NAS are installed on separate machines, the editor opens and displays the keys and values that apply to the web connector plug-in.

2. Open the following key:

   ```
   Software\Netscape\Application Server\4.0\CCSO\HTTPAPI
   ```

3. Double-click the `ListenPort` DWORD value and change the value data to an available port number.

4. Click OK.

5. Close the editor.

Specifying HTTP Variables for Input to Application Components

HTTP variables can be passed as part of the application request to application components like Enterprise Java Beans (EJBs). This allows the developer to determine certain information about the request and use that information when processing the request.

For example, the application might look at the HTTP_REFERER variable to determine where the request is coming from. This information might be used to present a more individualized greeting screen, or to keep statistics about where requests originate.

These variables are specified by setting the HTTP variable to a 1 for Microsoft web servers, or to a string for Netscape web servers. Use the current entries in the registry as an example.

To specify HTTP variables for input to application components, perform the following steps:

1. Open the Netscape Registry Editor by typing kregedit at the command line.

   The editor opens and displays the keys and values that apply to Netscape Application Server (NAS). If the web server and NAS are installed on separate machines, the editor opens and displays the keys and values that apply to the web connector plug-in.

2. Open the appropriate key:

   • For Netscape web servers, open the following key:

   Software\Netscape\Application Server\4.0\CCSO\HTTPAPI\INPUTNSAPI

   • For Microsoft web servers, open the following key:

   Software\Netscape\Application Server\4.0\CCSO\HTTPAPI\INPUTISAPI

   Each value shown represents an HTTP variable. The value also indicates the HTTP variable is passed to NAS with the application request. If the value is non-zero, the HTTP variable is passed to the NAS machine with the application request.

The HTTP variable is in ALL CAPS, such as `HTTP_REFERER`.

3. Add a new String value with the new HTTP variable name.

4. Double-click the new HTTP variable and enter the one of the following as the value data:

   • For Netscape web servers, enter a string value, such as that used for database mapping.

   • For Microsoft web servers, enter a 1.

5. Click OK.

6. Repeat steps 4 through 6 for each new HTTP variable.

7. Close the editor.

Administering Database Connectivity

Netscape Application Server applications are able to access a database, or several databases, to add, retrieve, and modify data. This chapter describes how to configure data access drivers and apply settings to database connectivity parameters.

The following topics are included in this chapter:

- About Data Access Drivers
- Adjusting Database Connectivity Parameters

About Data Access Drivers

Netscape Application Server (NAS) applications often require database access.Database access is achieved through a data access driver, which is software written either by the database vendor or a third-party vendor. The following types of data access drivers can be configured with NAS to provide database connectivity:

- Oracle
- DB2
- Informix

- Sybase

- MSSQL server (for NT)

- ODBC

Make sure that data access drivers are installed before installing an instance of NAS. This way, NAS can automatically configure the drivers.

Configuring Data Access Drivers

When you open the Database window of NAS Administrator, the left pane displays all data access drivers installed on a particular server whether the drivers are configured or not. A red X appears next to drivers that are not configured.

To configure a data access driver, perform the following steps:

1. From the NAS Administrator toolbar, click the Database button to open the Database window.

2. In the left pane of the Database window, click the driver you want to configure.
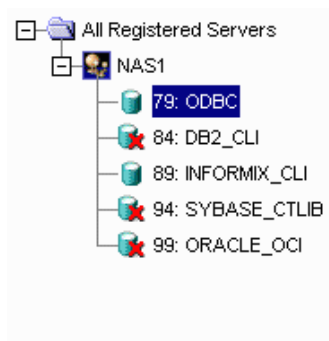
3. In the right pane of the Database window, click Load Data Access Driver.

   Information about the data access driver appears in the Database window.

4. In the Client Library field, you can edit the library corresponding to the data access driver.

5. In the Priority field, you can edit the priority of the data access driver.

   Giving a data access driver a priority of 1 means that driver has first priority over all other drivers. The higher the number, the lower the priority.

6. Click Apply Changes to save your changes to NAS.

   Changes are not applied until you restart the server.

Adjusting Database Connectivity Parameters

Netscape Application Server (NAS) allows you to adjust database connectivity through connection parameters. Connection parameters allow you to optimize the speed with which NAS connects to a database or databases. The connection parameters are grouped in the following categories:

- connection

- threads

- result set buffer

- database cache

Setting Connection Parameters

You can set the length of time NAS attempts to make a database connection. These parameters optimize the performance of the NAS machine by keeping the server from wasting resources. For example, because NAS waits for open database connections when a request is made, the connection time limit is useful to limit the server from endlessly trying to connect to a database that is down.

To set the connection parameters, perform the following steps:

1. From the NAS Administrator toolbar, click the Database button to open the Database window.

2. In the left pane of the Database window, click the database for which you want to adjust the timeout parameter.
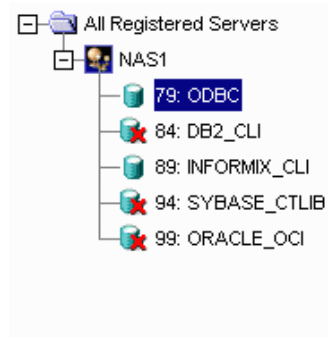
3. In the right pane of the Database window, in the Connection Timeout field, enter the number of seconds.
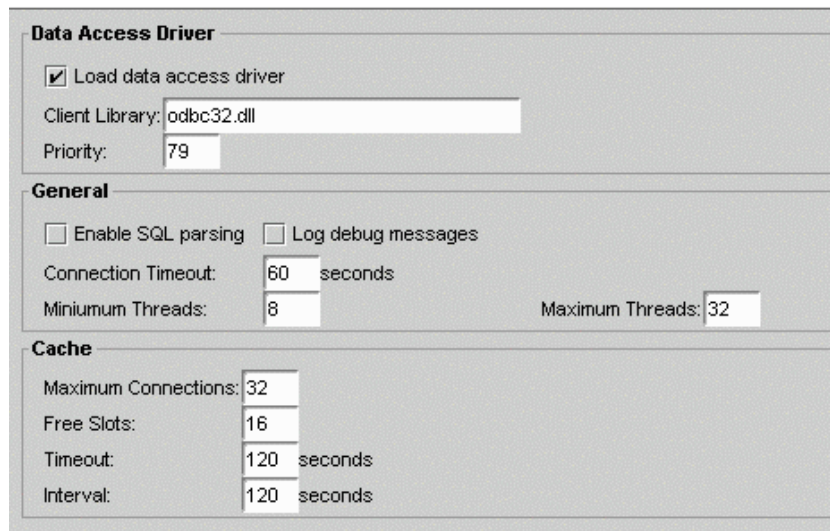
4. Click Apply Settings to save the changes to NAS.

Setting Thread Parameters

You can set the minimum and maximum number of threads available for database connections. The thread parameters determine how many threads NAS allocates for asynchronous database queries. Such threads are usually used for queries returning a large number of rows and allowing the application to do other tasks while waiting for the query to finish. Asynchronous database queries are not supported by JDBC 2.0, a Java programming interface used to build on top on database drivers.

The default thread allocations are adequate for most applications. If an application developer uses many asynchronous queries, you might want to increase the maximum number of available threads. Keep in mind that each thread does use a small stack allocation and pulls from the total number of available system threads. Therefore, if an application does not use any asynchronous queries, you can increase performance by setting the maximum available threads to zero.

To set the thread parameters, perform the following steps:

1. From the NAS Administrator toolbar, click the Database button to open the Database window.

2. In the left pane of the Database window, select the database for which you want to adjust the asynchronous thread parameters.
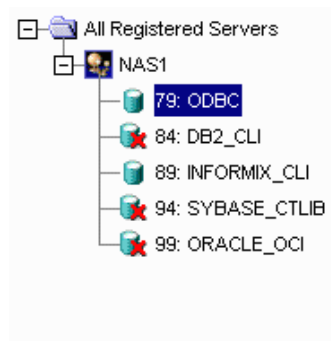
3. In the right pane of the Database window, in the Minimum Threads field, enter the number of threads.

4. In the right pane of the Database window, in the Maximum Threads field, enter the number of threads.

5. Click Apply Settings to save the changes to NAS.

Setting Database Cache Parameters

The database cache is an array used to hold active and recently used database connections. NAS adds database connections to cache when an application creates a database connection.

While the application is using that database connection, NAS marks that connection "in use." Once the database operations are finished, the server marks the database connection "free." The cache then holds the free connection in the cache for a configured period of time. This allows the server to use the free cached connection and quickly handle a new request to the same database. Once a free connection exceeds the timeout, a cleaning thread removes the connection from the cache and opens a slot for a new connection to be cached.

You can adjust the following cache parameters:

- the maximum number of connections allowed in the cache

- the number of slots held solely for free connections

- the timeout limit, in seconds, for free connections

- the interval, in seconds, at which the cache cleaner thread removes timed-out free connections

The default values are adequate for most applications, so adjustments are not usually required for initial application installations.

NAS dynamically adjusts the cache up to the maximum number of allowable connections. If there are no connections to cache, the array is allocated to zero spaces.

To set database cache parameters, perform the following steps:

1. From the NAS Administrator toolbar, click the Database button to open the Database window.

2. In the left pane of the Database window, select the database for which you want to adjust the database cache parameters.
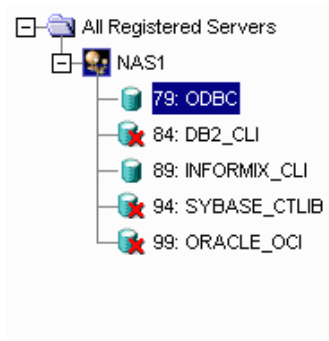
3. In the right pane of the Database window, under Cache, enter values for the following parameters:

- maximum connections
- free slots

- timeout
- interval

4. Click Apply Settings to save the changes to NAS.

Adjusting Database Connectivity Parameters

Administering Transactions

This chapter describes the tasks and conceptual information necessary for administering transactions using the Netscape Application Server (NAS) Administrator.

The following topics are included in this chapter:

- About the Transaction Manager

- Storing Distributed Transactions Log Data

- Administering Distributed Transactions in the Transaction Window

- Administering Distributed Transactions from the Command Line

- Setting Up Resource Managers for Distributed Transactions

- Enabling XA Logging

- Resolving In-Doubt Transactions

- Recovering from Log Failure

About the Transaction Manager

The transaction manager is installed with each instance of Netscape Application Server (NAS) to coordinate global transactions within a Java Server (KJS) process. Global transactions are a set of related operations that must be executed as a unit, though each operation may run in a different process.

You can use global transactions to update a database that uses one or more Enterprise Java Beans (EJBs) running concurrently for the same global transaction, from within one or more KJS processes. This occurs when an EJB triggers another EJB to run and they both participate in the same transaction. You can also update multiple databases that are distributed over different geographic locations or update multiple databases of different types (such as Oracle and Sybase).

The transaction manager runs within a KJS process and creates two files: a `restart` file and a `restart.bak` file. In addition, you need to provide a log file for each KJS process. You can administer these files from the command line or by using the Transaction window of NAS Administrator.

Storing Distributed Transactions Log Data

An installation of Netscape Application Server (NAS) consists of one Administration Server (KAS) process, one Executive Server (KXS) process, and at least one Java Server (KJS) process. A transaction manager exists for each KJS.

As a NAS administrator, you must maintain one logical volume and its restart data for each KJS in a NAS installation. A logical volume is made up of one or more physical volumes. A physical volume stores the state of all ongoing transactions. If you have more than one physical volume, additional physical volumes are backups, or mirrors, of the first physical volume.

When you initially start NAS, NAS looks in the registry for the location of the directory root. In this location is an empty log file for each KJS where NAS will write information about the state of all ongoing distributed transactions for that process. NAS then creates additional files called `restart` and `restart.bak` (a backup of `restart`) for each KJS, which record the location of the log file and the state of the logical and physical volumes. Thereafter, whenever you

start the server, NAS refers to the `restart` file for the location and state of the log file and does not refer to the registry. `Restart` and `restart.bak` are stored in the following directories:

`$DirectoryRoot/`*`KJS`* `#/restart`

`$DirectoryRoot/`*`KJS`* `#/restart.bak`

You should store `restart.bak` on a different device if possible. If `restart` becomes corrupted, NAS uses `restart.bak` to determine the location of the log file and state of ongoing distributed transactions. If both `restart` and `restart.bak` are corrupted, the transaction manager will become inoperable and you must "cold-start" the server. When you cold-start a server, NAS must look to the registry for the location of the log file as it did in its initial startup; all restart data is lost. The log file and all data will then be overwritten.

The following table lists the registry entries to which NAS refers along with their default values:

Registry Entry	Default values
`DirectoryRoot`	*NAS install directory*`/CCS0/TXNMGR`
`MirrorDirectoryRoot`	*NAS install directory*`/CCS0/TXNMGR_MIRROR`
KJS `#/LogVolumeDiskName`	`$DirectoryRoot/`*KJS* `#/logVol`, size is 4M

Administering Distributed Transactions in the Transaction Window

You can administer transactions using the Transaction window of Netscape Application Server (NAS) Administrator. To access the Transaction window from the NAS Administrator toolbar, click the Transaction button as shown in the following illustration:
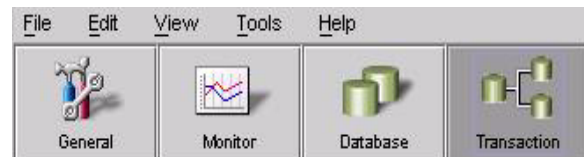
About the Transaction Window

The left pane of the Transaction window displays a tree of nodes as shown in the following illustration:
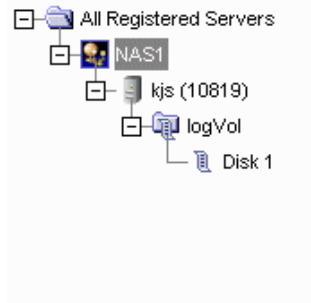
The top level of the tree lists which servers are registered with NAS Administrator. The second level, below each registered server name, displays one or more process nodes. These nodes indicate which processes are running on each registered server. Only Java Server (KJS) processes appear in the tree because only KJS processes support transactions. The third level of the tree displays the physical volumes for each process. Finally, the fourth level of the tree displays the disks in each physical volume. See "Storing Distributed Transactions Log Data" on page 166 for more information about physical volumes.

When you click a physical volume node, the right pane of the transactions window displays the page size, or size of a page used in the transaction manager, the total size of the physical volume, and the amount of unused disk space in the physical volume. You cannot edit these values.

A disk can be thought of as a partition of the physical volume. You can create an unlimited number of disks, but you cannot delete a disk once it's created. When you click a disk node, the right pane of the Transactions window displays the location and size of the selected disk.

Configuring Transactions per Server

To change transaction settings for an application server, click a registered server in the left pane of the Transaction window. The Configuration tab appears in the right pane as shown here:

Configuration	Resource Manager	Transaction Manager

Server Transaction Properties

Root Directory: C:\Netscape\Server4\nas\CCS0\TXNMGR

Mirror Root Directory: \Netscape\Server4\nas\CCS0\TXNMGR_MIRROR
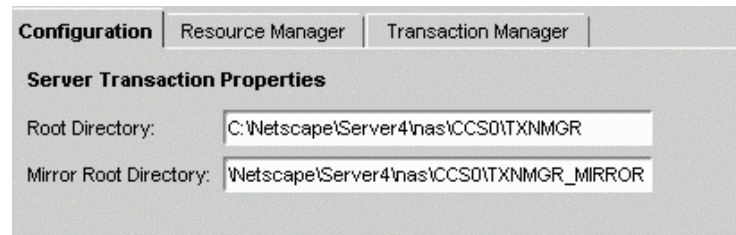
The selected server's current root and mirror directories are listed on the Configuration tab. Since no error checking is provided, it is not recommended that you edit these directories.

Viewing Transactions on a Selected Server

You can view transactions running on the selected server by clicking the Transaction Manager tab.

The following window appears:
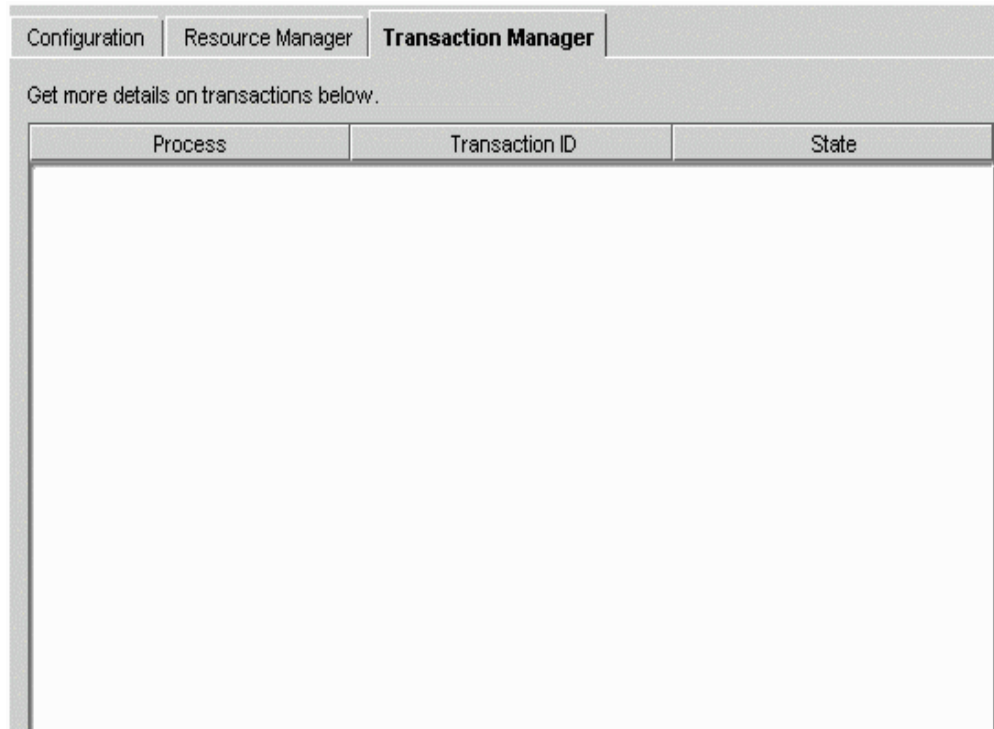
The Transactions tab displays details about all the transactions running on the selected server. For each transaction, the tab displays the following information:

- process: the Java Server process (KJS) where the transaction is running

- transaction ID: an arbitrary number used to identify the transaction

- the current state of the transaction

Click the Update button periodically to remove expired transactions from view and display currently running transactions in the window.

Viewing Transaction Details

To view details about a transaction, click the Details button.

The Transactions Detail dialog box appears.

In the text box, Originator indicates where the selected transaction originates. The Participants box indicates where the transaction is currently running.

You can force the transaction into a state by clicking the appropriate button (Abort, Force Abort, Force Commit, Force Finish).

Configuring Transactions per Process

Click the process in the left pane of the Transaction window to change transaction settings for a process on an application server.
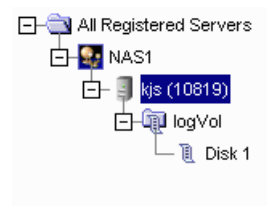
The Configuration tab appears in the right pane as shown in the following illustration:
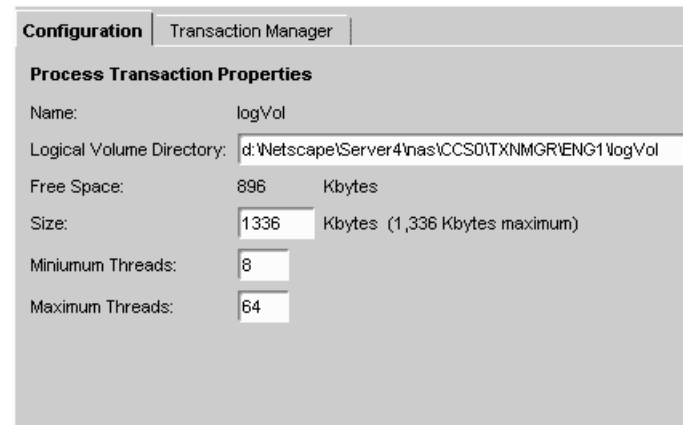
The logical volume size for the process is displayed. You can set the size of the logical volume by entering a number in the Logical Volume Size field. A logical volume must be at least 8 MB and less than 10 MB.

Viewing Transactions on a Selected Process

Click the Transaction Manager tab to view the details of all transactions running on the selected process. The following window appears:
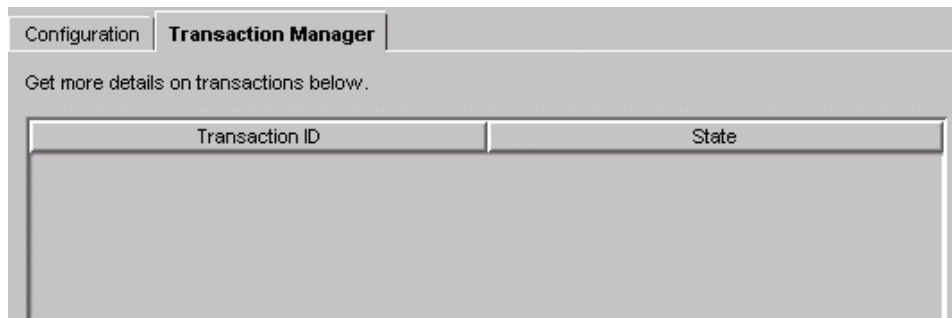
The transaction ID and state appear. See "Configuring Transactions per Server" on page 169 for more information.

Configuring Resource Managers

A resource manager enables you connect to a database back end for global transactions. If you enable a resource manager, the transaction manager within a KJS process attempts a connection to the database when the KJS process is started.
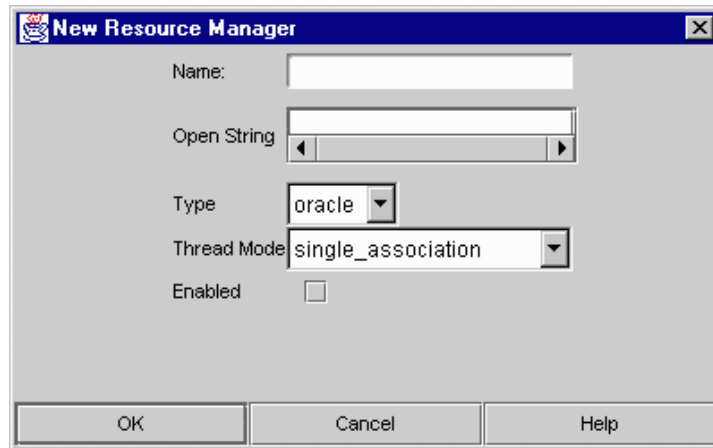
There is one resource manager for each database the application server can access. Click the Resource Manager tab in the Transaction window to configure resource managers. The following window appears:

Configuration	**Resource Manager**	Transaction Manager		
Name	Open String	Type	Thread Mode	Enabled
test this	hello	oracle	single_association	✔

The name of each resource manager (for instance, Microsoft SQL) as well as its status (enabled or disabled) is displayed. Click the Enabled checkbox to toggle the status of each resource manager. Note that you must restart the server before changes to your resource manager configuration take effect.

Adding and Editing Resource Managers

Click the Add or Modify buttons to add or edit a resource manager. The following dialog box appears:

In the Name field, enter a value to distinguish the selected resource manager from other resource managers.

In the OpenString field, enter the parameters for accessing a particular database (user name, password, permissions).

Select the type of database from the Type drop-down box (for instance, Microsoft SQL).

Choose the thread mode from the drop-down box:

- multiple_associations: the transactation manager XA (TM-XA) service performs no serialization of XA operations between threads.

- serialize_all_operation: the TM-XA service permits a maximum of one thread to make an XA call to the resource manager client library at a time.

- serialize_start_end: the TM-XA service ensures that only one association with the resource manager client library is attempted at a time.

- single_association: the TM-XA service does not prevent multiple threads from attempting different associations at the same time.

Finally, to enable or disable the resource manager, click the Enabled checkbox. Only one resource manager may be enabled for each database type.

You must restart the server before changes take effect.

Administering Distributed Transactions from the Command Line

You can also administer transactions from the command line. Invoke the command-line tool with the following script:

```
ksvradmin -l
```

The following table lists `nasadmin` commands you can execute from the command line. Once you invoke the command-line tool, each command in the following table is preceded by `nasadmin` command prompt as shown in the following example:

```
nasadmin > abort transaction
```

nasadmin Command	Function	Input parameter	Output parameter
`abort transaction`	Abort a server transaction.	`DWORD tid`	
`add trace`	Add a trace mask.	`STRING traceSpec`	

`add mirror`	Add a mirror to a logical volume.	`STRING lVol,`
		`STRING pVol,`
		`STRING diskName`	

`dump component`	Dumps the internal state of a component	`STRING componentName`	
`dump ringbuffer`	Dumps the current contents of the ringbuffer	`STRING destination`	
`expand lvol`	Expand a logical volume.	`STRING lVol,` `DWORD newSize`	
`expand pvol`	Expand a physical volume.	`STRING pVol,` `STRING diskName`	

`force transaction`	Force the outcome of a transaction.	`DWORD tid,WORD`
		`commitDesired,`
		`WORD finish`	

`help`	Display help message for given command	`{STRING commands}`	
`list trace`	Lists the current trace masks for Encina components		

nasadmin Command	Function	Input parameter	Output parameter
`list transactions`	List unresolved transactions in the server.	`DWORD originator, DWORD participant, DWORD globalID`	`DWORD tid, WORD state` (for example, active or inactive)
`list lvols`	List all known logical volumes.	`WORD enabled`	`{STRING lVol}`
`list pvols`	List all known physical volumes.		`{STRING pVol}`
`query transaction`	Query transaction attributes.	`DWORD tid, WORD state, WORD originator,WORD participants, WORD global`	`STRING globalID, WORD  state, STRING originator, {STRING participant}`
`query logvol`	Query a log volume.	`STRING logVol`	`STRING archiveDevice, DWORD freePages, DWORD numLogFile, {STRING logFile}`
`query lvol`	Obtain information about a logical volume.	`STRING lVol`	`DWORD pageSize, DWORD size, {STRING pVol,  WORD state` (e.g. clean or dirty)`, WORD isMounted}`
`query pvol`	Obtain information about a physical volume.	`STRING pVol`	`STRING lVol, DWORD chunkSize, DWORD numRegions, {STRING disk, DWORD offset, DWORD size}, DWORD totalSize`

nasadmin Command	Function	Input parameter	Output parameter
redirect trace	Redirects trace to the specified destination	`STRING destination {ringbuffer, stderr, stdout, filename}`	
remove mirror	Remove a mirror from a logical volume	`STRING lVol, STRING pVol`	
sync mirrors	Synchronize mirrors of a logical volume	`STRING lVol`	

The following table lists commands you can use in addition to those provided by `nasadmin`. As shown in the following example, these commands are not preceded by `nasadmin` at the command line.

```
%set server
```

Command	Function	Input parameter
logon	Log on to KAS for a NAS installation.	`STRING name, DWORD host, DWORD port, STRING userName, STRING password, WORD autoconnect`
list servers	List all the engines.	
set server	Set KES as the current server and one of the engines to be the current engine. By default, the first KXS is the current server and the main engine of the KXS is the current engine.	`STRING name, WORD engNum`

Command	Function	Input parameter
`create resourcemanager`	Create a resource manager.	`STRING name, STRING openString, STRING type,  STRING threadmode, WORD isenabled`
`delete resourcemanager`	Delete a resource manager.	`STRING name`
`set resourcemanager`	Set an existing resource manager by modifying its open string.	`STRING name, STRING openString, STRING threadmode, WORD isenabled`
`list resourcemanager`	List all the resource managers defined in the registry	
`get adminmode`	Return admin mode(0 or 1) for a KJS.	`WORD adminMode`
`set adminmode`	Set admin mode for a KJS.	

Setting Up Resource Managers for Distributed Transactions

Before you can connect to resource managers to use in distributed transactions, you must perform setup tasks that are not required for local transactions. The following section contains information about the following types of resource managers:

- Oracle

- Sybase

- DB2

- Microsoft SQL Server

You must restart the server after making changes to a resource manager.

Oracle

To set up an Oracle resource manager, perform the following steps:

1. Enter the open string in the following format:

```
Oracle_XA+DB=<Server_Instance>+Acc=P/<user name>/
<password>+Sqlnet=<Server Instance>+SesTm=<Session time
out>+Threads=<Thread safe mode>
```

   If you are trying to connect to the `bb734` instance using the user name `system` and the password `manager`, the open string appears as shown the following example:

```
Oracle_XA+DB=bb734+Acc=P/system/
manager+Sqlnet=bb734+SesTm=90+Threads=True
```

   Use the setting `Threads=True` only in the `multiple_associations` thread mode, which is the recommended mode for use with Oracle resource managers. Other thread modes reject this setting. Omit this parameter or use the setting `Threads=False` with other thread modes.

   It is strongly recommended that you use only one thread mode for all Oracle resource managers; do not mix and match thread modes for multiple resource managers.

2. Make sure the three required catalog tables for recovery exist. If they don't, create them using the following script:

```
$ORACLE_HOME/rdbms80/admin/xaviews.sql (see below)
rem
rem $Header: xaview.sql 7020200.1 95/04/05 13:07:30 rdhoopar
Generic<base> $ xaview2.sql Copyr (c) 1989 Oracle
rem
Rem ----------------------------------------------------------------
--
Rem NAME
Rem XAVIEW.SQL
Rem FUNCTION
Rem Create the view necessary to do XA recovery scan of prepared
Rem and heuristically completed transactions.
```

```
Rem NOTES
Rem The view 'XATRAN' basically combines information from two
Rem different types of tables:
Rem pending_trans$ & pending_sessions$
Rem x$k2gte2
Rem The view v$pending_xatrans$ combines and then filters
Rem information
Rem from the table pending_trans$ and pending_sessions$ into format
Rem that satisfy XA criteria.
Rem   Then the view v$xatrans$ combines information from x$k2gte2 and
Rem    v$pending_xatrans$.
Rem MODIFIED
Rem   cchew      07-15-92  - added fmt column
Rem   cchew      05-22-92  - No more fmt=0 condition
Rem   cchew      01-19-92  - Creation
Rem ---------------------------------------------------------------

DROP VIEW v$xatrans$;
DROP VIEW v$pending_xatrans$;

CREATE VIEW v$pending_xatrans$ AS
(SELECT global_tran_fmt, global_foreign_id, branch_id
   FROM   sys.pending_trans$ tran, sys.pending_sessions$ sess
   WHERE  tran.local_tran_id = sess.local_tran_id
     AND    tran.state != 'collecting'
     AND    BITAND(TO_NUMBER(tran.session_vector),
                 POWER(2, (sess.session_id - 1))) = sess.session_id)
/

CREATE VIEW v$xatrans$ AS
(((SELECT k2gtifmt, k2gtitid_ext, k2gtibid
   FROM x$k2gte2
   WHERE  k2gterct=k2gtdpct)
 MINUS
  SELECT global_tran_fmt, global_foreign_id, branch_id
   FROM   v$pending_xatrans$)
UNION
 SELECT global_tran_fmt, global_foreign_id, branch_id
   FROM   v$pending_xatrans$)
/
```

Sybase

Sybase is only available on Solaris platforms. To set up a Sybase resource manager, perform the following steps:

1. Name the resource manager by adding entries to `xa_config`. The entries should be in the following format:

```
[xa]
lrm=ksample_rm
server=ksample
```

2. Enter the open string in the following format:

```
-U<User name> -P<Password> -N<RM name> -Txa
```

For example, if you are trying to connect to `ksample_rm`, which is set up to connect to a ksample server instance, the open string is in the following format:

```
-Uuser -Ppswd -N ksample_rm -Tevent
```

If you want do not want to suppress logging user names and passwords to a trace file, use `-Txa` instead of `-Tevent` in the open string.

3. Make sure that `libxa.so` exists in the `$SYBASE/lib` directory.

XA libraries do not come by default with Sybase client libraries.

4. Run the following scripts available in the `$SYBASE/scripts/` directory:

```
xacommit.sql
xacompot.sql
xasproc.sql
xapropt.sql
xa_ld_q1.sql
xa_ld_q2.sql
```

DB2

To set up a DB2 resource manager, perform the following steps:

1. Enter the open string in the following format:

   ```
   <DataSourceName,UserName,Password>
   ```

   For example, if you are connecting to `ksample` and using `inst1/inst1` as user name and password, the open string is in the following format:

   ```
   ksample,inst1,inst1
   ```

2. Enter the following in the DB2 configuration:

   ```
   db2 update dbm cfg using TP_MON_NAME libEncServer_nodce
   ```

   DB2 uses dynamic registration to participate in distributed transactions. On NT, DB2 needs to know which shared library implements the dynamic registration functions like `ax_reg()` and `ax_unreg()`.

3. Make sure `$DB2DIR/lib/libdb2.so` has `755` permissions.

   If it does not, the Java Server (KJS) process will crash when calling `xa_open`.

4. Make sure that `$DB2LIB/sqllib/lib/libdb2.so` has `r-x` permissions

   If it does not, the KJS process will crash upon startup.

5. Set the `CURSORHOLD` parameter to zero in the `db2cli.ini` file.

   The cursor hold feature does not work in the XA environment.

6. In the `db2cli.ini` file, set `DISABLEMULTITHREAD` to 1.

   A sample entry in `db2cli.ini` should now look like the following example:

   ```
   [ksample]
   CURSORHOLD=0
   AUTOCOMMIT=0
   LONGDATACOMPAT=1
   DISABLEMULTITHREAD=1
   ```

Note You cannot mix local and global connections using DB2 on either Solaris or Windows NT platforms. Disable all DB2 global data sources for local transactions to function properly.

Microsoft SQL Server

To set up a Microsoft SQL resource manager, perform the following steps:

1. Enter the open string in the following format:

   ```
   Tm=transaction manager's name RmRecoveryGuid=GUID
   ```

   In the NAS environment, `tm` is Encina.

   Find and copy the value for `RmRecoveryGuid` in the following registry entry:

   ```
   HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\MSSQLServer\ResourceMgrID
   ```

   If this registry entry is missing, generate a GUID using the kguidgen tool.

2. Install and set up the Distributed Transaction Coordinator (DTC). You can get DTC from Microsoft's web site or from MSDN Windows NT option pack 4.0.

   When the DTC is installed, the Microsoft DTC (MS DTC) section exists in the `SOFTWARE\MICROSOFT\`hive.

   It is not necessary to install the Microsoft Transaction Server (MTS).

3. Make sure the ODBC driver on your server machine is version 3.5 or higher.

4. Make sure the following XA-related stored procedures are installed on the MS SQL Server machine where the application server connects:

   `sp_start_xact`, `sp_scan_xact`, `sp_commit_xact` or their deprecated names such as `start_xact`, `scan_xact` and `commit_xact`.

Enabling XA Error Logging

To log XS error messages, follow the directions for the type of resource manager you are using:

- Oracle

- Sybase

- DB2

- Microsoft SQL Server

Oracle

In the open string, add a log directory as shown in the following example:

```
Oracle_XA+DB=<bb734>+Acc=P/system/
manager+Sqlnet=bb734+SesTm=90+Threads=True+LogDir=/export/logs
```

where `/export/logs` is the log directory.

Make sure that the log file generated by LogDir allows administrator access only as it contains the user names and passwords for the database.

Sybase

In the open string, add a log directory as shown in the following example:

```
-Uuser -Ppswd -N ksample_rm -Tevent -L/export/logs/syb_xa_log
```

where `/export/logs` is the log directory.

Make sure that the log file generated by LogDir allows administrator access only as it contains the user names and passwords for the database.

DB2

Enter the following commands to enable the logging of XA calls and/or interfaces:

```
db2 update dbm cfg using DIAGLEVEL 4
db2 update dbm cfg using DIAGPATH $GX_ROOTDIR/logs
```

The log will be created under file name called `db2diag.log`.

XA failures appear in the following format:

```
String Title: XA Interface SQLCA  PID:28084 Node:000
SQLCODE = -998  REASON CODE: 4  SUBCODE: 4
```

Using the `REASON CODE` and `SUB CODE`, you can find the cause of an error by looking up the code in the following table:

Code	Cause of error	Action
01 - (XAER_ASYNC)	Asynchronous operation already outstanding.	Entry is made in system log.
02 - (XAER_RMERR)	Resource manager error occurred in transaction branch.	Entry is made in system log.
03 - (XAER_NOTA)	XID is not valid.	Entry is made in system log.
04 - (XAER_INVAL)	Invalid arguments given.	Entry is made in system log. Verify content of xa open string and make necessary corrections.
04 - 01 - (xa_info)	Pointer is invalid (for example, the XAOpen string is null).	
04 - 02	Database name exceeds maximum length.	
04 - 03	User name exceeds maximum length.	

Enabling XA Error Logging

Code	Cause of error	Action
04 - 04	Password exceeds maximum length.	
04 - 05	User name specified but not a password.	
04 - 06	Password specified but not a user name.	
04 - 07	Too many parameters in the xa_info string.	
04 - 08	Multiple xa_opens generate different RM ids for the same database name.	
04 - 09	Database name not specified.	
05 - (XAER_PROTO)	Routine invoked in improper context.	Entry is made in system log.
06 - (XAER_RMFAIL)	Resource manager unavailable.	Entry is made in system log.
07 - (XAER_DUPID)	XID already exists.	Entry is made in system log.
08 - (XAER_OUTSIDE)	Resource manager doing work outside distributed transaction.	Entry is made in system log.
09	Registration (`ax_reg`) with transaction manager failed.	
09 - 01	Joining XID not found.	
09 - 02	Dynamic library specified in the `tp_mon_name` configuration parameter could not be loaded.	Ensure that the `tp_mon_name` configuration parameter contains the name of the dynamic library in the external product which has the `ax_reg()` function used for dynamic registration of transactions.
10	Attempted to start a different transaction while suspended.	

Code	Cause of error	Action
12	Unregistering (`ax_unreg`) with transaction manager failed.	
13	Ax interface failure: `ax_reg()` and `ax_unreg()` not found.	
35	Heuristic operations invalid for non-XA database.	Heuristic operation attempted against a database that only participates only as a read-only resource manager in a distributed transaction (for example, any DRDA databases like DB2 on MVS).
36	XID not known by database manager.	Invalid heuristic operation attempted on an in-doubt transaction. Wrong XID specified or a heuristic or resync operation took place since you recorded XID. Perform a Heuristic Query request to get the current list of in-doubt transactions to verify if you still need to perform your heuristic operation.
37	Transaction has already been heuristically committed.	Invalid heuristic operation attempted on an in-doubt transaction. Wrong XID specified or a heuristic or resync operation took place since you recorded XID. Perform a Heuristic Query request to get the current list of in-doubt transactions to verify if you still need to perform your heuristic operation.

Code	Cause of error	Action
38	Transaction has already been heuristically rolled back.	Invalid heuristic operation attempted on an in-doubt transaction. Wrong XID specified or a heuristic or resync operation took place since you recorded XID. Perform a Heuristic Query request to get the current list of in-doubt transactions to verify if you still need to perform your heuristic operation.
39	Transaction is not an in-doubt transaction.	XID specified is for a transaction that has ended and is waiting for the two-phase commit process to begin. Only perform heuristic operations on transactions in the two-phase commit process and have become in-doubt transactions.
40	Only rollbacks allowed for this transaction.	SQL statement attempted under a failed transaction.
69	Database log ID mismatch during DUOW re-synchronization.	Transaction manager database or resource manager database names could be referencing different database instances.
85	As a result of heuristic processing, transaction has partially committed and rolled back.	Attempting to update multiple data sources. Some data sources have been heuristically rolled back or committed, resulting in partially committed transaction that has been rolled back. To correct the data, you must manually check every data source updated by the transaction.

Microsoft SQL Server

The log file for the XA interface, dtcxa.log, is created under the current KJS directory.

Resolving In-Doubt Transactions

Occasionally, particularly when a Java Server (KJS) process quits suddenly, you may find "hanging" or in-doubt transactions. For Microsoft SQL Server, in order to manually commit or rollback in-doubt transactions, use DTC administrator control. This is also known as DAC. `dac.exe` is found in the `WINNT\SYSTEM32\` directory and is installed with DTC.

After starting DAC, perform the following steps to manually commit or rollback in-doubt transactions:

1. From the NAS Administrator toolbar, click the Transactions button to open the Transactions window.

2. Click the Transaction Manager tab.

3. Select the transaction that you want to force and click Details.

4. Click the Resolve/Abort button to force rollback the transaction.

Oracle   For Oracle resource managers, if you encounter a "lock held by distributed transaction" error, you must connect to the database and rollback the global transaction explicitly. To do so, perform the following steps:

1. Find out the local transaction ID that corresponds to the transaction by looking at `dba_2pc_pending`, which has all the details about pending global transactions.

   For example, type the following at the `SQLPLUS` prompt:

   `SQLPLUS>select * from dba_2pc_pending`

2. Rollback the transaction by typing

   `rollback force transaction_id`

   at the command line.

Sybase   For Sybase resource managers, if you encounter a "lock held by distributed transaction" error, you must connect to the database and rollback the global transaction explicitly. To do so, perform the following steps:

1. Find out the local transaction ID that corresponds to the transaction by running `sp_xa_scan_xact`, which supplies a list of transaction identifiers.

2. Use `sp_finish_xact` with a transaction identifier and a stat (either `commit` or `rollback`) to force the branch to complete.

Recovering from Log Failure

This section describes common Netscape Application Server (NAS) log failure scenarios and explains how NAS can recover from these scenarios.

Logs record the state of each transaction processed by NAS. If this data is completely lost, some transactions - those in the prepared state before the failure - can be left in an undesirable state. You may have to resolve such transactions manually by either aborting or committing them at the resource manager. The server can then be cold-started with new volume information and the system can be brought back online. However, the transaction manager provides means for recovering from some failures without resorting to a cold-start. These means are described in the following sections:

• Recovering from Log Disk Failure: Running Server

• Add a new mirror using the new disk.

• Recovering from Loss

Recovering from Log Disk Failure: Running Server

Log volumes in the transaction manager are backed up by physical volumes. Physical volumes are backed up by disks.

A disk failure can disable a log volume which can, in turn, disable the application server. Creating a mirror of the log volume helps increase the availability of the NAS machine. Without a mirror, disk failure disables the NAS machine. If a volume is mirrored, the NAS machine can continue normal operation even if the log volume fails.

If one of the disks backing up the log volume fails, you can perform the following steps to restart the application server and continue normal operation:

1. Query the logical volume to obtain a list of the mirrors backing it.

2. Query the failed physical volume to obtain the size of the volume.

3. Create a disk at least as large as the physical volume.

4. Remove the old mirror.

5. Add a new mirror using the new disk.

Recovering from Log Disk Failure: Stopped Server

If a log disk fails when the server is stopped, or when the server has crashed after a disk failure, you must restart the server in administration mode.

If you know which disk has failed, perform the following steps to recover from the failure:

1. Restart the server in administration mode.

2. Remove the bad mirror.

3. Add a new mirror to replace the faulty mirror.

4. Restart the server in normal operations mode.

If you do not know which disk has crashed, restart the server in normal operations mode. The server will not start properly, but it will print the name of the failed disk.

Recovering from Loss

You can obtain information about log volume configuration from the transaction manager's `restart` file. If the `restart` file is lost, you must cold-start the server, a process that can be undesirable; when a server is cold-started, existing volume information is lost. To avoid cold-starting the server, use the backup file (`restart.bak`) that the transaction manager creates by default. Place the `restart` and `restart.bak` files on separate disks. The transaction manager can recover from the loss of one of these files, but if both files are lost, the server must be cold-started.

Warning   Do not reuse log disks. A bug in the transaction manager prevents it from knowing whether a log disk is in use by another server. As a result, if a log disk is being used by one Java Server process (KJS1) and NAS Administrator attempts to use the same disk as a mirror for a second Java Server (KJS2), the transaction manager destroys the contents of the disk for KJS1.

Administering Multiple Netscape Application Servers

3

- • Configuring Multiple Servers

- • Administering Multi-Server Applications

- • Balancing User-Request Loads

- • Managing Distributed Data Synchronization

- • Troubleshooting

11

Configuring Multiple Servers

This chapter describes how to configure multiple Netscape Application Server (NAS) machines using NAS Administrator.

The following topics are included in this chapter:

- The Web Connector in a Multiple-Server Enterprise

- Distributed Data Synchronization and Load Balancing

- Multicast Communication

The Web Connector in a Multiple-Server Enterprise

The web connector plug-in directs users' requests to applications on your Netscape Application Server (NAS) machine. In a multiple-server enterprise, you can specify the application server where the web connector connects and logs web server requests. The application server you specify is the default server where the web connector exchanges requests and other application information. When the load balancer plug-in does not specify an alternate application server where application requests are forwarded, application requests are sent to this default server.

You can also specify the application server where the web connector sends the application request information for logging.

Configuring the Web Connector for Multiple Servers

When you use multiple NAS machines to support your enterprise application or applications, you must choose how to configure the web server to forward requests to NAS. These configuration options are provided by the web connector plug-in. Use the configuration scenarios described in the following table to help you decide how best to configure the web connector plug-in for your enterprise:

Configuration scenarios	What to do
One web server supporting multiple NAS machines without load balancing	It is assumed that the application is partitioned. Configure the web plug-in to forward requests to the application server that hosts the application objects that process the initial requests from the web browser. Use the other NAS machines to host the application components invoked by the objects on the first server.
Multiple web servers supporting multiple NAS machines without load balancing	If the application is not partitioned, configure each plug-in to forward requests to each appropriate NAS machine. If the application is partitioned, configure each plug-in to forward requests to a NAS machine that hosts the components that process the initial web browser requests. You can have multiple plug-ins connect to a single NAS machine.
One web server supporting multiple NAS machines with load balancing	The load balancing plug-in forwards application requests to the appropriate NAS machine. As a default, configure the web connector plug-in to forward requests to a NAS machine that either performs the best or hosts the application components that process the initial web browser requests.

Configuration scenarios	What to do
Multiple web servers supporting multiple NAS machines with load balancing	The load balancing plug-in forwards application requests to the appropriate NAS machine. As a default, configure the web connector plug-ins to forward requests to each NAS machine, or to the NAS machine that either performs the best or hosts the application components that process the initial web browser requests.

When you balance application loads, the web connector plug-in works with the load balancer plug-in to automatically distribute requests across multiple NAS machines. This prevents all requests from going to one NAS machine.

If you are not balancing application loads, you must determine where a web server forwards application requests.

Specifying the Application Server Where Requests Are Sent

In a multiple application server enterprise, you can specify where the web connector sends application requests.

If you have enabled load balancing, the load balancer plug-in first dictates where the request is forwarded. However, if you have not configured the load balancer plug-in to decide where to send the request, the web connector forwards the request to the NAS machine you specify.

To specify the NAS machine to which the web server connects, perform the following steps:

1. Open the Netscape Registry Editor by typing `kregedit` at the command line.

   The editor opens and displays the keys and values that apply to the NAS machine.

2. Open the following key:

   ```
   SOFTWARE\NETSCAPE\APPLICATION SERVER\4.0\CCSO\HTTPAPI
   ```

3. Double-click the GXIP String value.

   The Modify Value dialog box appears.

4. For the value data, enter the host IP address for the default NAS machine and click OK.

Specifying the Application Server Responsible for Logging

In a multiple-server enterprise, you can specify the application server used for web server logging.

In a single-server enterprise, the single server is the NAS machine where the web connector forwards application requests by default. For single-server enterprises, this value should not be changed.

In a multiple-server enterprise, the logging application server is the same server where the web connector sends application requests by default.

To specify the NAS machine responsible for logging, perform the following steps:

1. Open the Netscape Registry Editor by typing kregedit at the command line.

   The editor displays the keys and values that apply to the application server.

2. Open the following key:

   SOFTWARE\NETSCAPE\APPLICATION SERVER\4.0\CCSO\HTTPLOG

3. Double-click the Host String value.

   The Modify Value dialog box appears.

4. For the value data, enter the host IP address for the application server you want to perform web server logging and click OK.

5. Double-click the Port DWORD value.

6. For the value data, enter the port number for the Executive Server process of the same application server and click OK.

7. Close the editor tool.

Distributed Data Synchronization and Load Balancing

When you create a multiple application server enterprise, you must know if you want to enable load balancing across those servers. Applications that are distributed for load balancing might have dependencies on the distributed sychronization service of the application server if those applications require state and session management.

Distributed data synchronization is configured when you install Netscape Application Server (NAS). The installation script asks whether the server will participate in distributed data synchronization, as well as the host name and port number of the primary server. For more information about distributed data synchronization, see "About Distributed Data Synchronization" on page 235.

Configuring a Distributed Data Synchronization Environment

Once you install NAS on multiple machines, you must update the cluster keys of the servers participating in distributed data synchronization. This is done using the Netscape Registry Editor.

Updating the keys of servers in a cluster ensures that each server has the same information about the primary server, the immediate backups, and the priority in which other servers might become a primary server in the event of a server failure.

To configure a distributed data synchronization environment, see Chapter 14, "Managing Distributed Data Synchronization."

Multicast Communication

In a multiple-server enterprise, application servers communicate with each other, for purposes of load balancing and administration, using a multicast wide area network (WAN) service. The multicast service provides a virtual server to which all messages can be posted and distributed. The application servers use an N-Way multicast configuration that allows each server to send or receive the broadcast information. The following illustration shows how this network looks:
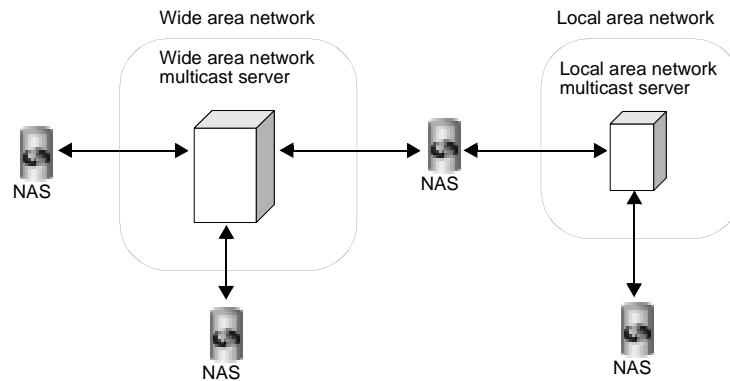
Multicast services are handled by the network hardware for all servers within a local area network (LAN). For these servers, you do not have to register or change the default multicast address. When you are implementing an enterprise in a wide area network, you should use a publicly registered multicast address that allows only your NAS machines to communicate with each other.

How Multicast Services Apply to Load Balancing

For load balancing, you can have all servers communicate with each other, or you can create islands of servers that only balance application loads between themselves. For example, an application in New York does not need to load

balance with the same application in Los Angeles. However, an application in Cupertino, Sunnyvale, and Santa Clara probably would share load responsibilities for all the users in the San Jose area.

For load balancing, multicast communication is determined by the Executive Server multicast address.

Multicast Communication

12

Administering Multi-Server Applications

This chapter describes how to administer applications on multiple Netscape Application Server (NAS) machines using NAS Administrator.

The following topics are included in this chapter:

- Hosting Applications Locally on Multiple Servers

- Hosting Partitioned Applications on Multiple Servers

- Hosting and Deploying Applications for Load Balancing

- Changing Attributes of Distributed Application Components

Netscape Application Server (NAS) Administrator allows you to simultaneously administer applications that are stored on multiple servers. Settings made to application components, such as Enterprise Java Beans (EJBs), distributed across multiple application servers are automatically updated across those servers. In addition, settings made to one NAS machine can be copied and applied to the other NAS machines in a group or the entire enterprise.

Using the administrator tool, you can view each NAS machine in the enterprise and make changes to one or more servers at the same time.

To host applications on multiple Netscape Application Server (NAS) machines, you can perform either of the following tasks:

- Distribute applications or parts of applications across two or more servers to specialize request and application processing.

- Duplicate application components on two or more servers to increase application performance with load balancing.

The more servers you have to work with, the greater your choice of application hosting configurations.

The following table describes three common ways to host an application on multiple NAS machines:

Hosting configuration	Description
Local	The application is installed on each NAS machine and uses multiple web servers to traffic requests to each server. The NAS machines do not communicate with each other.
Partitioned	Parts of the application are hosted on different NAS machines. Each server knows where the application components of the application are hosted on other servers and forwards requests to the appropriate server.
Distributed for load balancing	Parts or all of the application are duplicated on two or more NAS machines. You can then configure the servers to balance application-request loads.

Hosting Applications Locally on Multiple Servers

Hosting applications locally on multiple servers is the simplest of the three most common server configurations. In this configuration, you deploy the complete application on each NAS machine. If the application is already installed on a NAS machine, you can use the Deployment Manager to deploy the application to other servers.

Supporting applications locally on multiple servers means that each server stands alone. That is, the two or more NAS machines in the configuration do not communicate with each other. You must have at least one web server for each NAS machine. The following illustration depicts a local hosting configuration.

Web servers	Netscape Application Server	Database
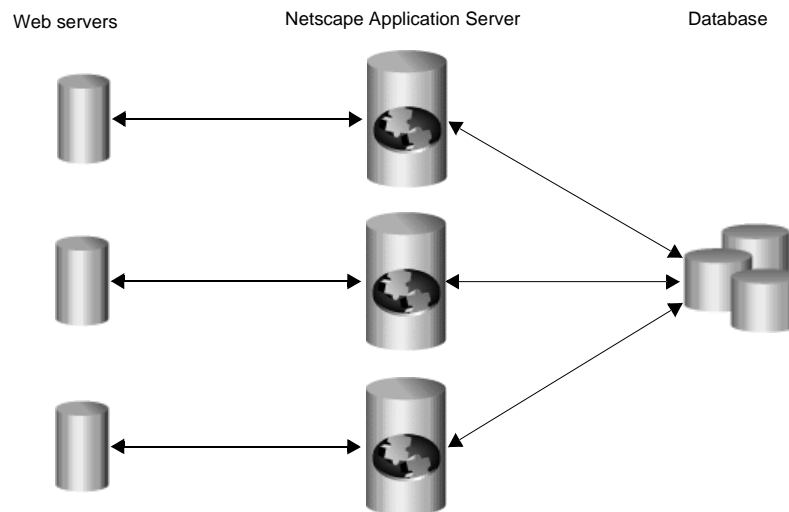
Local hosting requires that you configure each web connector plug-in to forward requests to the appropriate NAS machine.

Hosting Partitioned Applications on Multiple Servers

To partition an application, you must divide up the application components that make up an application. Application components are then hosted by separate NAS machines. Partitioning applications allows you to specialize the type of processing each NAS machine is performing.

For example, servlets responsible primarily for data access are I/O-intensive, while servlets responsible for performing calculations are CPU and active-memory intensive. To maximize your application's overall performance, you can partition the application to host these different types of servlets on separate NAS machines.

To configure a partitioned application, perform the following steps:

1. Deploy the complete application to all participating NAS machines using the Deployment Manager.

2. Enable load balancing, which will allow each server to find application components hosted on other servers.

3. Disable specific application components on a server-by-server basis, leaving no component enabled on more than one server.

   See "Disabling and Enabling Application Components" on page 207.

Alternately, if you want to allow load balancing for specific application components, you can leave them enabled on more than one server.

To partition an application, perform the following steps:

1. Launch the Deployment Manager.

2. Deploy the JAR files containing the application components you want to load balance to each application server participating in load balancing.

   When you deploy a JAR file to an application server, the file's application components are registered with a single directory server shared by multiple application servers in a directory server cluster. All application servers in this cluster can "see" the registered application components. However, you must still deploy the JAR file to each application server participating in load balancing.

3. Open the Application window of NAS Administrator.

4. In the left pane of the Application window, click the server whose application components you want to partition.

   Deployed application components appear in the right pane of the Application window.

Be sure the application components that appear in the right pane of the Application window are actually installed on the NAS machine you selected in step 4:

1. Select an application component in the right pane of the Application window.

2. Click the Application Component Properties button.

A dialog box appears displaying the application servers where the component is installed. If the selected NAS machine is not listed, you must deploy the JAR file containing the necessary application components to that machine.

3. Click OK to dismiss the dialog box.

5. Click Apply Changes for each NAS machine to which you make changes.

Disabling and Enabling Application Components

Disabling a component of your application (such as a servlet) stops users from accessing that component. Current requests are allowed to finish when a component is disabled, but no new requests are accepted until the component is reenabled.

To disable an application component, perform the following steps:

1. On the NAS Administrator toolbar, click the Application button to open the Application window.

2. In the left pane of the Application window, double-click the server where the application to be upgraded resides.

3. Select the folder containing the application components to disable.
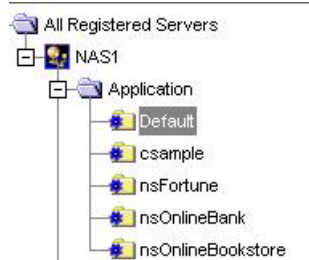
4. In the right pane of the Application window, select the component to disable.

5. Click the Enabled checkbox to deselect and disable it.
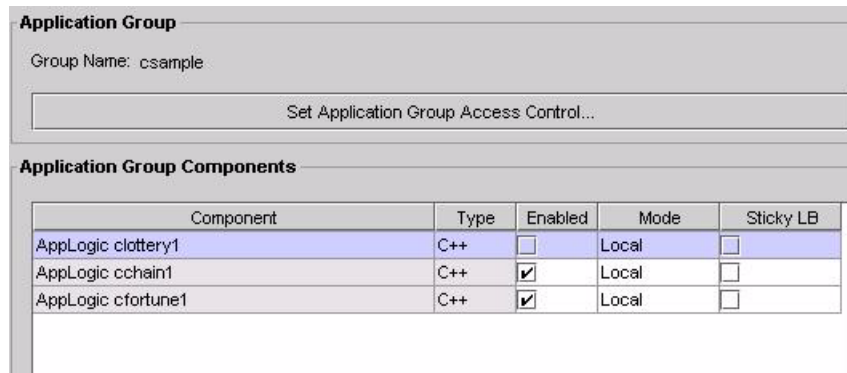
6. Repeat steps 2 through 5 for each application component you want to disable.

7. Click Toggle Enabled if you want to disable all the application components in a group.

    To enable application components, click their corresponding Enabled checkboxes to select them.

8. Click Apply Changes to save your changes to your NAS machine.

Hosting and Deploying Applications for Load Balancing

Balancing application-request loads, or load balancing, differs from partitioning applications. Load balancing requires you to place one or more copies of an application component on multiple NAS machines rather than simply dividing an application's components among multiple servers (or partitioning the application). You then configure each server, allowing it to find application components on other servers.

When you deploy an application, you must decide if you want to configure the application for load balancing and, if so, how you will configure it. Choose among the following load balancing configurations:

- Balancing application loads between the deployment server and the destination server or servers (joining the distributed servers).

- Balancing loads only between the destination servers, if deploying to more than one NAS machine.

- Deploy the application locally to the server or servers (no load balancing).

Select the configuration that is most useful for your current scenario; for example, you might have three NAS machines used for testing applications. Your production environment, where users' requests are actually processed, also consists of three NAS machines. Because the application components could be different between the two groups of servers, you do not want to enable application load balancing. Therefore, when you deploy an application from the test servers to the production servers, you should choose only to balance the loads between the destination servers.

Later, should you scale the enterprise to include three more NAS machines in the production group, you can join all the servers in that group when deploying the applications from one of the existing production servers to the new servers. The application loads are then balanced between the existing servers and the new servers.

The following illustration depicts a load-balancing distribution among the destination servers only:

Initial deployment from
Netscape Application
Builder to test servers

Deployment from test
servers to runtime servers

Application
Builder

NAS   NAS   NAS

Test group

NAS   NAS   NAS

Runtime group

Load balancing occurs discretely
within the test group

Load balancing occurs discretely
within the runtime group
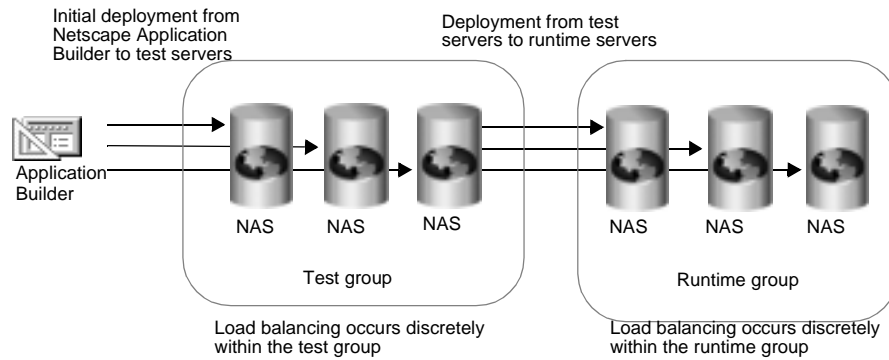
The next illustration depicts a joining of servers when adding new servers to a group and deploying an application to those servers with the join option.

Deployment to new servers

NAS   NAS   NAS

Original servers

NAS   NAS   NAS

New servers

When deploying and joining the servers, load balancing occurs between the original
and the new servers
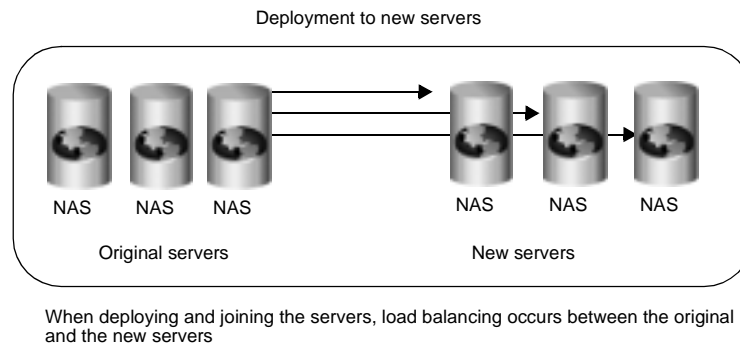
If you choose a local distribution during deployment, no application-request load balancing occurs between any of the servers.

Changing Attributes of Distributed Application Components

When you change such attributes as enabled sticky load balancing for an application component that is distributed across multiple servers, those changes replicate themselves on the servers where that component is hosted. Changing the distribution level of installed application components is useful if you previously installed an application locally, but now want to distribute the application for load balancing. You can also disable load balancing by changing a distributed application to a local configuration on the specified server.

If you change a component from a distributed state to a local state on one server, each server that hosts that component ceases to balance loads with the server where the distribution was set to local.

For example, an application component called ShopCart is distributed across servers A, B, and C. Should you decide to run ShopCart locally on server A, but continue to allow it to run in a distributed state across servers B and C, each server (A, B, and C) is automatically updated so that requests for ShopCart are no longer passed to server A from servers B and C. Instead, requests for ShopCart made to servers B or C are passed only between those two servers. All requests for ShopCart made to server A are processed only by server A.

To change the distribution level for an application component, perform the following steps:

1. Open the Application window of NAS Administrator.

2. In the left pane of the Application window, double-click the server for which you want to change application settings.

3. Select the group of application components you want to modify.
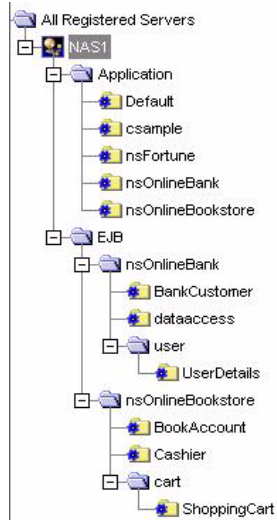
4. In the right pane of the Application window, select each application component for which you want to change the distribution level.

5. In the Mode column, change the distribution level.

- If you are changing the distribution level for all components in the selected group, click Toggle Mode. All application components are updated simultaneously.

- If you are modifying the Mode from Local to Distributed, you must modify the application properties to specify across which NAS machines load balancing is to occur.

If you are modifying the Mode from Distributed to Local, there is nothing more you need to do.

When you change an application component's Mode to Distributed, all registered servers appearing in the left pane of the Application window are added to that application component's server list. You can access the server list by clicking the Application Component Properties button.

6. In the left pane, under Registered Servers, choose which NAS machines will participate in load balancing of the selected application component. The application component must be installed on each NAS machine participating in load balancing.

7. If you need to register additional application servers, go to the File menu and choose New, then choose Server.

8. Repeat these steps for each application component.

9. Click Apply Changes to save your changes to the NAS machine.

Changing Attributes of Distributed Application Components

13

Balancing User-Request Loads

This chapter describes load balancing, which optimizes the ability of each Netscape Application Server (NAS) to process users' requests by keeping those requests balanced among several NAS machines.

This chapter contains the following topics:

- How Load Balancing Works

- Using the Load Balancer Plug-in

- What Is "Sticky" Load Balancing?

- Adjusting Load Balancing Weight Factors

- Adjusting Update and Broadcast Intervals

- Changing the Multicast Host Address for Load Balancing

How Load Balancing Works

Load balancing sends user requests to the server with the least load, directing requests away from servers too busy to handle additional requests.

For example, if you find that many users access an application during peak usage hours, you can duplicate the application's components, such as AppLogics and servlets, on several Netscape Application Server (NAS) machines and enable load balancing. As one NAS machine reaches its optimal handling capacity, subsequent requests are sent to another NAS machine with duplicate application components. With multiple servers handling requests, you can increase response time.

The distribution of requests across multiple NAS machines is monitored and assigned by a load-balancing service built into each instance of NAS. The load-balancing service determines which NAS machine is best suited to handle a request by comparing servers based on the following two values:

- The Server Load value

  The Server Load value is calculated from various system criteria, such as the CPU load and amount of memory thrash. These criteria are indicative of how busy a NAS machine is and, therefore, how well it might handle additional requests. For more information about these values, see "Adjusting Weight Factors for Server Load Criteria" on page 223.

- The Application Component Performance value

  The Application Component Performance value is calculated from various application component criteria, such as the number of times the component was run on a particular server or whether the component's results were cached. These criteria are indicative of how well the component is running on a particular NAS machine. For more information about these values, see "Adjusting Weight Factors for Application Component Performance Criteria" on page 226.

Exchanging Load Balancing Information

There is one load-balancing service in each instance of NAS. Each load-balancing service calculates the Server Load and Application Component Performance values for the server that hosts the load-balancing service. The load balancer then broadcasts, at the broadcast interval, its calculated values to the load-balancing services of the other NAS machines in the enterprise.

You can set the broadcast time intervals for each load-balancing service. For more information, see "Adjusting Update and Broadcast Intervals" on page 230.

Each load-balancing service calculates which NAS machine is currently able to most efficiently process new requests. This decision is made for each distributed application component. A list of available servers for each component is maintained by the load-balancing service.

The NAS machine best suited to process requests for an application component is ranked first in the list, followed by, in order, the other servers that are able to process requests for that component. The load-balancing service updates this order each time it receives a broadcast from another NAS machine or re-calculates its own server's values. The other servers are backups to the first server and the first server can forward a request to one of the other servers at any time.

NAS machines can pass requests to more capable servers up to a maximum number of hops. When an application component reaches that maximum number, it must be processed by the server to which the component was last passed. For information about setting the maximum number of hops, see "Adjusting Update and Broadcast Intervals" on page 230.

Load Balancing and Network Traffic

Only NAS machines that have enabled load balancing exchange load balancing information. Each server sends about 128 bytes of information when it broadcasts its calculated values. This value does not include network packet overhead. If a request is routed to another NAS machine, the hop size is at least 16 bytes, but could be more depending on the number of `ValList` variables.

All NAS machines in your enterprise that are configured for load balancing exchange Server Load values. In contrast, the Application Component Performance values are exchanged only with NAS machines hosting and distributing requests for the same application components.

Requirements for Load Balancing

Before your application is load balanced, the following requirements must be met:

- The application's components must be duplicated on at least two NAS machines or on every NAS machine that is to participate in load balancing.

- The distribution levels for the application components must be distributed for either specific NAS machines or globally to all NAS machines in the enterprise.

For information about enabling load balancing, see "Hosting and Deploying Applications for Load Balancing" on page 209.

Using the Load Balancer Plug-in

The load balancer plug-in is a component of the web connector plug-in. The load-balancing service within an instance of Netscape Application Server (NAS) communicates with the load balancer plug-in to provide information to the web connector about where to forward an application request.

The web connector plug-in defaults to sending application requests to one NAS machine. In a multiple-server enterprise where load balancing is being used, sending all requests to one server is not effective. It creates a potential bottleneck. Using the load balancer component, the web connector can determine, for each request, which NAS machine should process that request, thereby eliminating the bottleneck.

The load balancer plug-in is installed and configured automatically with the web connector plug-in. For more information about configuring the web connector plug-in, see Chapter 8, "Configuring the Web Connector Plug-In."

What Is "Sticky" Load Balancing?

If requests within the same session are processed by more than one Netscape Application Server (NAS) machine or process, session information that cannot be distributed is lost. Therefore, certain application components are marked for session or "sticky" load balancing and processed on the same server, thereby eliminating the loss of session information.

When an application component is marked for sticky load balancing, it is processed by the same NAS machine or process where it is initially invoked. For example, an application component called ShopCart is duplicated on two application servers for load balancing, Server A and Server B. If ShopCart is invoked by Client 1 on Server B, all subsequent sticky requests for that ShopCart from Client 1 are processed on Server B only. In other words, ShopCart "sticks" to Server B for the duration of Client 1's session. However, at the same time, Client 2 may access ShopCart on Server A without affecting Client 1's use of ShopCart on Server B. This maintains the integrity of state and session information for an application component that does not distribute session information.

When to Use Sticky Load Balancing

Sticky load balancing is necessary for application components that have interdependencies, but are running in a distributed environment. Such application components typically have the following characteristics:

- originally written to run on one machine

- depend on session information to run properly

- wrapped, not rewritten, to run in a NAS environment

For example, a heavily used, pre-existing application is ported to run on NAS. Because the application is heavily used, it is distributed across several NAS machines to increase availability. When a user makes a request that invokes a sticky application component, the load-balancing service determines which NAS machine should handle that request. Once that server is chosen, all subsequent requests that use sticky application components are handled by that server. If that server becomes burdened with many users' requests, the load

balancer forwards new requests to another NAS machine and that server processes all new session requests. This maintains an effective degree of load balancing.

Enabling Sticky Load Balancing

Enable sticky load balancing if there are multiple processes running on a NAS machine and certain application components cannot distribute session and state information.

To enable sticky load balancing, perform the following steps:

1. On the NAS Administrator toolbar, click the Application button to open the Application window.

2. In the left pane of the Application window, select the server where you want to enable sticky load balancing.

3. Open the application group that contains the application component or components for which you want to enable sticky load balancing.
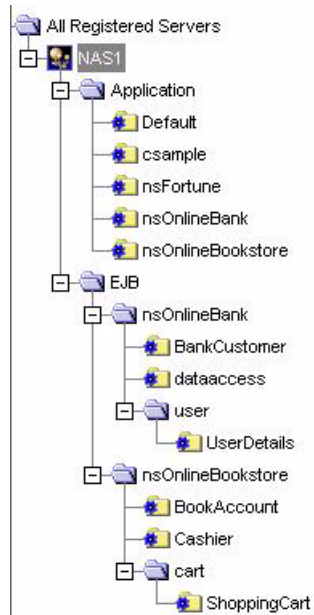
4. In the right pane of the Application window, select the application component for which you want to enable sticky load balancing.

5. In the Sticky LB column, click the checkbox for the selected application component.

Application Group

Group Name: Default

Set Application Group Access Control...

Application Group Components

Component	Type	Enabled	Mode	Sticky LB
Bean GXApp.nsOnlineBookstore.account.IBookA...	Java	✔	Local	☐
Bean GXApp.nsOnlineBank.user.IUserDetails	Java	✔	Local	☐
Bean GXApp.nsOnlineBank.dataaccess.IDataAc...	Java	✔	Local	✔
Bean GXApp.nsOnlineBookstore.cart.IShoppingC...	Java	✔	Local	☐
Servlet nsOnlineBookstore_Bookstore	Java	✔	Local	☐
Bean GXApp.nsOnlineBookstore.cashier.ICashier	Java	✔	Local	☐
Bean GXApp.nsOnlineBank.customer.IBankCusto...	Java	✔	Local	☐

Sticky load balancing is turned on for the selected component.

6. Repeat steps 4 and 5 for each application component where you want to enable sticky load balancing.

7. Click Toggle Sticky LB to select or deselect all Sticky LB checkboxes.

Selecting a Load Balancing Method

When configuring your server for load balancing, you must choose a load balancing method. Each method provides a different way to decide "who" makes the load balancing decisions. In other words, are load balancing decisions left to the server itself or does the web server plug-in make the decisions?

If load balancing decisions are left to Netscape Application Server (NAS), the application server uses a combination of hardware resource profiles (including CPU load and disk I/O) and Request Execution profiles (including result caching and servlet execution rate) to load balance individual requests. Server and request statistics are communicated from one NAS machine to another in a cluster via multicasting. Multicasting gives more control to the administrator, and is suitable for sophisticated scenarios.

If load balancing is left to the web server plug-in, the only factor considered in load balancing is response time as seen by the plug-in. The plug-in gathers response-time data as requests enter and leave the plug-in, then uses this data to load balance future requests. Since there is no inter-process communication involved and per-request resource usage is not factored into the load-balancing decisions, this is a simpler method than server-based load balancing.

Leaving load balancing decisions to the web server plug-in requires you to choose between two types of plug-in methods: load balancing by per-component response time or load balancing by per-server response time. Per-component response time is a NAS machine's average response time for a specific application component. Per-server response time is a NAS machine's average response time across all the application components that machine processes.

Use the per-component method to enable richer, more detailed load balancing decisions, keeping in mind that this scenario involves a little more overhead than the per-server method. The per-component method is best suited to situations where one application component has a response time that varies widely from server to server, while the per-server method is best in situations where an application component has a similar response time from server to server.

Adjusting Load Balancing Weight Factors

If you decide Netscape Application Server (NAS) -- not the web server plug-in --will make the load-balancing decisions for your enterprise, the load-balancing service then decides which NAS machine should process a request based on the weight factors you specify for the Server Load and Application Component Performance criteria. You set these factors using the NAS Administrator tool's Load Balancing window. When determining weight factors, you must decide how important each criteria is for keeping your applications running optimally.

The weight factors in NAS Administrator are initially set to default values based on the most typical applications that run on a NAS machine. You can adjust these factors for either Server Load criteria or Application Component criteria to optimize your specific application.

Adjusting Weight Factors for Server Load Criteria

The Server Load value quantifies the load on a NAS machine while the server is processing users' requests. This value is calculated for each NAS machine by the load-balancing service within the respective server. You can adjust the weight factors for Server Load criteria to optimize how application requests are distributed across multiple NAS machines based on system resources.

The Server Load value is used as one of the criterion for calculating the Application Component Performance value. The Server Load criteria are described in the following table:

Server load criteria	Description
CPU Load	The average percentage of time all processors in a computer are in use.
Disk Input/Output	The rate at which the system is issuing Read and Write operations to the hard disk drive.
Memory Thrash	The number of pages read from or written to the hard disk drive to resolve memory references to pages that were not in memory at the time of the reference.

Server load criteria	Description
Number of Requests Queued	The number of user and application requests a server is currently processing.
Server Response Time	Average response time from a specific server for all application components.

Each Server Load criterion is multiplied with a weight factor you set. That value is averaged with the other values to determine the final Server Load value. This value is then used as one of the Application Component Performance criteria.

To adjust the weight factors for Server Load criteria, perform the following steps:

1. On the NAS Administrator toolbar, click the Load Balancing button to open the Load Balancing window.

2. Click the Server Load tab.

The following window appears:

3. In the left pane, select the server for which you want to adjust the weight factors.

4. In the right pane of the Load Balancing window, select a load balancing method from the Load Balancing drop-down box.

   Select User Defined Criteria (NAS driven) if you want the server to make load balancing decisions. You can then adjust the weight factors as your enterprise requires.

   Select Per Component Response Time (Web Connector Driven) or Per Server Response Time (Web Connector Driven) if you want the web connector plug-in to make load balancing decisions based on response time. Since the web connector does not use weight factors to make load-balancing decisions, skip the next step.

   See "Selecting a Load Balancing Method" on page 222 for more information.

5. In the right pane of the Load Balancing window, use the sliding scale markers to adjust the weight factor for each criterion.

   The grand total of all weight factors must equal 100.

6. When finished, click Apply Changes to save the settings.

Adjusting Weight Factors for Application Component Performance Criteria

The Application Component Performance value represents the performance of the application components running on a NAS machine. This value is calculated for each application component participating in load balancing. Load balancing then occurs on an application component basis and increases distribution.

The Application Component Performance value includes five application criteria. The load-balancing service compares NAS machines based on the weight factor you assign for each application criterion. The server with the highest total value is chosen to process requests for that application component. The Application Component Performance criteria are described in the following table:

Application Component Performance Criteria	Description
Server Load	The value calculated for all Server Load criteria.
Cached Results Available	A flag that signals whether the results of the application component are cached. A user's request is typically processed faster when the application component's results are cached.
Lowest Average Execution Time	The time with which an application component takes to run on each NAS machine.
Most Recently Executed	The server that most recently ran an application component. The system on which the server is running might have cached application data, resulting in a faster execution time if that component were to be run again soon.
Fewest Executions	The number of times the application component ran on a NAS machine. The goal of load balancing is to equally distribute requests among all servers in the enterprise. Therefore, the server that has run the application component the least number of times is most preferred.
Application Component Response Time	Average response time from a specific server for a specific application component.

Each application criterion is multiplied by a weight factor you set. Each value is then averaged to determine the final Application Component Performance value. The final value is used by the load-balancing service to determine which NAS machine is best able to handle new users' requests.

To adjust the weight factors for Application Component Performance criteria, perform the following steps:

1. On the NAS Administrator toolbar, click the Load Balancing button to open the Load Balancing window.

2. Click the Application Component Criteria tab.

The following window appears:

3. In the left pane, select the server for which you want to adjust the weight factors.

4. In the right pane of the Load Balancing window, select a load balancing method from the Load Balancing drop-down box.

Select User Defined Criteria (NAS driven) if you want the server to make load balancing decisions. You can then adjust the weight factors as your enterprise requires.

Select Per Component Response Time (Web Connector Driven) or Per Server Response Time (Web Connector Driven) if you want the web connector plug-in to make load balancing decisions based on response time. Since the web connector does not use weight factors to make load-balancing decisions, skip the next step.

See "Selecting a Load Balancing Method" on page 222 for more information.

5. In the right pane of the Load Balancing window, use the sliding scale markers to adjust the weight factor for each criterion.

The grand total of all weight factors must equal 100.

6. When finished, click Apply Changes to save the settings.

Adjusting Update and Broadcast Intervals

You can set the time at which a NAS machine updates the Server Load and Application Component Performance criteria. If these values change frequently and drastically, it is useful to update the values often. Unfortunately, you increase the amount of work the NAS machine is doing by updating values frequently. You can save server resources by increasing the time between updates if the criteria values do not change often.

This theory applies to setting the broadcast intervals, as well; if values are changing often and drastically, the broadcast intervals should be short, updating servers often. This increases network traffic load, so it is important to find an optimal balance.

Broadcast and update intervals are relative to the Base Broadcast/Update Interval. This is the interval at which the load-balancing service "wakes up" and performs any updates, checks to see if any updates were received, and broadcasts any new values.

Broadcast and update intervals that are even multiples of the base interval are invoked when the load-balancing service "wakes up." In other words, if the base value is 300 seconds, and the Server Load and Application Component Criteria broadcast intervals are at 900 seconds each, these values are broadcast every third time the load-balancing service "wakes up." The other two times the load-balancing service awakens, it reevaluates the distribution order based on whether it received any updates from other NAS machines.

You can set update and broadcast intervals for several entities, as described in the following table:

Set interval for	Description
Base Broadcast/Update Interval	The interval at which the load-balancing service "wakes up."
Application Component Criteria	The interval at which the load-balancing service broadcasts the Application Component Performance value.
Server Load Criteria	The interval at which the load-balancing service broadcasts the Server Load value.
Server Load	The interval at which the load-balancing service updates the Server Load value.

Set interval for	Description
CPU Load	The interval at which the load-balancing service updates the CPU Load value.
Disk Input/Output	The interval at which the load-balancing service updates the Disk I/O value.
Memory Thrash	The interval at which the load-balancing service updates the Memory Thrash value.
Number of Requests Queued	The interval at which the load-balancing service updates the Number of Requests Queued value.
Max Hops	The maximum number of times a request is allowed to be passed between servers.

To adjust the update and broadcast intervals, perform the following steps:

1. Click the Load Balancing button on the NAS Administrator toolbar to open the Load Balancing window.

2. Click the Advanced Settings tab.

The following window appears:

3. In the left pane of the Load Balancing window, select the server for which you want to adjust the advanced settings.
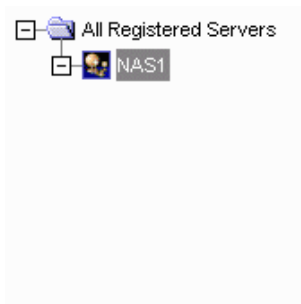
4. In the right pane of the Load Balancing window, select a load balancing method from the Load Balancing drop-down box.

   Select User Defined Criteria (NAS driven) if you want the server to make load balancing decisions. You can then adjust the update and broadcast intervals as your enterprise requires.

   Select Per Component Response Time (Web Connector Driven) or Per Server Response Time (Web Connector Driven) if you want the web connector plug-in to make load balancing decisions based on response time. Since the web connector does not use weight factors to make load-balancing decisions, skip the next two steps.

   See "Selecting a Load Balancing Method" on page 222 for more information.

5. In the right pane of the Load Balancing window, under each interval parameter, set the time as a multiple of the base time for that parameter.

6. In the Max Hops text area, specify the maximum number of times an application component is passed between servers.

7. When finished, click Apply Changes to save your changes.

Changing the Multicast Host Address for Load Balancing

Change the multicast server host address and port number to balance application loads across networks, such as across cities. Within a network, the default address does not need to be changed unless you are experiencing a conflict.

To change the multicast host address, perform the following steps:

1. Open the Netscape Registry Editor by typing `kregedit` at the command line.

   The editor opens and displays the keys and values that apply to NAS.

2. Open the following key:

   ```
   Software\Netscape\Application Server\4.0\GMS\KES
   ```

3. Double-click the `MCastHost` String value.

   The String editor dialog box appears.

4. For the value data, specify the IP address for the new host and click OK.

5. Double-click the `MCastPort` DWORD value.

   The DWORD editor dialog box appears.

6. For the value data, specify the port number for the new host and click OK.

7. Close the editor.

   The new multicast address is in effect.

Managing Distributed Data Synchronization

This chapter describes how to group Netscape Application Servers into data synchronization clusters.

The following subjects are described in this chapter:

- About Distributed Data Synchronization

- How Failover Keeps Data Accessible

- What Is a Cluster?

- Setting Up and Managing Clusters

- Using the Administrator to Configure Clusters

About Distributed Data Synchronization

Distributed data synchronization maintains the integrity of shared state and session information across multiple Netscape Application Server (NAS) machines. This is crucial for partitioned and distributed applications that are hosted on multiple NAS machines.

In most enterprises, several NAS machines support one or more distributed applications. For such distributed applications to run successfully, each server must have access to the pertinent information for that application, such as state and session information.

Support for this distribution of information is provided through a system-level distributed data synchronization service that is built into NAS.

How Failover Keeps Data Accessible

The distributed data synchronizer is a system-level service that controls how distributed data, such as application session information, is maintained and made accessible across multiple Netscape Application Server (NAS) machines.

Each NAS machine is made up of the following four "engines:"

- Administrative Server (KAS) – An Administrative Server brings up and monitors the other engines and makes sure that any engines that fail are brought up again.

- Executive Server (KXS) – Only an Executive Server can be the primary synchronization engine (the synchronizer) for a NAS cluster.

  In a cluster of NAS machines, one of the Executive Servers maintains the distributed (synchronized) information and sets up server roles for all the other servers participating in the cluster. All engines in a cluster know how to access this primary engine and the information that is on this primary engine.

- Zero or more Java Servers (KJS)

- Zero or more C++ Servers (KCS)

  If the Java or C++ engine on a Netscape Application Server fails, the Administrative Server simply restarts the KJS or KCS. However, if the Executive Server fails, the Administrative Server performs the following actions:

  - Brings the Executive Server back up in the currently appropriate role. This role is determined in synchronization with other Executive Servers in the cluster, and is not necessarily the previous role.

- Brings down the Java and C++ engines.

- Brings the Java and C++ engines back up.

What Is a Cluster?

A cluster is a group of Netscape Application Server (NAS) machines that synchronizes data. Servers in a cluster are connected by the same network.

Data that is shared by all the NAS machines in a cluster is stored in Netscape Directory Server. Each NAS machine in your cluster should share one Directory Server; if the NAS machines in your cluster do not share a single Directory Server, cluster settings must be copied from one Directory Server to another so each server has access to identical cluster information. This defeats the purpose of Directory Server, which is designed to simplify information storage by storing the data shared by servers in your enterprise in a central location.

Note You access cluster information using the Netscape Registry Editor. You cannot edit a NAS machine's cluster settings using the Windows NT regedit tool or any other editor tool. Each folder in the Netscape Registry Editor tree structure, which looks similar to Windows NT's registry tree structure, is referred to as a kregedit key or cluster key in this document.

Setting Up Data Synchronization

To set up data synchronization between servers, you must first decide what general role each server performs in the cluster. Then you can edit each cluster entry to set up the server roles and to register the cluster with the synchronizer service. Finally, start each NAS in the order that is determined by server roles.

Synchronization Server Roles

Each server that participates in data synchronization can be set up to fill any one of the roles described in the following table.

Server role		Description
Sync Server		Any NAS machine that can potentially become a Sync Primary. The Sync Server category contains the Sync Primary, Sync Backups and Sync Alternates.
		All Sync Servers are listed in the `SyncServers` key of kregedit.
	Sync Primary	The server that is the primary data store, to which all other cluster members communicate for the latest distributed data information.
		The first NAS to be started in a cluster must be a Sync Server, and that Sync Server becomes the Sync Primary for the cluster simply because it is started first.
	Sync Backup	Any number of Sync Servers, up to a maximum number (`MaxBackups`) set by you, that mirrors the information on the Sync Primary. Because each Sync Backup increases the load on the cluster, weigh safety against performance impacts when deciding how many backups to assign.
		If the Sync Primary becomes inaccessible, the Sync Backup with the highest priority (which is the lowest integer value) relative to other Sync Backups becomes the next Sync Primary.
	Sync Alternate	A server listed in the `SyncServers` kregedit key that is eligible to become a Sync Backup. If the number of Sync Backups falls below the set maximum, the Sync Alternate with the highest priority relative to other Sync Alternates is promoted to Sync Backup.
		Each Sync Alternate performs work similar to that of a Sync Local until the Sync Alternate is promoted to Sync Backup.

Server role	Description
Sync Local	A server that uses data synchronization services, but is not eligible to become a Sync Primary, Sync Backup, or Sync Alternate. Sync Locals can use, create, and destroy all distributed data, but are never responsible for maintaining that data. Sync Locals are not listed in the `SyncServers` kregedit key. However, the `SyncServers` list in every registry in the cluster contains identification and priority information for each of the Sync Servers in the cluster. Each Sync Local contacts each of the servers listed in its `SyncServers` kregedit key until the Sync Local finds the Sync Primary, at which time the Sync Local becomes active in the cluster. If the Sync Local goes through its entire `SyncServers` kregedit key without finding the Sync Primary, the Sync Local assumes that the cluster is down, and acts as a local server. Sync Locals communicate only with the Sync Primary, and the other servers in the cluster are not aware of them.

How a Cluster Communicates

Servers in a cluster communicate using the GXCONN communication protocol. However, before the servers in a cluster can communicate with each other, each server has to know what cluster it belongs to. NAS becomes an active part of a cluster when you map its synchronizer to the cluster. This procedure is described in "Mapping the Synchronizer to the Cluster" on page 250.

When an application component requests "write" access to a distributed data source, the write occurs first on the Sync Primary. When the data changes on the Sync Primary, the Sync Primary immediately updates the Sync Backups.

Although you can define as many clusters as you like, the synchronizer for each NAS machine can be mapped to only one cluster at a time.

Information Flow Within a Cluster

Sync Backups, Sync Alternates, and Sync Locals communicate with the Sync Primary in a star configuration, as shown in the following illustration:
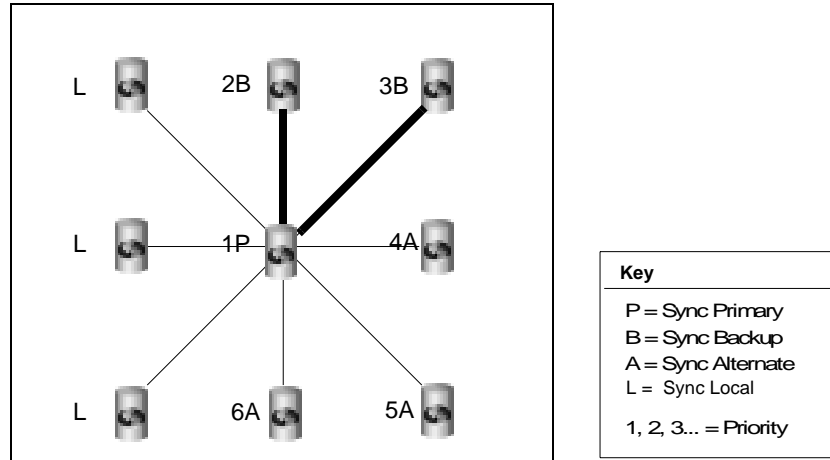
In this illustration, notice that all servers are communicating with the Sync Primary, although the Sync Backups communicate with it most closely. Also, notice that no Sync Local is assigned a priority number.

Note also that the illustration is an ideal representation of a cluster that has probably just started and has not experienced failover, in that the priority numbers correspond gracefully with the currently assigned roles.

Setting Up and Managing Clusters

Before you set up and begin managing clusters, review the following steps, which provide an overview of the general procedure. More specific procedures for setting up and managing clusters are described in subsequent sections.

1. Decide which servers will participate in a synchronization group (cluster), and which of those servers will be Sync Servers, eligible to act as the Sync Primary and as Sync Backups, and which will be Sync Locals.

2. Edit the kregedit keys under `Clusters` and `ClusterName` on one of the Sync Servers. Duplicate the `ClusterName` edits to the registries of all the other servers in the cluster (including the Sync Locals). You need not duplicate edits to the `Clusters` key since this information is stored in a centrally located Directory Server.

   1. Create the kregedit keys that will contain synchronization information, if necessary.

   2. Edit the `SyncServers` kregedit key to contain identification information and the priority setting for each Sync Server in the cluster. Often, the larger and more powerful servers are chosen to be the highest-priority Sync Servers.

   3. Set the `MaxBackups` kregedit key to the number of Sync Backups. Sync Backups are servers that duplicate the data on the Sync Primary.

3. Enter the name of the cluster in the `ClusterName` key.

   Make sure that the kregedit keys under `ClusterName` are identical on all servers in the cluster, including the Sync Locals. Each `SyncServers` kregedit key must list the same Sync Servers with the same priority numbers, or the cluster will not function properly.

4. Start the Sync Server that will be the Sync Primary. The server that you want to be the Sync Primary must always be the first server to be started in the cluster, and it becomes the Sync Primary simply because it started first.

5. After starting the Sync Primary, start the other servers (including the Sync Locals). Although the starting order is not mandated after the Sync Primary starts, it is a good practice to start the Sync Servers in priority order, and then to start the Sync Locals.

   1. Start the servers that will become the Sync Backups, up to the value of `MaxBackups`. By default, the next servers listed in the `SyncServer` key that start, up to the value stored under the `MaxBackups` kregedit key, will become the Sync Backups.

   2. After `MaxBackups` number of servers have started, remaining Sync Servers that start become Sync Alternates.

   3. All servers not listed in the `SyncServers` kregedit key become Sync Locals. Sync Locals are part of the cluster simply because each is mapped to the cluster and the `SyncServers` kregedit key on each contains a list of all the Sync Servers in the cluster.

Determining Sync Server Priority

The specific procedure for setting priority is covered in "Modifying the Default Cluster for Fast Cluster Setup" on page 245 and "Defining a Cluster" on page 252. The following section discusses general priority issues and gives a comprehensive example of cluster coordination.

Priority is indicated by an integer value that is set in the `SyncServers` kregedit key. The lower the value, the higher the priority, so the server assigned a value of 0 has the highest possible priority. The highest acceptable value, and so the lowest priority value, is 65,535.

Priority values are used only to select between Sync Servers in the same status (either between a group of Sync Backups or between a group of Sync Alternates). Only the order in which instances of NAS are started, not priority, determines which server should be the Sync Primary and which Sync Servers will start out as Sync Backups or Sync Alternates.

A Sync Local is not assigned a priority because it is not eligible to become a Sync Server, so a Sync Local cannot become a Sync Primary, Sync Backup, or Sync Alternate.

Which Sync Server becomes the Sync Primary in a cluster is determined simply by which Sync Server is started before any of the other servers. The next Sync Servers that start, up to the value in `MaxBackups`, become Sync Backups. When the Sync Primary fails, the Sync Backup with the highest priority, which is the lowest integer value, becomes the new Sync Primary.

When a Sync Backup becomes a Sync Primary, the number of Sync Backups falls below the value of `MaxBackups`. To restore the number of Sync Backups, the Sync Alternate with the highest priority becomes a Sync Backup.

Example: Coordination Within a Seven-Server Cluster

The following example illustrates cluster coordination through server roles, and the part that priority plays in determining those roles. As you trace the role changes through the example, keep in mind that server fallibility has been purposely exaggerated to provide many scenarios.

Although not required, you can ease cluster maintenance by assigning the highest priority to the NAS machine that you will start as the Sync Primary, and the next highest priorities (in descending order) to the Sync Backups. Be aware that the cluster in this example does not do this. Also, notice that this cluster does not follow the recommended practice of starting the servers in priority order.

Assume a seven-server cluster with NAS machines that are numbered 0 to 6. Servers 0 through 4 are Sync Servers that are assigned the same priorities as their server numbers (for example, server 0 has a priority of zero). Servers 5 and 6 are Sync Locals. `MaxBackups` for the cluster is set to two.

- Server 3 is brought up first, so it becomes the Sync Primary.
- Server 4 is started next, and it becomes a Sync Backup.
- Server 6 is started next, and it is a Sync Local.
- Server 1 is started next, and it becomes a Sync Backup.
- Server 2 is started next, and it becomes a Sync Alternate.
- Server 5 is started next, and it is a Sync Local.
- Server 0 is started next, and it becomes a Sync Alternate.

Server 3 fails and goes down. Between the two Sync Backups, server 4 and server 1, server 1 has the higher priority (lower integer value) and it becomes the new Sync Primary. This leaves server 4 as the only Sync Backup.

Because `MaxBackups` is set to two, one of the Sync Alternates is converted to a Sync Backup. Server 0 becomes the new Sync Backup because it has a higher priority than the other remaining Sync Alternate, server 2. At this point:

- Server 1 is the Sync Primary.
- Servers 0 and 4 are Sync Backups.
- Server 2 is a Sync Alternate.
- Servers 5 and 6 are Sync Locals.
- Server 3 is off-line.

Server 3 comes back online. It becomes a Sync Alternate. Even though it was originally a Sync Primary, the synchronizer now sees it as just another Sync Server, so the server does not resume its Sync Primary role. At this point:

- Server 1 is the Sync Primary.
- Servers 0 and 4 are Sync Backups.
- Servers 2 and 3 are Sync Alternates.
- Servers 5 and 6 are Sync Locals.

Server 0 fails. Server 2 becomes a Sync Backup because it has the higher priority (lower integer value) among the Sync Alternates. At this point:

- Server 1 is the Sync Primary.
- Servers 2 and 4 are Sync Backups.
- Server 3 is a Sync Alternate.
- Servers 5 and 6 are Sync Local servers.
- Server 0 is off-line.

Server 0 comes back online and becomes a Sync Alternate. Server 1, the Sync Primary, fails. Among the Sync Backups, server 2 has a higher priority than server 4, so server 2 becomes the new Sync Primary. Server 0 becomes a Sync Backup. At this point:

- Server 2 is the Sync Primary.
- Servers 0 and 4 are Sync Backups.
- Server 3 is a Sync Alternate.
- Servers 5 and 6 are Sync Locals.
- Server 1 is off-line.

Server 2 fails. Server 0 becomes the Sync Primary and server 3 becomes a Sync Backup. At this point:

- Server 0 is the Sync Primary.
- Servers 3 and 4 are Sync Backups.
- Servers 5 and 6 are Sync Locals.
- Servers 1 and 2 are off-line.

Server 3 fails. Even though only one Sync Backup remains, neither server 5 nor server 6 is considered because neither is a Sync Server. At this point:

- Server 0 is the Sync Primary.
- Server 4 is a Sync Backup.
- Servers 5 and 6 are Sync Locals.
- Servers 1 and 2 and 3 are off-line.

Modifying the Default Cluster for Fast Cluster Setup

The fastest and easiest way to set up a cluster is to set the cluster up when you install NAS.

After installation, the easiest way to set up a cluster is to modify the default cluster that was automatically created when you installed NAS. At installation, the `SyncServers` kregedit key for the default cluster lists only one server— the server itself. The default cluster is the name of *hostname*-NoDsync, where *hostname* is the name of your local machine. For instance, if you install NAS on a machine named "acarey," the default cluster is `acarey-NoDysnc`. The default cluster contains all that a cluster needs to be complete and active except for the new name for the cluster and the names of all Sync Servers with which to synchronize.

Because the default cluster already contains all the kregedit keys that a cluster needs, you can easily set up a cluster by making a few substitutions in the kregedit keys for the default cluster. If you were creating a completely new cluster, you would have to create the kregedit keys for that new cluster.

Entering IP Addresses Using kregedit

When you edit the `SyncServers` key for the default cluster, you will enter the IP address for each of the Sync Servers in your cluster.

At installation, the IP address for each NAS machine is placed in the `SyncServers` key of that server's default cluster. When you enter the address for each Sync Server into the first `SyncServer` registry key, remember that you can find the information in the registry for each NAS machine.

Note, however, that you will remove this entry on each Sync Local. If you decide later to promote a Sync Local to a Sync Server, you will have to find the address information elsewhere.

Editing Default Cluster Keys

Sync Locals are never listed in the `SyncServers` key for a legitimate cluster. But, because each Sync Local is automatically listed in its own default cluster, you must remove each Sync Local from its own `SyncServers` key.

This necessity will be obvious if the cluster settings you edit belong to a Sync Server.

To edit the default cluster keys, perform the following steps:

1. Stop the application server whose settings you will edit.

   Be aware that editing the server registry while the server is running can cause serious problems. Also, some changes take effect only after the engine is recycled.

2. Open kregedit by typing `kregedit` at the command prompt.

   The kregedit tool displays the keys and values that apply to the NAS
   machine as shown in the following illustration:
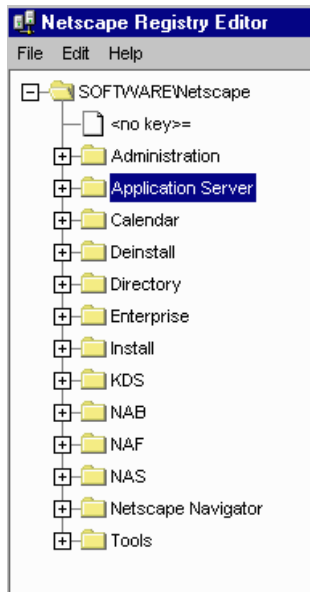
3. Open the following folder:

   ```
   SOFTWARE\Netscape\Application Server\Clusters\
   ```

   In this example, the default cluster is named `acarey-NoDsync` and, so
   far, contains one Sync Server with a priority of zero.

   ```
   SOFTWARE\Netscape\Application Server\Clusters\acarey-
   NoDsync\SyncServers
   ```
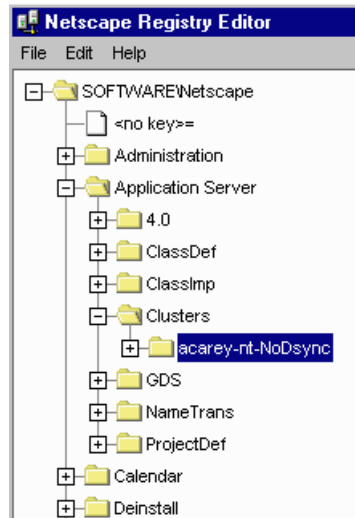
Whenever one of the following steps directs you to modify a key name or value, you can modify that name or value by performing the following steps:

1. Double-click the key name to bring up the Modify Value dialog box.

2. Enter the new name or value in the dialog box.

3. Click OK.

Note   Ignore the `SyncPersChunkSz` value. This key relates to an unsupported feature and is now ignored by the server.

4. Check and modify the `SyncTimerInterval` value, as necessary, which is found in the following location:

`SOFTWARE\Netscape\Application Server\Clusters\`*hostname*`-NoDsync`

This key specifies the intervals, in seconds, at which the synchronization service wakes up and checks to see whether any data has expired. Specifically, this key specifies how often the timer thread goes through the node list and removes all the nodes that have expired.

If this value is too large, expired data will still be accessible. If this value is too small, the frequent waking up and checking can degrade system performance. The default value of 60 seconds is good for most clusters.

5. Change the default name (in this case, `acarey-NoDsync`) to a new, unique name for your cluster.

6. Check and modify the `MaxBackups` value, as necessary.

   The maximum number of backup data synchronization servers determines how many Sync Backups are updated with data from the Sync Primary at the same time. For more information about backup data synchronization servers, see "What Is a Cluster?" on page 237.

   Because all Sync Backups are updated at the same time, an extra load is created for each additional backup server. Consider the performance impact when you set the number of backups, and try to choose a number that is high enough to provide safety, while not so high as to negatively affect performance. The default value of 1 is usually sufficient.

Note   Ignore the `MaxHops` key. This key relates to an unsupported feature and is now ignored by the server.

7. Add each Sync Server to the cluster under `SyncServers`.

   The IP addresses and port numbers under the `SyncServers` key are the Executive Server processes of the NAS machines that belong to this cluster. Each server is listed by its `host IP address:KXS port number=priority level`.

   1. Add the IP address and port number for the Sync Server.

   2. Set the priority for each Sync Server by double-clicking the priority value to bring up a pop-up window, entering the priority number, and clicking OK. The IP address, port number, and priority for the Sync Server should have been listed under the `SyncServers` key at installation.

   The priority setting for a data synchronization server determines which Sync Backup in a group of Sync Backups will become the replacement Sync Primary, and which Sync Alternate in a group of Sync Alternates will become the replacement Sync Backup.

Priority settings start at zero, the highest priority setting. The lowest priority is 65,535. For more information about priority, see "Determining Sync Server Priority" on page 242.

8. Close kregedit when you're finished.

9. Restart all application servers effected by these modifications.

All changes you make to the `SyncServers` key now apply to each server in the cluster.

After correctly completing these steps, you have redefined the default cluster. Now, follow the procedure in the next section to enable communication between the servers in the cluster.

Mapping the Synchronizer to the Cluster

For a cluster to communicate, the synchronizer in each server must know to which cluster the synchronizer belongs. This is done by mapping the `ClusterName` key of each synchronizer to the name of an actual cluster.

To map the synchronizer to a cluster, perform the following steps:

1. Stop the application server whose registry you will edit.

   Be aware that editing the server registry while NAS is running can cause serious problems. Also, some changes take effect only after the engine is recycled.

2. Open kregedit by typing `kregedit` at the command prompt.

   The kregedit tool displays the keys and values that apply to the NAS machine.
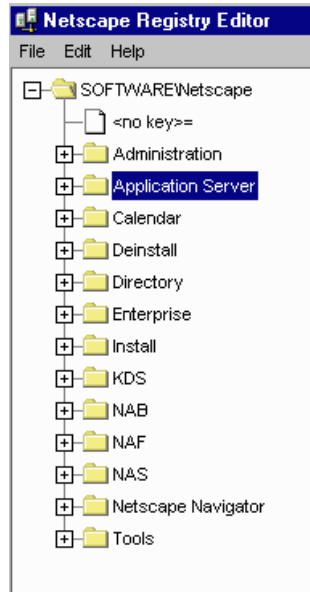
3. Open the following key:

```
SOFTWARE\Netscape\Application Server\4.0 \CCS0\Clustername
```

The following example shows the default cluster that has already been renamed "SampleCluster:"

```
SOFTWARE\Application Server\4.0\CCS0\ClusterName\hostname-NoDsync=0
```
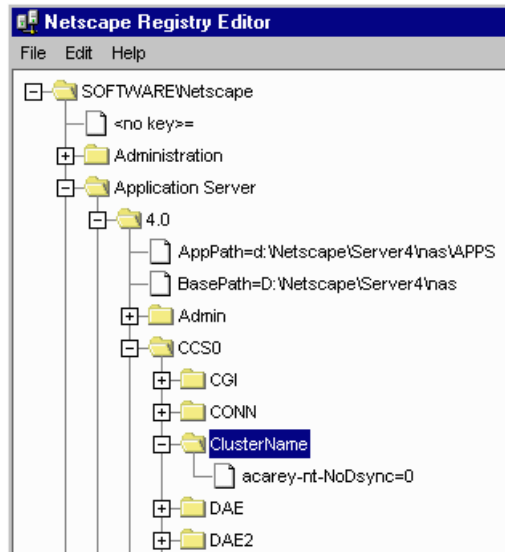
4. Rename the key under `ClusterName` to the name of the cluster to which the synchronizer should connect.

   If this key has not been previously modified, then the name under `ClusterName` will be *hostname*-`NoDsync`, where *hostname* is the name of your local machine.

5. Close kregedit when you are finished. The synchronizer should now be mapped to the cluster

Defining a Cluster

Create a cluster to organize NAS machines into data-synchronizing network-centric groups.

Even though each NAS machine can be mapped to only one cluster at a time, you can define as many clusters as you like. Some installations might define multiple clusters for testing purposes, for example.

While you can edit the default cluster to easily set up your first cluster definition, editing the default cluster defines only one cluster. To get more than one definition, create the additional clusters.

To create a cluster, perform the following steps:

1. Stop the application server whose registry you will edit.

   Be aware that editing the server registry while the server is running can cause serious problems. Also, some changes take effect only after the engine is recycled.

2. Open kregedit by typing `kregedit` at the command prompt.

   The kregedit tool displays the keys and values that apply to the NAS machine.
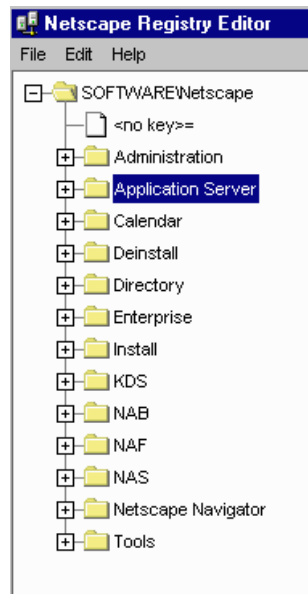
3. Open the following key:

   `SOFTWARE\Netscape\Application Server\Clusters`

   In this example, the default cluster is named `acarey-NoDsync` and, so far, contains one Sync Server with a priority of zero:

   `SOFTWARE\Netscape\Application Server\Clusters\acarey-NoDsync\`

Note Ignore the `SyncPersChunkSz` value. This key relates to an unsupported feature and is now ignored by the server.

4. Add the `SyncTimerInterval` value, as necessary, which can be found in the following location:

   `SOFTWARE\Netscape\Application Server\Clusters\`*ClusterName*

   This key specifies the intervals, in seconds, at which the synchronization service wakes up and checks to see whether any data has expired. Specifically, this key specifies how often the timer thread goes through the node list and removes all the nodes that have expired.

   If this value is too large, expired data will still be accessible. If this value is too small, the frequent waking up and checking can degrade system performance. The default value of 60 seconds is good for most clusters.

5. Add a new, unique name for your cluster.

6. Add the `MaxBackups` value, as necessary.

   The maximum number of backup data synchronization servers determines how many Sync Backups are updated with data from the Sync Primary at the same time. For more information about backup data synchronization servers, see "What Is a Cluster?" on page 237.

   Because all Sync Backups are updated at the same time, an extra load is created for each additional backup server. Consider the performance impact when you set the number of backups, and try to choose a number that is high enough to provide safety, while not so high as to negatively affect performance. The default value of 1 is usually sufficient.

Note   Ignore the `MaxHops` key. This key relates to an unsupported feature and is now ignored by the server.

7. Add the `SyncServers` key under the name of the cluster.

8. Add each Sync Server to the cluster under `SyncServers`.

   The IP addresses and port numbers under the `SyncServers` key are the Executive Server processes of the NAS machines that belong to this cluster. Each server is listed by its `host IP address:KXS port number=priority level`.

   1. Add the IP address and port number for the Sync Server.

   2. Set the priority for each Sync Server.

The priority setting for a data synchronization server determines which Sync Backup in a group of Sync Backups will become the replacement Sync Primary, and which Sync Alternate in a group of Sync Alternates will become the replacement Sync Backup.

Priority settings start at zero, the highest priority setting. The lowest priority is 65,535. For more information about priority, see "Determining Sync Server Priority" on page 242.

9. Close kregedit when you're finished.

After correctly completing these steps, you have defined a cluster. You can define as many clusters as you like, but you can map the synchronizer to only one cluster at a time. See "Mapping the Synchronizer to the Cluster" on page 250 for the procedure that enables communication.

Using the Administrator to Configure Clusters

You can perform the following tasks to configure clusters using Netscape Application Server (NAS) Administrator:

• Create a cluster.

• Add a server to a cluster.

• Remove a server from a cluster

• Change a server's Sync Server priority

• Modify the maximum number of Sync Backups

Note that to properly configure a cluster using NAS Administrator, you must register all the servers in the cluster. Otherwise, configuration changes will not apply across all the servers in the cluster.

For information about editing cluster settings directly, see the various sections earlier in this chapter that discuss how to configure clusters.

Creating a Cluster

To create a new cluster, perform the following steps:

1. From the NAS Administrator toolbar, click the General button to open the General window.

2. In the right pane of the General window, click the Cluster tab.

   The following window appears:
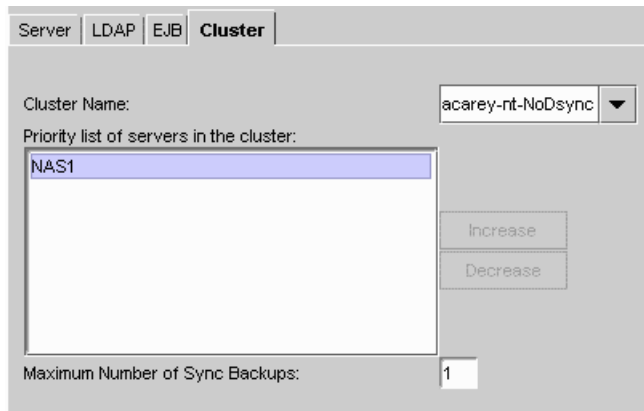
3. Double-click inside the Cluster Name drop-down box to select `hostname-No-Dsync.`

4. Use the Delete key on your keyboard to clear the Cluster Name drop-down box.

5. Type the name of your new cluster in the Cluster Name drop-down box and press the Enter key on your keyboard.

   You can choose any unique name for the new cluster.

6. Click Apply Changes.

   Your changes do not take effect until you restart the server.

After you restart the server, you can add NAS machines to the new cluster as described in the following section.

Adding a Server to a Cluster

To add an unassigned server to a cluster, or to reassign a server to a different cluster, perform the following steps:

1. From the NAS Administrator toolbar, click the General button to open the General window.

2. Click the Cluster tab.

   The following window appears:
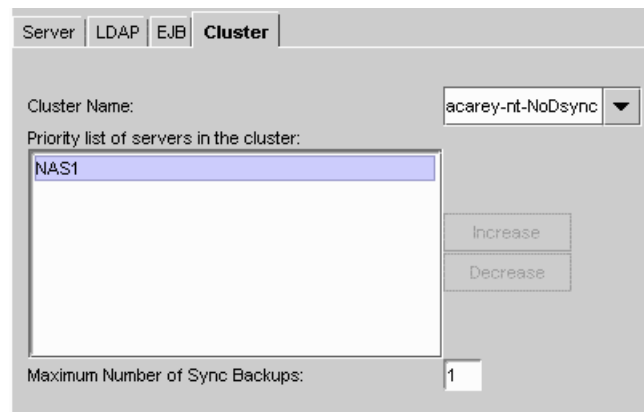
   A list of all registered servers is displayed in the left pane of the General window. Another list of servers, sorted by priority in a cluster, is displayed in the right pane as shown in the previous illustration.

3. In the left pane of the General window, click the name of the server you want to add to a cluster.

   A server that is not a member of a cluster, hence not participating in data synchronization, is listed under *hostname*-NoDsync, in the cluster list on the right.

4. From the Cluster Name drop-down box, select the name of the cluster you want to add the server to.

The Cluster Name drop-down list is populated with the cluster names that all *registered* servers belong to. If the servers in a cluster are not registered by NAS Administrator, then that cluster does not appear in the Cluster Name drop-down box. For the name of a cluster to appear in Cluster Name, you must register one or more of the servers in that cluster.

5. Click Apply Changes.

6. Shut down and restart every server in the cluster, including the server you just added. For changes to apply across the cluster, you must restart every machine to reload the memory on each machine with the cluster configuration changes. If at least one machine has a different cluster configuration loaded into memory than the other machines in the cluster, the new settings will not take effect and data synchronization will not work properly.

7. If when adding the server to a cluster, you removed it from another, you must also shut down and restart every server in the cluster from which it was removed.

Removing a Server from a Cluster

To remove a server from a cluster, perform the following steps:

1. From the NAS Administrator toolbar, click the General button to open the General window.

2. Click the Clusters tab to display the following window:
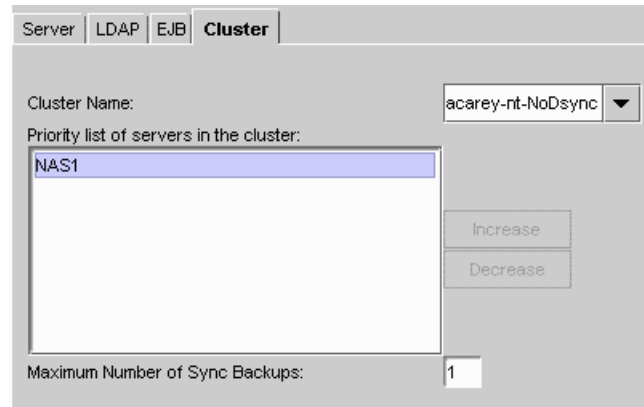
A list of registered servers is displayed in the left pane of the General window. Another list of servers, sorted by priority in a cluster, is displayed in the right pane.

3. In the left pane of the General window, click the name of the server you want to remove from the cluster.

You can remove a server from a cluster only when it is assigned to a cluster and registered with NAS Administrator. A server that is not a member of a cluster, hence not participating in data synchronization, is listed under *hostname*-NoDsync, in the cluster list. You cannot remove a server from the *hostname*-NoDsync list.

Note that you cannot remove an unregistered server from a cluster.

4. Click Remove from Cluster.

5. Click Apply Changes.

6. Shut down and restart every server in the cluster, including the server you just removed. For changes to apply across the cluster, you must restart every machine to reload the memory on each machine with the cluster configuration changes. If at least one machine has a different cluster configuration loaded into memory than the other machines in the cluster, the new settings will not take effect and data synchronization will not work properly.

Changing Sync Server Priority

To assign a new Sync Server priority to a server that is in a cluster, perform the following steps:

1. From the NAS Administrator toolbar, click the General button to open the General window.

2. Click the Clusters tab to display the following window:
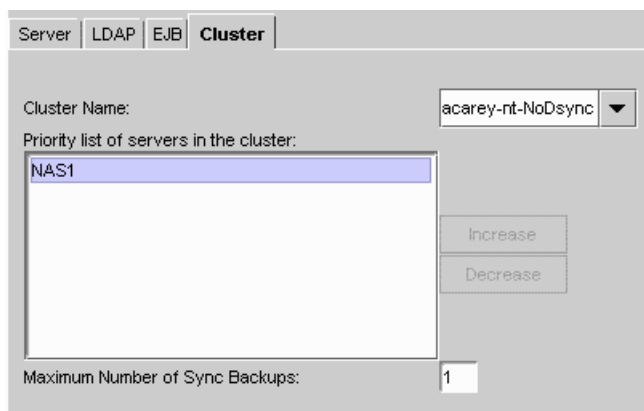
A list of registered servers is displayed in the left pane of the General window. Another list of servers, sorted by priority in a cluster, is displayed in the right pane.

3. In the left pane of the General window, click a server that is a member of the cluster whose Sync Server priority you want to change.

4. In the Priority List text box, click the name of the server whose Sync Server priority you want to change.

   You can change Sync Server priority order only for a registered server that belongs to a cluster. A server that is not a member of a cluster, hence not participating in data synchronization, is listed under *hostname-NoDsync*, in the cluster list on the right.

5. Click one of the following:

  - Increase to assign a higher priority.

  - Decrease to assign a lower priority.

  Click as many times as you want to increase or decrease the priority. For example, if a server has a Sync Server priority of third in line to take over for the Sync Primary, clicking Increase once changes the priority from third to second.

6. When you finish reassigning priorities, click Apply Changes.

7. Restart every server in the cluster, including the one whose priority you just changed. For changes in Sync Server priority to apply across a cluster, you must restart every machine so that they are all aware of their new priority sequence, relative to one another.
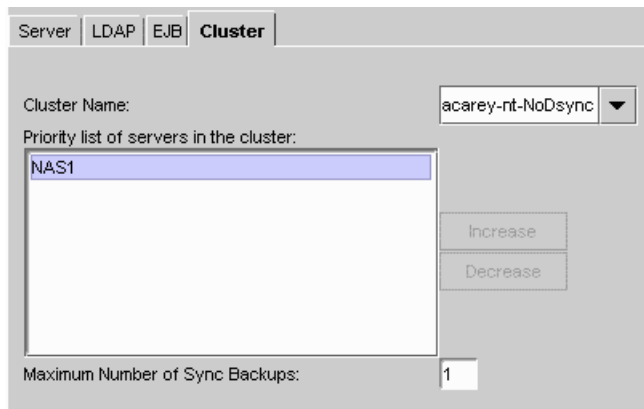
Modifying the Maximum Number of Sync Backups

To change the maximum number of Sync Backups allowed during a cluster session, perform the following steps:

1. From the NAS Administrator toolbar, click the General button to open the General window.

2. Click the Clusters tab to display the following window:

3. In the left pane of the General window, click the name of a server that is a member of the cluster you want to modify.

4. Enter the maximum number of Sync Backups allowed during a single cluster session in Maximum Number of Sync Backups.

Restart every server in the cluster. For changes to apply across a cluster, you must restart every machine so that they are all aware of the change in the maximum number of Sync Backups allowed.

Troubleshooting

This appendix contains the following information about troubleshooting Netscape Application Server:

- Configuring the Class Path

- Setting up Transactions

- Setting Environment Variables for Databases

Configuring the Class Path

When running applications, if the NAS Class Loader is unable to find the AppLogic class file through the SYSTEM_JAVA parameter (the registry parameter that contains both the CLASSPATH and GX_CLASSPATH settings) in the registry, NAS hands the request over to the Java Class Loader, which in turn reads the CLASSPATH environment variable to find the class file. This allows AppLogics and servlets to execute even if the user class path is not specified.

Setting up Transactions

When configuring your resource manager for use in global transactions, you might encounter one or more of the following problems:

- What if xa_open Fails?

- What if xa_recover Fails?

- What Is a "Lock Held by In-Doubt" Error?

- How Do I Configure the Number of Server-Side Connections?

What if xa_open Fails?

If an `xa_open` failure message appears in your log file, you may have a problem with the open string. Global connections rely on open strings, which provide information for global transaction initialization. When installing Netscape Application Server, the installation program puts default values in this open string. Check to be sure that the server name, user name and password are set correctly. Refer to "Setting Up Resource Managers for Distributed Transactions" on page 178 for the appropriate open string format for your database.

If you find an `XAER_RMERR` error, you have set the server instance incorrectly in the open string or the server is down.

If you find an `XAER_INVAL` error, there is a syntax error in your open string.

What if xa_recover Fails?

The following is an example of an `xa_recover` failure:

```
1 00271 99/04/30-10:00:28.124250 5c2c0837 W  xa_recover to RM 0 returned
x tCode -- 0xfffffffd (XAER_RMERR)

1 00271 99/04/30-10:00:28.124250 5c3c1017 W  Terminating recovery scan
for  0.
```

An `xa_recover` failure indicates that the database server is not set up for recovery. You must run the appropriate database setup script to create recovery tables and procedures.

For example, for Oracle databases, run the following scripts with `sys` permissions from the `sqlplus` prompt:

```
ftp://ftp1.netscape.com/private/nas/40beta2/extra/xa_sql/xaviews.sql
```

```
ftp://ftp1.netscape.com/private/nas/40beta2/extra/xa_sql/
xaviews_add.sql
```

What Is a "Lock Held by In-Doubt" Error?

Global transactions are left "hanging" or in-doubt when a Java Server (KJS) process is abruptly killed or crashes. When the KJS process restarts, these transactions are rolled back, but if you want to manually delete them, refer to "Resolving In-Doubt Transactions" on page 189.

How Do I Configure the Number of Server-Side Connections?

Once a global transaction is started on a thread, a connection is tied to that thread. Therefore, when configuring the number of server-side connections, use the total number of Java Server (KJS) threads in your enterprise.

For example, for Oracle databases, change the value of `max_number_processes` in the `init`*instancename*`.ora` file in the `pfile` directory of the Oracle server installation.

Setting Environment Variables for Databases

See "Post-Installation Notes" in the *Installation Guide*.

Setting Environment Variables for Databases

Index

L

LDAP
described 104
ldapmodify
described 124
modifying entries with 124
LDIF 123
described 123
entries, adding to Directory Server 123
entries, described 123
entries, formatting 123
`ListenPort` 152
load balancing 196, 199, 200, 206, 209, 222
adjusting weight factors 223, 224, 227
broadcast interval 230
calculating loads 223
communication with web connector 218
described 215, 216
disabling 211
distributing applications for 204
effect on user requests 197
hosting applications for 209
and multicast communication 200
plug-in 195, 196, 218
prerequisites 218
sticky 219
sticky, enabling 220
update interval 230
load-balancing service 216, 217, 218, 223, 226, 230
log( ) 91
Log_db2.sql 95
Log_ifmx.sql 95
Log_mssql.sql 95
Log_ora.sql 95
Log_syb.sql 95
log buffer 95
log failure
recovering from 190
logging
described 89

process data to a file 74
server messages 89
specifying NAS machine for 198
to file 74
to process console 94
web server requests 144
logging service
configuring 93
enabling 94
logical volumes 166, 172, 191
`logtime` field 100

M

Management Information Base (MIB)
described 86
formatting entries 86
mapping HTTP variables 148
master agent 82
configuring 82
starting 84
starting on a nonstandard port 85
MaxBackups
described 238, 241
modifying value 249, 254
MaxHops 249, 254
`MCastHost` 234
`MCastPort` 234
memory thrash 70, 223, 231
message-logging service 89
configuring 93
enabling 94
messages
choosing which type to log 90
console 93
error 90
event logging 89
formatting 92
information 90
types 90
warning 90
method override 63