TME® 10 Software Distribution
for AIX®

Quick Beginnings

Version 3.1.5

IBM

TME® 10 Software Distribution
for AIX®

Quick Beginnings

Version 3.1.5

IBM

> **Note**
>
> Before using this information and the product it supports, be sure to read the general information under "Notices" on page xv.

> **ISO 9001 Certification**
>
> ISO 9001 Certification This product was developed using an ISO 9001 certified quality system.
>
> Certification has been awarded by Bureau Veritas Quality International (BVQI) (Certification No. BVQI - 92053).
>
> BVQI is a world leader in quality certification and is currently recognized by more than 20 accreditation bodies.

Contents

Figures

Tables

Notices

References in this publication to Tivoli Systems or IBM products, programs, or services do not imply that they will be available in all countries in which Tivoli Systems or IBM operates. Any reference to these products, programs, or services is not intended to imply that only Tivoli Systems or IBM products, programs, or services can be used. Subject to valid intellectual property or other legally protectable right of Tivoli Systems or IBM, any functionally equivalent product, program, or service can be used instead of the referenced product, program, or service. The evaluation and verification of operation in conjunction with other products, except those expressly designated by Tivoli Systems or IBM, are the responsibility of the user.

Tivoli Systems or IBM may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing,
IBM Corporation,
North Castle Drive,
Armonk,
New York
10504-1785,
U.S.A.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation
P.O. Box 12195
3039 Cornwallis
Research Triangle Park, NC 27709-2195
USA

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

Trademarks and Service Marks

The following terms are trademarks or registered trademarks of International Business Machines Corporation or Tivoli Systems Inc. in the United States, other countries, or both:

IBM	NetFinity
The IBM logo	NetView
Tivoli	OS/2
The Tivoli logo	OS/400
AIX	Portmaster
NetView	PowerPC
AIXwindows	POWERserver
Application System/400	RACF
APPN	RISC System/6000
AS/400	RS/6000
AT	SAA
BookManager	SystemView
DB2	System/370
IBM	VTAM
MVS/ESA	WIN-OS/2
MVS/XA	

Microsoft, Windows, and the Windows NT Logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

PC Direct is a trademark of Ziff Communications Company in the United States, other countries, or both and is used by IBM Corporation under license.

UNIX is a registered trademark of the Open Group in the United States and other countries.

C-bus is a trademark of Corollary, Inc. in the United States, other countries, or both.

Other company, product, and service names may be trademarks or service marks of others.

About This Book

This book describes the general concepts, installation, configuration, and use of
TME 10 Software Distribution, Version 3.1.5 for AIX, hereafter also referred to as
TME 10 Software Distribution.

Who Should Use This Book

This book is intended for information systems professionals who require a basic
understanding of TME 10 Software Distribution in order to evaluate its suitability for
software and data distribution and change control in networks based on the AIX/6000
operating system.

How This Book Is Organized

This book has the following parts and chapters:

Part 1, "Introduction to TME 10 Software Distribution"
  This part includes chapters that provide conceptual information about the product.
  They are:

- Chapter 1, "Overview of TME 10 Software Distribution" describes the benefits
  you obtain by using change control functions to manage a network of
  workstations. It introduces different network topologies supported by
  TME 10 Software Distribution, outlines product users, and tells you how to start
  and stop the product.

- Chapter 2, "Setting Up a TME 10 Software Distribution Network" provides the
  information you need to configure the workstations in your network as
  TME 10 Software Distribution targets.

- Chapter 3, "Preparing Software for Change Control" describes what must be
  done to prepare software in the format required by TME 10 Software
  Distribution.

- Chapter 4, "Using Change Control Operations" describes change control and
  distribution functions, and introduces how to schedule operations in your
  network.

- Chapter 5, "Tracking Network Operations" describes the facilities provided by
  TME 10 Software Distribution to keep track of the hardware and software
  configurations on workstations in your network.

- Chapter 6, "Using Security Functions" describes the security facilities provided
  by TME 10 Software Distribution.

- Chapter 7, "Finding and Using TME 10 Software Distribution Information"
  indicates where you can turn to find additional information required to use
  TME 10 Software Distribution functions.

How This Book Is Organized

Part 2, "Installing and Configuring TME 10 Software Distribution"
This part provides you with the information you need to install and configure the product. It includes the following chapters:

- Chapter 8, "Installation Requirements" introduces TME 10 Software Distribution, Version 3.1.5 for AIX product options, and lists the software and hardware requirements that must be met before installing the product

- Chapter 9, "Installing TME 10 Software Distribution, Version 3.1.5 for AIX" describes how to install TME 10 Software Distribution, Version 3.1.5 for AIX on an AIX server

- Chapter 10, "Editing the Base Configuration File" describes what must be done to configure the base configuration file at the CC server.

- Chapter 11, "Defining Users and Targets" describes how to define users and targets to TME 10 Software Distribution, Version 3.1.5 for AIX.

- Chapter 12, "Configuring STS Remote Connection Files" describes how to configure remote connections in STS networks.

- Chapter 13, "Configuring SNA/DS Remote Connection Files" describes how to configure remote connections in SNA/DS networks.

- Chapter 14, "Configuring TCP/IP Communication" describes how to configure TCP/IP in a network.

- Chapter 15, "Configuring APPC Communication" describes how to configure APPC in a network.

- Chapter 16, "Configuring VTAM and NetView DM for MVS" describes how to configure NetView DM for MVS and VTAM to communicate with TME 10 Software Distribution, Version 3.1.5 for AIX

Part 3, "TME 10 Software Distribution Scenarios"
This part of the book includes the following chapter:

- Chapter 17, "Software Distribution Scenarios" presents typical user scenarios for preparing software for distribution, cataloging it, and installing it on client workstations.

Part 4, "Appendixes"
The manual has the following appendixes:

- Appendix A, "Implementing Inventory Discovery" provides a description of the inventory discovery procedure.

- Appendix B, "Replacing the Quiesce Check" describes how the quiesce check can be replaced at a target. It is used to see whether users are logged on at a target before operations are begun.

- Appendix C, "Writing Change Control Scripts" describes change management scripts and how to create them.

- Appendix D, "Writing User Exits" on page 227 describes how to edit the C language source code provided with TME 10 Software Distribution to write user exits to create additional functions on a distribution server.

- Appendix E, "Using TME 10 Software Distribution from NetView/6000" describes how you can integrate this product with NetView/6000.

- Appendix F, "Customizing Message Alert Logging" describes how specify whether TME 10 Software Distribution is to send alerts to NetView/6000 and to NetView, and the level of the alert each message is to have.

- Appendix G, "Tivoli Enterprise Console and the Event Adapter" describes how specify whether TME 10 Software Distribution is to send alerts to T/EC Event Server using TME 10 Software Distribution Event Adapter.

- Appendix H, "Configuring the TME 10 Software Distribution Event Adapter" describes how to configure the TME 10 Software Distribution Event Adapter so that TME 10 Software Distribution sends alerts to the Tivoli Event Server.

- Appendix I, "Customizing the Tivoli Event Server" describes how to customizing the Tivoli Event Server to receive, recognize, and classify an event coming from TME 10 Software Distribution.

- Appendix J, "The TME 10 Software Distribution Event Adapter Alert Fields" describes how to customize the TME 10 Software Distribution Event Adapter alert fields for each event.

- Appendix K, "Setting Environment Variables" describes how to set the environment variables.

TME 10 Software Distribution Publications

For conceptual information and installation instructions for the TME 10 Software Distribution family of products, consult the appropriate publications:

- *TME 10 Software Distribution for AIX Quick Beginnings*, SH19-4333

- *TME 10 Software Distribution for OS/2 Quick Beginnings*, SH19-4334

- *TME 10 Software Distribution for Windows NT and Windows 2000 Quick Beginnings*, SH19-4335

- *TME 10 Software Distribution for NetWare Quick Beginnings*, SH19-4341

- *TME 10 Software Distribution for NetWare Command Reference*, SH19-4342

- *TME 10 Software Distribution Clients Installation and Configuration*, SH19-4337

Notation Used in This Book

This book uses the following highlighting conventions in text:

Bold	Bold print indicates choices made from a menu or action bar. It is also used to highlight fields and push buttons on panels.
Italics	Italic print is used for introducing new terms in the text or for emphasis.
`Monospacing`	Monospacing indicates system messages, special characters, statuses, directory names, user input, and examples.

Notation Used in This Book

UPPERCASE	Uppercase letters are used for commands, devices, and file names.
<angle brackets>	Angle brackets are used to enclose the names of variables where you must substitute an appropriate value.

Where a command and its associated parameters are too long to be shown on one line, the symbol "➤" at the end of a line means that the next line is a continuation of the command string. When you enter the command, enter it all on one line.

What's New in This Release

Release 3.1.5 of TME 10 Software Distribution contains the following new or changed functionalities:

Support for New Platforms

TME 10 Software Distribution, Version 3.1.5 adds support for the following platforms:

- Windows 2000 (Professional and Server)
- Windows NT 4.0 (Service Pack 5 and 6a)
- OS/2, version 4.5 (Warp server for e-business)
- AIX, version 4.3.*x*

New Pristine Scenarios

TME 10 Software Distribution, Version 3.1.5 Client can be installed on a pristine workstation in the following environments:

- Windows 2000 Professional
- Windows 2000 Server
- Windows NT 4.0 Server/Workstation
- OS/2 4.5 (Warp Server for e-business)
- AIX 4.3.3

This is in addition to the following pristine installation environments, which are maintained from the previous release:

- Windows 3.11
- Windows 95
- Windows NT Version 3.51
- OS/2 3.0.*x* (Warp)

Complete Platform Support Table

Table 1 on page xxii shows details of the platforms on which TME 10 Software Distribution is available. The columns in the table contain the following information:

Server Scratch Indicates whether the Server software can be installed from scratch. Scenarios describing how to carry out the scratch installations can be found in the relevant Quick Beginnings manuals.

Server Upgrade Indicates which version of the TME 10 Software Distribution Server can be upgraded, by supplying a reference that can be looked up in Table 2 on page xxiii. Scenarios describing how to carry out the upgrade can be found in the README file.

Client Scratch Indicates whether the Client software can be installed from scratch. Scenarios describing how to carry out the scratch installations can be found in the Client Installation and Customization manual.

Support for New Platforms

Client Pristine   Indicates whether the Client software can be installed on a pristine workstation (i.e. a workstation with no operating system installed). Scenarios describing how to carry out the pristine installations can be found in the Pristine and Migration Scenarios manual or the Installation Scenarios for AIX manual.

Client Upgrade   Indicates which version of which Client software can be upgraded, by supplying a reference that can be looked up in Table 2 on page xxiii. Scenarios describing how to carry out the upgrade can be found in the relevant README files.

Table 1. TME 10 Software Distribution, Version 3.1.5 Platform Support

Platform		Server		Client		
OS	**Version**	**Scratch**	**Upgrade**	**Scratch**	**Pristine**	**Upgrade**
Windows	2000 Professional	Y		Y	Y	
	2000 Server	Y		Y	Y	
	NT 4.0 (SP5 & 6a)	Y	1	Y	Y	5
	NT 3.51	Y	1	Y	Y	5
	98			Y		6
	95			Y	Y	6
	3.11			Y	Y	7
OS/2	3.0*x*	Y	2	Y	Y	8, 11
	4.0	Y	2	Y		8, 11
	4.5 (Warp server for e-business)	Y		Y	Y	
AIX	3.2.5 - 4.2.1	Y	3	Y		9
	4.3.3	Y	3	Y	Y	9
NetWare	4.11 - 4.2*x*	Y	4	Y		10

Table 2 on page xxiii shows the products (and versions) that can be upgraded to TME 10 Software Distribution, Version 3.1.5; the Reference column refers to Table 1.

Table 2. Products from which TME 10 Software Distribution, Version 3.1.5 can be upgraded		
Reference (see Table 1)	**Version installed**	**CSD or Fix Pack installed**
TME 10 Software Distribution		
1	3.1.3 Server for Windows NT	XR21923
2	3.1.3 Server for OS/2	XR21923
3	3.1.4 Server for AIX	99/10
4	3.1.3 Server for NetWare	XR21924
5	3.1.3 Client for Windows NT	XR21923
6	3.1.3 Client for Windows 9x	XR21923
7	3.1.3 Client for Windows 3.1	XR21923
8	3.1.3 Client for OS/2	XR21923
9	3.1.4 Client for AIX	99/10
10	3.1.3 Client for NetWare	XR21924
NetView DM/2		
11	2.1	

Deletion of Pending Requests from Host

In the circumstances where TME 10 Software Distribution is executing software distribution requests from a focal point running Tivoli NetView Distribution Manager (NetView DM for MVS) Release 7, the MVS focal point can now issue a request to delete any distribution requests that are waiting to be processed or are being processed at the TME 10 Software Distribution server.

- In the case of a distribution request waiting to be processed, the original request will be deleted, and a report sent to the focal point confirming the deletion.

- In the case of a distribution request that is in execution when the deletion request arrives, the original request will be completed, and a report sent to the MVW focal point confirming the successful completion of the original request; no report concerning the unfulfilled deletion request will be sent.

In the case of nodes in a distribution network that are not running TME 10 Software Distribution, Version 3.1.5 (i.e. older versions of TME 10 Software Distribution or NetView DM/2) the deletion requests from the MVS focal point will be ignored.

This functionality runs in the background with no intervention required by the operator of the TME 10 Software Distribution server.

Note: As a consequence of this new functionality global names starting with $DELETE.$PENDING are reserved, and may not be used.

The stattg command gives details of the status of the agent at the local target. A new parameter has been added to the command to reveal additional information.

In the previous releases, and when used without the new parameter, the command reports these statuses:

Available Agent running and ready to process a request

Not Available Agent not running or not accessible

Busy   Agent running a request and not available to process any other request.

There are circumstances in which it is possible for the server to have in its database more than one workstation name for the same agent.

For example, if a workstation has been re-defined to the server for some reason, the operator may have supplied a different workstation name than that originally used, but have used the original hostname. In this event, the agent now has the new workstation name, but the server has both workstation names defined; prior to this release the agent reported itself as being *Available* under both workstation names.

With this release, by using the parameter -c, in the event that the agent is *Available* and not *Busy*, the command now returns the status *Unknown* if the hostname of the agent is correct but the workstation name in the status request does not match the workstation name of the agent. Thus, by using the -c parameter, polling both workstation names will allow you to identify which is the correct one, as one will return the status *Available* and the other *Unknown*. If the parameter is not used, the original functionality is maintained.

However, before using this parameter you should consider the question of the timing of the stattg requests. When an agent receives a stattg request it sends the status to the server but is then not immediately available to satisfy another request. This means that a second request, received within, say, one minute of the first request, will return the status *Not Available*. If you are polling two suspect workstation names you should wait for this period before sending the second request.

This also means that if you send a stattg request using the asterisk wildcard to obtain the status of all or a group of workstations, the results received will depend on whether the *incorrect* workstation name comes before or after the *correct* one in the server's database:

Incorrect workstation name is polled first
> The status of the *incorrect* workstation name will be given as *Unknown*, while the *correct* workstation will give *Not Available*

Correct workstation name is polled first
> The status of the *correct* workstation name will be given as *Available* while the *incorrect* workstation will give *Not Available*

| Thus, after using the asterisk wildcard with the -c parameter, you should individually
| poll each workstation name given as *Not Available*, waiting for approximately one
| minute before issuing each command. Workstations that are genuinely unavailable will
| report the same status as before; workstations that were unavailable while they were
| recovering from a previous stattg command will now report their true status.

| The full details of the stattg command are given in *TME 10 Software Distribution
| Command Reference*, *TME 10 Software Distribution for NetWare Command Reference*
| and *TME 10 Software Distribution for AIX Reference*.

Part 1. Introduction to TME 10 Software Distribution

This part provides you with basic information about TME 10 Software Distribution. Its purpose is to help you understand how to use the product effectively to perform data distribution and change control tasks in your network. It includes:

Chapter 1. Overview of TME 10 Software Distribution

Distributed computer systems are essential to the information technology strategies of many companies. The added benefits resulting from automated work processes, however, bring to the forefront additional concerns—the electronic distribution, installation, and maintenance of the software used to perform these processes. How can base software, applications, related configuration data, and user data files on hundreds, even thousands, of workstations be kept up-to-date, consistent, and maintained in a cost-effective and efficient way?

This manual describes how TME 10 Software Distribution, a client/server systems management product, can help solve this complex problem in your enterprise. With TME 10 Software Distribution you can:

- ***Electronically distribute and install software from a central site***

  When a system software component or an application that runs on numerous workstations located at different sites, or even in different cities, has to be updated or upgraded, TME 10 Software Distribution can completely automate the procedure. You can prepare and package software at one workstation, send it to the affected target workstations, and install it automatically.

  You can facilitate the administration of software installation and maintenance on large numbers of heterogeneous workstations by taking advantage of TME 10 Software Distribution's dynamic functions. By organizing your target workstations into dynamic groups whose members change according to the criteria you establish, you can selectively administer software. You can also organize the content of software packages dynamically, so as to install only certain files on certain workstations.

- ***Keep track of hardware and software installed across the network on server and client workstations***

  TME 10 Software Distribution automatically stores a history and inventory record of all the hardware and software installed on each workstation in a network. This means that you are always aware of the configuration of all the workstations in your network, and consequently of the activities that need to be performed to ensure consistency where you require it.

- ***Distribute and collect data***

  System data files and user data files (such as user flat files or database exports) can be exchanged between the central site and target workstations and across workstations in the network. You can distribute data from the central site to targets by grouping workstations that need to receive the same files. Data distribution operations also provide data compression and page code translation options.

- ***Plan and schedule distribution and installation operations***

  If you must schedule specific operations repeatedly on workstations, these operations can be automated using TME 10 Software Distribution. As an administrator at a central site, you can use *transmission plans* to organize and

schedule electronic software and data distribution, installation, and maintenance activities.

A *plan* is a group of operations ordered in the sequence they are to be executed. It can include the specification of conditions which must be met to enable execution on all target workstations. For example, a request to back out of any unsuccessful installations can be included in a plan to install software upgrades on targets. You can specify the day and the time that you want the group of functions to start running. You can also set time intervals for recursive executions of a plan.

- **Manage a multiplatform environment**

  You can take advantage of the benefits of TME 10 Software Distribution in a multiplatform environment. Workstations running AIX, OS/2, Windows, UNIX, and many other operating systems on different hardware platforms, as listed later in this chapter, can be controlled from a TME 10 Software Distribution server. TME 10 Software Distribution servers can also interoperate with servers running different operating systems (such as OS/2 and Windows) in the same network. A NetView DM for MVS Release 7 site can also act as a central site for software and data distribution.

TME 10 Software Distribution can, therefore, become a key element in ensuring the productivity and efficiency of your workstations and users by providing you with the means to:

- **Save resources, time, and money**

  When you install software manually on individual workstations, the process is time-consuming both for those who perform the installation and for those who work at the workstation. When you use TME 10 Software Distribution to automate software management, *one* person (the administrator) from *one* workstation can update thousands of computers, and plan the installation for a convenient time when it does not interfere with anyone's work schedule. What's more, you can automatically create backups of old software, which can immediately be reinstalled should an installation be unsuccessful.

- **Improve efficiency**

  Controlling *all* the software installed on *all* the workstations means ensuring constant compatibility and consistency across your network. No longer will different versions of software be run on different workstations. Unpleasant surprises of incompatibility between workstations will no longer be a common occurrence.

In this book the functions provided by the product are referred to as software distribution and change control.[1] The same term is used across the TME 10 Software Distribution family of products.

---

[1] In TME 10 Software Distribution terminology, *change control* is at times referred to as *change management*.

Using TME 10 Software Distribution in a Network

You can use TME 10 Software Distribution functions in many different network topologies, which differ greatly in complexity. In the simplest networks, TME 10 Software Distribution is installed on a workstation that is referred to as the network *change control server* (server). The other workstations in the network become *change control clients* (clients) that can work in conjunction with a TME 10 Software Distribution server.

Figure 1 shows a simple network with a server and its clients. Clients controlled by a server are referred to as *local targets*. The set of local clients, together with the server that controls them, is known as a change control *domain*.
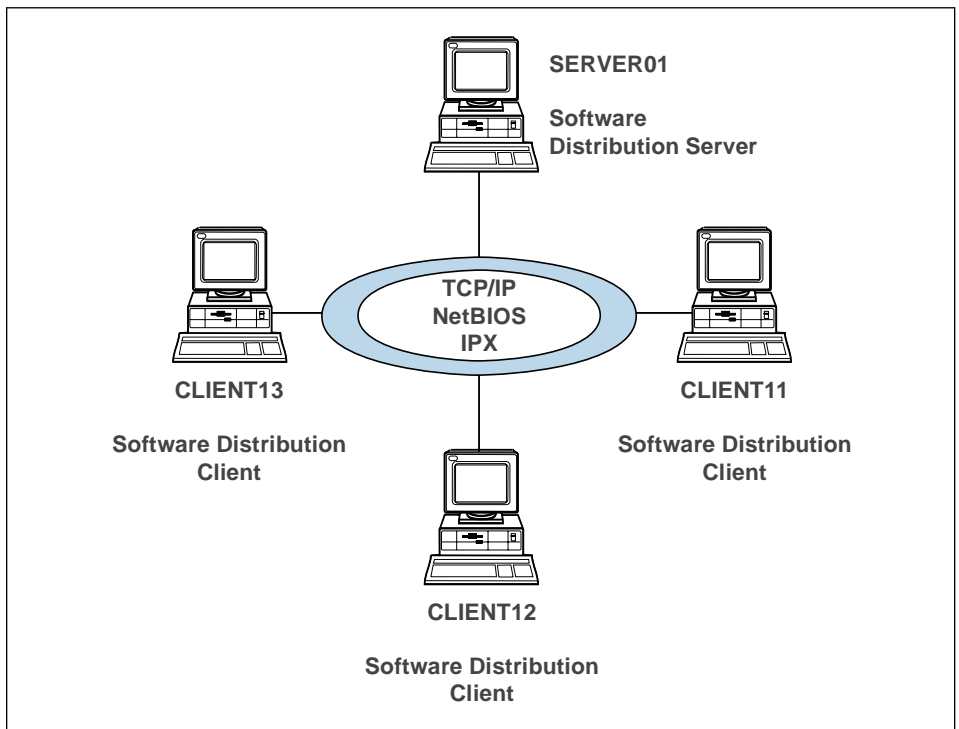
Figure 1. A Single-server TME 10 Software Distribution network

Change control and distribution activity in a network is usually initiated from a server. However, if a client is configured with the necessary authorizations, it can be an *active* client, meaning that it can initiate operations on other clients in the network, even those outside its own domain. A *passive* client can only have operations performed on it.

More complex networks can combine interconnected domains. Servers and clients in other domains are referred to as *remote targets*. Figure 2 on page 6 shows a network with two domains.

Using TME 10 Software Distribution in a Network
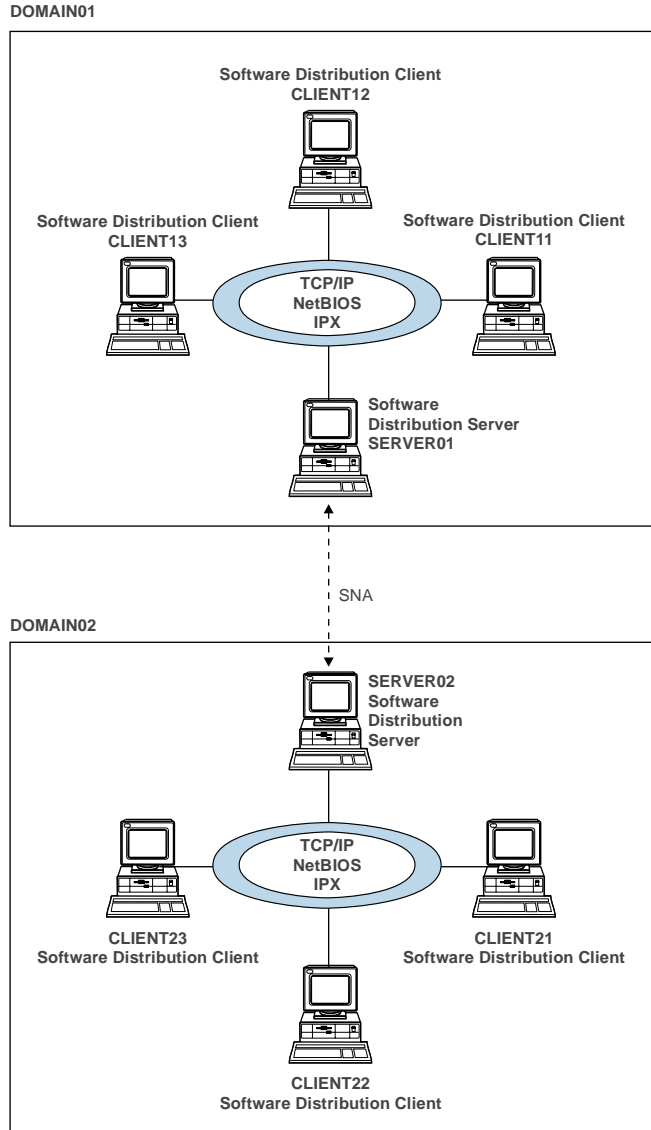
Figure 2. Interconnected domains

Servers can be connected in a hierarchical relationship, allowing the management of the thousands of clients connected to them.

Servers can be controlled from one or more central sites that maintain a total picture of the entire change control network. Figure 3 on page 7 is a simple configuration of multiple domains connected to a central site.
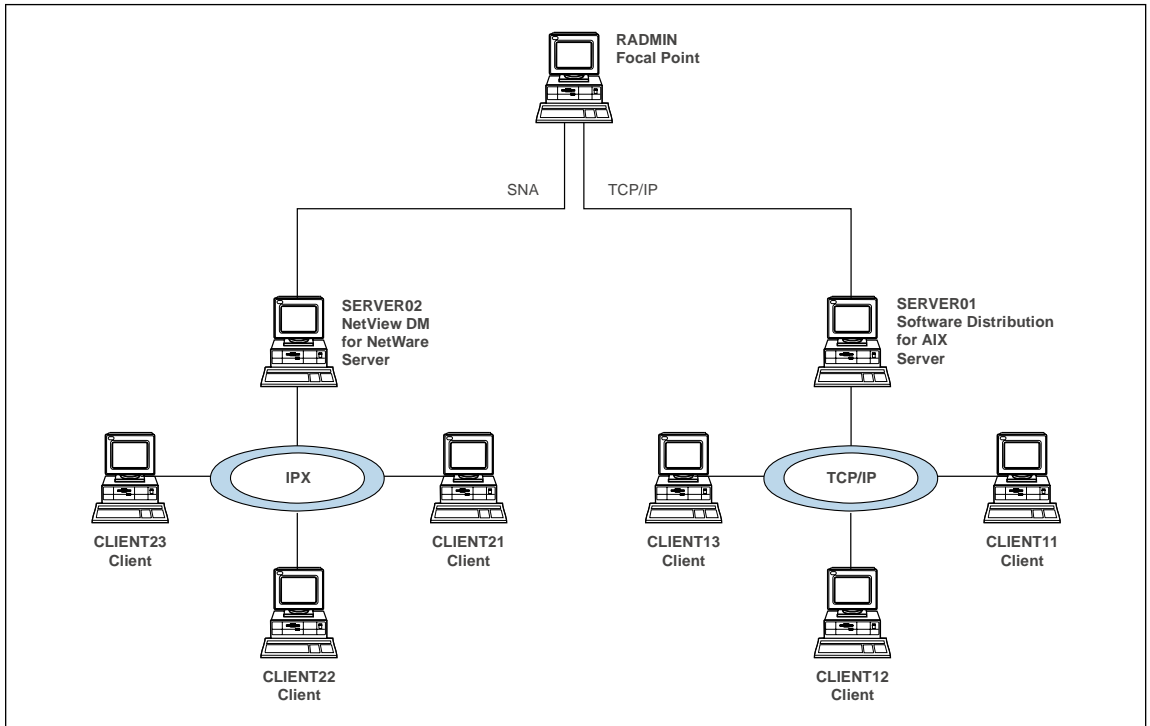
Figure 3. Centrally managed multidomain network

Networks can also include remote connections to systems running any of the products in the TME 10 Software Distribution/NetView DM for MVS family.

TME 10 Software Distribution Users

Three default classes of users are defined in a TME 10 Software Distribution system:

- Administrator
- Builder
- User

Change Control Administrator

A TME 10 Software Distribution *administrator* is responsible for ensuring that software packages are installed across the network, kept up-to-date, and removed when no longer required. Administrators use TME 10 Software Distribution typically to:

- Define the targets that a server can communicate with, both in its own domain and in remote domains

- Define the users of the product and their authorizations

- Define target, user, and software access authorizations

- Initiate the distribution of data files

TME 10 Software Distribution User Interfaces

- Initiate the managed installation of software packages by the server on client workstations
- Change the software installed on workstations across the network
- View the status of the software packages already installed on client workstations
- Track the status of change control and distribution operations across a network
- Run applications on remote, unattended workstations across the network

In addition to using a server workstation, the administrator can carry out software installation and data distribution functions from any client workstation that is configured in *user-interface-only* (UI only) mode. A UI only client provides an interface to access the server and the capability to initiate change control actions across targets in the network, but no change control operations can be directed to it.

Builder

The *builder* is typically a programmer who is authorized by the change control administrator to prepare the software to be installed using TME 10 Software Distribution. Each separate application or system software product must be prepared as a package suitable for installation by the client program it is destined for.

The builder uses either a server or a client workstation as a *preparation site* for software. The builder can also use a *single node* as a preparation site.

User

The *user* is anyone who uses the services of TME 10 Software Distribution on a client workstation to:

- Perform change control and distribution functions on their own or on other workstations in a network
- Distribute data files across the network to any workstation with a client product installed on it.

User Profiles

User profiles that correspond to these descriptions are installed automatically with the product. However, administrators of the system can create customized profiles that authorize product users to perform any combination of tasks.

TME 10 Software Distribution User Interfaces

You can choose either one of two user interfaces to work with TME 10 Software Distribution:

- Graphical user interface
- Command line interface

Graphical User Interface

The graphical interface provides access to all TME 10 Software Distribution functions. It can be started even if the product is not active, in which case you can perform only a limited number of operations (for example, you can configure new targets but you cannot initiate change control operations).

The graphical interface presents lists of *objects* (such as a list of files) that you select using the mouse or cursor keys. After you select the items you want to work with, select an *action* to perform on them.

The interface is divided logically into main windows from which you access functions. These windows are:

Catalog

This window contains a list of all files currently in the server *catalog*, a database that contains records of all files maintained by the TME 10 Software Distribution system. From this window, you can choose options that enable you to:

- Prepare change files
- Prepare transmission plans
- Perform distributions of change files and data files
- Initiate change control activities
- View change control history records
- Start and stop the product

Targets

Select this window to view a list of the targets that have been set up. From this window you can:

- Create new targets and group targets together
- Modify details about existing targets
- Delete existing targets
- View change control history of targets
- Alter the server configuration

Message Log

This window displays a list of all messages that have been logged. Access it to:

- View messages
- Display help information for any message
- Save the messages to a file

Help

This window can be used as an alternative way to access help information, which can be displayed from any dialog window.

Local Queues

Select this window to display a list of the queues that route requests to clients in a domain. From this window, you can:

- View the contents of queues
- Perform operations on queues

Remote Queues
> This window displays a list of the queues that route files to remote domains in a network. From this window, you can:

- View the contents of queues
- Perform operations on queues

> This window can be accessed only if the communication option is installed on your system.

Requests
> Select this window to track and control the progress of requests that have been submitted to TME 10 Software Distribution. From this window, you can:

- List the status of requests at the request, domain, and target levels.
- Hold, release, reschedule, restart, and delete requests. You can also erase requests from the product database.

From each of these windows, you can access any of the other main windows.

Command Line Interface

The command line interface is composed of commands that you issue from a command prompt. A command is a string of letters with corresponding parameters that you use to make specific requests. Expert users often prefer using the command line as an alternative to the graphical interface because it saves time.

You can perform all TME 10 Software Distribution operations using this interface. For example, to install the change file called `test.file.ref.1.1.2` on a group of targets called `groupone` at 10:00 on March 14, 2000, you would use this line command:

```
nvdm inst test.file.ref.1.1.2 -w groupone -d "14/3/2000" -t "10:00"
```

Line commands and their syntax are documented in the online Command Reference file.

Starting and Stopping the Command Line Interface

You do not have to enter any particular commands to start the command line interface. Simply enter commands from a prompt. Each specific command must be preceded by `nvdm`. For example, to display a list of targets defined in your network, at the NVDM > prompt, enter:

```
lstg *
```

Graphical User Interface

For OS/2 and Windows clients, a graphical user interface is available. See the *Quick Beginnings* manual for TME 10 Software Distribution for OS/2 or Windows NT for an overview of the graphical user interface.

Starting TME 10 Software Distribution

You can start TME 10 Software Distribution during the normal startup processing of your system. You can also start and stop the product's components from the graphical interface or the command line. To do so, you must be logged on to the operating system as the user root. click on the TME 10 Software Distribution icon.

To start TME 10 Software Distribution from the graphical user interface:

1. In the Catalog window, select **Engine** from the menu bar and select **Start the system** from the pull-down menu.

   A window with the message `The system has been started successfully` is displayed.

2. Select **OK** to return to the Catalog window.

To start TME 10 Software Distribution from the command line interface, enter the following command from a command prompt:

`nvdm start`

Starting the Graphical Interface
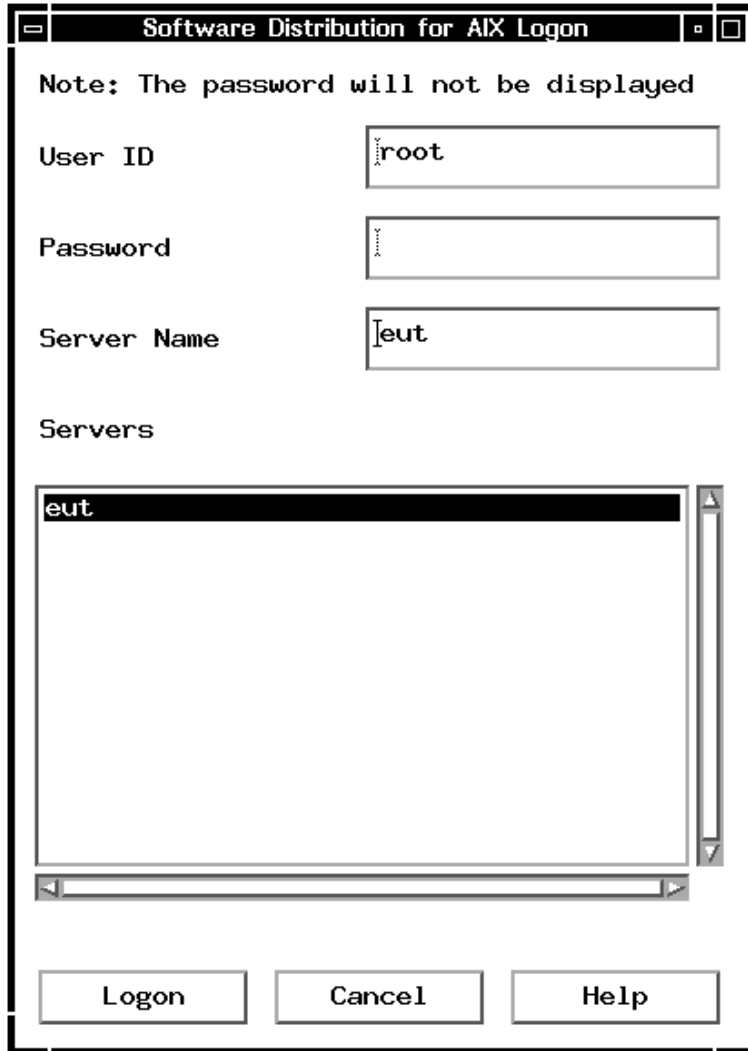
To start the graphical interface from the command line, complete the following steps:

1. Start X-Windows on your workstation by entering the following command at a command prompt: `xinit`

2. Enter the command: `nvdmgi`

   The TME 10 Software Distribution Logo window appears.

3. Select the **OK** push button to continue (If you do not select it, the Logo window will disappear after 10 seconds.). The TME 10 Software Distribution Logon window appears, as shown in Figure 4 on page 12.

Starting TME 10 Software Distribution


```
┌──────────────────────────────────────────────────────┐
│ ▭  Software Distribution for AIX Logon      ▫ □ │
├──────────────────────────────────────────────────────┤
│ Note: The password will not be displayed             │
│                                                      │
│ User ID                  │root              │        │
│                                                      │
│ Password                 │                  │        │
│                                                      │
│ Server Name              │eut               │        │
│                                                      │
│ Servers                                              │
│                                                      │
│ │eut                              │▲│                │
│ │                                 │ │                │
│ │                                 │ │                │
│ │                                 │ │                │
│ │                                 │▼│                │
│ │◄│                          │►│                     │
│                                                      │
│  │  Logon  │   │  Cancel  │   │  Help  │            │
└──────────────────────────────────────────────────────┘
```

Figure 4. Logging on to TME 10 Software Distribution

4. Enter your **User ID**, followed by your **Password** if you have one.

5. In the **Server Name** field enter the name of the server you want to be connected to for this work session. If a list of servers is displayed, select a server from the list.

6. Select the **Logon** push button. The Catalog window is displayed.

Note that starting the graphical interface does not automatically start TME 10 Software Distribution. If the product is not active, follow the instructions given in "Starting TME 10 Software Distribution" on page 11.

Stopping TME 10 Software Distribution

To stop the product from the graphical interface:

1. In the Catalog window, select **Engine** from the menu bar and select **Stop the system** from the pull-down menu.

2. Select **OK** to return to the Catalog window.

The graphical interface stops when all the main windows have been closed. To close a window, do one of the following:

- Press Alt+F4
- Select **Close** from the System menu
- Double-click on the System Menu icon
- Select **Close all** from the Windows menu to close all the windows and stop the graphical interface.

To stop TME 10 Software Distribution from the command line interface, enter the following command from a command prompt:

```
nvdm stop
```

Chapter 2. Setting Up a TME 10 Software Distribution Network

This chapter provides information to help you set up workstations in your network as TME 10 Software Distribution targets, and to configure their communication links. It also describes the functions you can use to keep track of the hardware configuration of the workstations in your network.

Types of Target in a TME 10 Software Distribution Network

Targets are defined in a TME 10 Software Distribution network according to the role they are to perform and how they are to perform that role (specified as target *mode*). This section describes the various roles you can assign to targets.

Targets are either local or remote, depending on their location in the network. A target that is in the same domain as the target communicating with it is a *local* target. When targets communicate across domains in multidomain networks, they are *remote* targets.

With the graphical interface ...

Define targets in the Targets window. Select:

- ► Target
- ► New target

To modify an existing target select it in the Target window then select:

- ► Selected
- ► Open
- ► Details

With the command line interface ...

Define targets using the **addtg** command. Modify targets using the **updtg** command, view information for existing targets using the **lstg** command, and delete targets using the **deltg** command.

Types of Local Target that Can Be Defined

You can define local targets to perform one of the roles described in the following sections.

Server

Servers contain the catalog where change files and data files, as well as change control history records, are stored. They also contain the configuration information for all the targets they can communicate with, be they local or remote. Whenever a client communicates with another target, the communication is routed through its server.

Types of Local Target that Can Be Defined

A server can initiate change control and distribution operations on targets in their own domain. If the remote administrator option is installed, they can also perform change control and distribution on remote targets.

Role	Target Type	Target Mode
Initiate change control on targets	Server	Push
Initiate distribution on targets	Server	Push

Client

Client targets work in conjunction with a server, and must have a TME 10 Software Distribution Client product installed. A local client can:

- Be the object of change control operations initiated by a server, when configured in *push* mode
- Distribute data files (send, retrieve, and delete them) to other clients in its domain
- Initiate change control operations on itself, using any software available to it from the server catalog, when configured in *pull* mode
- Build change files and store them in the server catalog
- View information about the software installed on its workstation, as well as information about the other clients and software available in the network
- Catalog and uncatalog data files

The following modes can be defined for a client target:

Role	Target Type	Target Mode
Server/other client initiates change control	Client	Push
Initiates change control on self	Client	Pull
Prepare change files	Client	Push or pull
Send, retrieve, delete data files	Client	Push or pull

Mobile Client

Mobile clients are local targets on which it is possible to do change management operations without connection to a server. Mobile clients must be installed with the mobile feature.

You connect a mobile client using the GUI or the **connect** command on the command line. With this command you can also establish the time when the connection window is opened, the connection duration, and a recursive daily connection. You disconnect a mobile client using the GUI or the **disconnect** command on the command line. With this command you can also stop the established recursive daily connection from the client to the server. See the helps or the *Command Reference* for more details.

Because mobile clients are not always connected with a server whose catalog they can access, they have their own local catalog. Once a mobile client's catalog has been updated, the client can perform the following operations locally:

- Install software and software changes

- Remove software

- Accept changes to software

- Activate pending changes on your workstation

- Uninstall software packages and software changes, along with any updates or fixes applied

- Catalog and uncatalog files in the local catalog

- Import change files from external devices

- View change file information

- View and purge pending requests

Mobile clients are not forced to work locally; they can connect to a server and function as non-mobile clients, performing all the tasks described in "Client" on page 16.

Mobile clients can also function as *fully disconnected* clients, meaning that they never connect to a server. They update their local catalog with change files manually from external media (CD-ROMs, tapes, diskettes), and then perform change control on themselves. Change control status information stored in the server catalog for fully disconnected mobile clients can then be updated manually, using the **updcm** command. A fully disconnected client must set the FULLY DISCONNECTED keyword in its base configuration file to YES.

Define the following attributes for mobile clients:

Role	Target Type	Target Mode
Work as mobile client	Mobile	Push or pull
Work as fully disconnected mobile client	Client	Disconnected

User Interface Only Client

These targets can be used *only* to run the TME 10 Software Distribution user interfaces. This type of target is useful when you have an environment with more than one server. It allows an administrator to access all servers from the same target, either to perform administrative tasks or to schedule operations.

A workstation configured in this manner is used to initiate change control on *other* targets, or to request distributions to and from the server. User interface only targets cannot be the object of change control instructions from the same server they are defined as a user interface only target for.

Define the following attributes for this type of target:

Types of Remote Target that Can Be Defined

Role	Target Type	Target Mode
User interfaces at server	UI only	None

Types of Remote Target that Can Be Defined

Remote targets can perform a greater variety of roles, in networks with more complex topologies. The possibilities are described in the following sections.

Server

Servers contain the catalog where change files and data files, as well as change control history records, are stored. They also contain the configuration information for all the targets they can communicate with, be they local or remote. Whenever a client communicates with another target, the communication is routed through its server.

A server can initiate change control and distribution operations on targets in their own domain. If the remote administrator option is installed, they can also perform change control and distribution on remote targets.

Role	Target Type	Target Mode
Initiate change control on targets	Server	Push
Initiate distribution on targets	Server	Push

Manager

Servers with the remote administrator option installed can be configured as *managers*. A manager can perform change control operations on both local and remote targets, and consequently on other servers in the network and the clients in their domains.

Role	Target Type	Target Mode
Perform change control on remote targets	Server	Manager
Perform change control on local targets	Server	Manager

Focal Point

A server that has the remote administrator option installed can be defined as a focal point. It can perform the same change control and distribution functions as a manager target. In addition, it receives and stores all change control reports from servers that have defined it as their focal point. Focal points can thus hold the central catalog for a network, as they store change control history records for all its subordinate workstations. Each server in a network can have one or more (up to eight) focal points.

A focal point can also be configured on a single-node target, in which case it can act solely as a report repository for a network without performing change control and distribution operations.

Role	Target Type	Target Mode
Perform change control on remote targets	Server	Manager
Perform change control on local targets	Server	Manager
Receive and store downstream change control reports	Server	Focal point
Receive and store downstream change control reports	Single	Focal point

Intermediate Node

Remote servers can act as intermediate nodes, whose task is to fan out requests to downstream nodes. The fan-out operation creates multiple copies of the same object to be distributed to a number of, or a group of, targets.

Using intermediate nodes is often an efficient and cheaper way of distributing large quantities of data to numerous destination targets, because they eliminate the need for a direct connection between the target that originates an operation and the targets addressed by the operation.

Role	Target Type	Target Mode
Fan-out change control requests	Server	Push

Single Node

You can use single-node targets as preparation sites for software, or as focal points that receive and store change control reports. They can be the destination of change control operations from servers that have the remote administrator option installed.

Role	Target Type	Target Mode
Act as preparation site	Single	Push
Receive and store change control reports	Single	Focal point

Defining Client Targets Automatically

You do not have to configure all client targets in a network individually from a server. You can set up your network so that they are configured automatically, or autoregistered, the first time a client target connects to a server. For autoregistration to take place, the AUTOMATIC TARGET REGISTRATION keyword in a server's base configuration file must be set to YES, and the TARGET ADDRESS and TARGET MODE keywords must be specified in the client base configuration file. See Chapter 10, "Editing the Base Configuration File" on page 79 for a description of the server base configuration file, and the *Installation and Configuration* manual for the client base configuration files.

When a client is automatically configured, its address and mode are inserted in its server database. Any other parameters for the client target must be specified manually using the graphical interface or the command line interface.

If a remote server in the hierarchy receives the registration of a target with a workstation name that already exists in this server's database, the new target is registered as `<target address_domain address>`.

Automatic configuration is performed at all remote servers connected in a linear hierarchy to the first server that registers the target, as long as AUTOMATIC TARGET REGISTRATION is set to YES in each server's database. However, automatic target registration information cannot be managed at NetView DM/2 servers or NetView DM for MVS focal points.

Platforms Supported in a TME 10 Software Distribution Network

TME 10 Software Distribution networks can include workstations that are connected across local and remote links. As a rule, local links connect servers to their clients and remote links connect servers to other servers in different domains.

Client Platforms Supported

You can connect a server to clients that run any of the following operating systems, provided they have the corresponding TME 10 Software Distribution Client product installed:

Table 3. Client Platforms Supported by TME 10 Software Distribution	
Platform	**Product**
AIX/6000®	TME 10 Software Distribution for AIX Client
OS/2	TME 10 Software Distribution for OS/2 Client
Windows 2000	TME 10 Software Distribution for Windows 2000 Client
Windows NT	TME 10 Software Distribution for Windows NT Client
Windows 95 & 98	TME 10 Software Distribution for Windows 9x Client
Windows 3.11	TME 10 Software Distribution for Windows 3.11 Client
NetWare	TME 10 Software Distribution Client for NetWare

Remote Platforms Supported

You can connect a server to workstations in other domains. As described in "Types of Remote Target that Can Be Defined" on page 18, these workstations can be configured as:

- Servers configured as managers or focal points
- Intermediate nodes
- Single nodes

Remote workstations can run any of the following operating systems, provided they have the corresponding TME 10 Software Distribution product installed. Table 4 on page 21 lists supported platforms and counterpart products, and the roles they perform within the network.

Table 4. Remote Platforms Supported by TME 10 Software Distribution		
Platform	**Product**	**Role**
AIX/6000	TME 10 Software Distribution, Version 3.1.5 for AIX	Manager Focal point Intermediate node Single node
OS/2	TME 10 Software Distribution for OS/2	Manager Intermediate node Single node
Windows 2000	TME 10 Software Distribution for Windows 2000	Manager Intermediate node Single node
Windows NT	TME 10 Software Distribution for Windows NT	Manager Intermediate node Single node
NetWare	TME 10 Software Distribution for NetWare	Manager Intermediate node Single node
MVS	NetView DM for MVS Release 7	Manager Focal point

Communication Protocols that Can Be Used to Link Nodes

Clients, servers, and TME 10 Software Distribution/NetView DM for MVS family products can be linked in a network using different transmission protocols:

SNA/DS across APPC or TCP/IP
You can configure SNA/DS across APPC or TCP/IP for both local client/server connections and remote server/server connections.

STS across APPC or TCP/IP
STS is the acronym for the term *server-to-server*. This is an internal TME 10 Software Distribution transport mechanism that can be configured in networks connected:

- Over TCP/IP for remote server/server connections to an NetWare or Windows NT server,

- Over TCP/IP or AIX for remote server/server connections to an AIX or OS/2 server.

STS communication offers better performance for many transmission operations, as well as these additional functions:

- You can send, retrieve, and delete data files that are not stored in the catalog, called *uncataloged* files. Uncataloged data files can also be used as procedures to be executed at targets.

- You can authorize data files so that they can be executed at remote targets. You can authorize change files so that they can be installed at remote targets.

Grouping Targets to Facilitate Change Control Operations

When STS communication is not used, authorization commands can only be issued at the local level.

- When remote targets are linked across STS, you can run inventory discovery on them. Inventory discovery detects and records the hardware and software products installed at a workstation. Without STS links, inventory discovery can only be performed on local targets.

- You can translate the encoding techniques (for example from ASCII to EBCDIC) when you send files remotely across STS links. Without STS, data translation can only be performed on locally transmitted files.

- Server connectivity is greatly improved when this transport mechanism is used. See Table 5 on page 24 for more details.

Support is not provided for networks that use mixed communication links. That is, if workstation A is connected to workstation B across an SNA/DS link, and workstation B is connected to workstation C across an STS link, then workstation A cannot initiate operations on or receive reports from workstation C.

"Communications Protocols" on page 67 provides full details of the communications protocols that can be used from a server with TME 10 Software Distribution, Version 3.1.5 for AIX installed to any client or server with which TME 10 Software Distribution can communicate.

How Grouping Targets Can Facilitate Change Control Operations

Grouping targets into homogeneous units can be a useful aid in organizing and scheduling change control operations, especially when you are dealing with an extensive network. When you request change control and distribution operations for a group, the operation is performed on all the targets included in it.

Groups can include up to 1000 local and remote targets, whose mode can be either push, pull, manager, or focal. What's more, you can define *static groups*, whose members are always the same, or *dynamic groups*, whose members change according to criteria you define for populating the group.

A dynamic group can contain both static and dynamic members. The static members are always included in the group, while the dynamic members change according to the filters and rules you establish for populating the group. Filters allow you to define the members of a group according to:

- Target names
- Target types (client, mobile, server, single, UI only)
- Target modes (push, pull, focal point, manager)
- Target operating systems
- Target access keys

Rules are statements that specify the logical relationship between values you specify for any of these tokens:

Target installation parameters
> Tokens substituted during the installation process. Installation parameters usually correspond to directory and path names.

Hardware tokens
> Hardware requirements specified when targets are defined individually to the system.

Change management statuses
> The statuses that form part of the history records stored in the catalog for change files and targets.

The following examples give you an idea of how you can use static and dynamic groups to your advantage:

- If you must send DATA_A1 to all the targets in EuroTravel's branch offices every week, you can create a static group called BRANCHES, that holds all the targets the file must be sent to. Then every week you simply have to send DATA_A1 to BRANCHES.

- Suppose you have to install a new version of a word processor only on those targets in EuroTravel offices where the word processor is already installed. You can create a dynamic group called HAVE_WORDPROC; the rule used to populate the group would state "include those targets whose change management status is equal to install for the EURO.WORDPROC.REF.0 change file". Then install the word processor on HAVE_WORDPROC.

With the graphical interface ...

Define groups of targets in the Targets window. Select the targets to be included in the group in the window and select:

- ► Selected
- ► Group

To define a dynamic group, select:

- ► Selected
- ► Dynamic group

With the command line interface ...

Define static target groups using the **addgp** command. You cannot use the command line interface to define dynamic groups of targets.

View information for existing target groups (both static and dynamic) using the **lsgp** command, and delete target groups using the **delgp** command.

The Number of Targets a Network Can Include

When you are planning the topology of a network, you must know how many workstations TME 10 Software Distribution can support, as shown in Table 5.

Table 5. Connection and Performance Considerations	
Maximum number of clients per server	2000
Maximum number of concurrently active clients in a domain	100
Maximum number of servers in a network	1600 minus number of SNA/DS servers
Maximum number of concurrently active STS servers	customizable up to 512 (256 in/256 out)
Recommended number of concurrently active STS servers	100 (50 in/50 out)
Recommended number of concurrently active SNA/DS servers	100 (50 in/50 out)

Chapter 3. Preparing Software for Change Control

This chapter describes what you must do to the software package so that it can be used in change control operations.

Change Files

Changes to software on a workstation are distributed in packages that contain software files together with instructions about how they are to be installed. These packages are called *change files*.

Contents of a Change File

The main information included in change files is:

- Specifications of the files to be included in the package and their location.

- Specifications of the location of files stored in directories at a *remote site*. These remote directories are mounted from a target when an installation or other change control operation is performed. When you use this remote source method, you can make use of file servers in your network and create smaller change files that occupy minimum disk space at both the server and clients.

- Specifications of directories to be created or deleted at the target workstation.

- Response files (for packages with CID conformance only; see "CID Change Files" on page 28).

- Information about hardware and software prerequisites and corequisites, including disk storage space requirements. TME 10 Software Distribution checks for the presence of prerequisite and corequisite software and the amount of disk space available for a package before attempting to install it.

  Change files that include all the prerequisite files required for an installation can be generated automatically.

- Scripts or programs that tailor the installation and other operations to be executed on the target.

- A description of each product or file that is included in the change file. This information is convenient for tracing change file history.

- Compression options and packing instructions to apply if transmitting or storing the change file in compressed format. You can also specify if each file in the change file is to be compressed at build time and decompressed at install time.

Change Files

Types of Change File

You create different types of change file depending on whether you have to install new software, update existing software, or apply fixes to existing software at targets. These types of change file are referred to by the following *change names*:

Refresh (REF)   Change files that contain a complete new copy of the software or data item being changed to a new level, release, or version.

Update (UPD)   Change files that contain an update to a component and are installed on top of the original. Updates change the level of the component ID. To install an update, you must first install a refresh change file as not removable.

Fix (FIX)   Change files that contain a fix for an existing component and are installed on top of the original or an update. Fixes do not change the level of the component ID. To install a fix, you must first install a refresh change file as not removable.

Naming Change Files

The names you assign to change files follow the convention for *global names*. Using global names ensures that each change file is identified uniquely in a network environment that contains multiple servers connected across multiple domains. A global name consists of three main elements:

- The name of the component or software package
- The type of change to the component or software package (refresh, update, or fix)
- The level of change to the component or software package.

The following conventions apply when you create a global name:

- The name can have from 2 to 10 parts
- Parts must be separated by a period
- The global name cannot start or end with a period
- Each part cannot exceed 16 characters
- The global name, including periods, cannot exceed 64 characters
- Numbers, uppercase letters, and the characters $, #, @, and _ can be used in global names.
- Global names starting with *$DELETE.$PENDING* are reserved, and may not be used.

```
┌─── With the graphical interface ... ──────────────────────────┐
│                                                                │
│  Create change files in the Catalog window.  Select:           │
│                                                                │
│    ►  Change file                                              │
│    ►  Create new                                               │
│    ►  Refresh (or Update, or Fix)                              │
│                                                                │
│  To modify an existing change file, select it in the Catalog window then select: │
│                                                                │
│    ►  Selected                                                 │
│    ►  Create another                                           │
│    ►  As is (or Refresh, Update, or Fix)                       │
│                                                                │
└────────────────────────────────────────────────────────────────┘
```

```
┌─── With the command line interface ... ──────────────────────┐
│                                                                │
│  Build change files using the bld command.  This command builds a change file │
│  based on a change file profile that you create using a test editor.  Change file │
│  profiles are described online.                                │
│                                                                │
└────────────────────────────────────────────────────────────────┘
```

Change File Platform Dependencies

A change file's format must be compatible with the method used at a target to install software. When builders create change files, they must specify the type of installation procedure that will be used, which can be:

- Generic
- installp
- CID

The software installation methods supported by TME 10 Software Distribution depend on the type of workstation where the installation is being performed. All the methods described below require that change files be stored at the server. A change file can, however, contain the specification of software files to be accessed remotely for the actual installation process. This applies whether the software installation process is initiated from a server or from an active client workstation.

Generic Change Files

Generic change files are created when the generic installation method is used. It is a robust all-purpose method that can be used to install software products in the following environments:

AIX	OS/2
DOS	Solaris
HP-UX	SunOS
Macintosh	Windows
NCR	Windows NT

Change File Platform Dependencies

It can also be used to install in-house applications that do not conform to the rules for the installp method or that do not require the rich set of options that installp provides. This method replicates the files contained in a software package.

The generic method of installation is simpler and faster than the installp method.

Installp Change Files

Installp change files can be used to install the following types of software for RISC System/6000 servers and clients:

- IBM software products, such as RISC System/6000 or SNA Services
- Non-IBM products that conform to the rules for the installp installation standard
- Software that is very extensively affected by the target workstation hardware or software configuration.
- Software that requires installp options and output.

Information about software installed using the installp method is kept both in the RISC System/6000 object data manager (ODM) database and in the TME 10 Software Distribution catalog.

CID Change Files

The configuration, installation, and distribution (CID) method can be used to prepare software for clients in the following environments:

- OS/2
- DOS
- Windows

Use CID change files for:

- IBM products, such as Database Server or Communications Server
- Other products that conform to the CID standards and that are extensively affected by the target workstation hardware and software configurations

The CID method of installation uses redirection of code images from files installed on the hard disk of any workstation within the same domain as the target workstation. The code images are not required to be stored on the server. Response files that allow for unattended installation are prepared at the preparation site.

The redirection of code images can be performed by the Network File System (NFS) or by the IBM File Server/Requester. The method used depends on the workstation from which the code images are to be redirected.

The TME 10 Software Distribution Catalog

The TME 10 Software Distribution catalog is a change control history database maintained on the server. The catalog contains a list of the names of all change files and data files that are available to authorized users of client workstations across the network. It also lists the status of change files at each workstation.

Items in catalogs are identified using global names so that each is unique across a multiserver, multidomain environment. Tokens in the global name can also be used to identify the software platform a target runs under.

Items can be added to or removed from the catalog by any authorized user. The objects named in the catalog can reside on the server or on any of the clients in the domain.

The catalog can be used by all the clients associated with the server on a single-server network to check the availability of new software packages, updates to existing software packages, and data files that are available for general distribution to authorized client workstations.

Installing the Same Change File on Different Workstations

You can create change files that are *static*, meaning that the installations they trigger are always the same regardless of the workstations they are installed on, or *dynamic*, meaning that the installation procedures are different depending on the configuration of the workstations they are performed on.

Dynamic change files are defined by expressing conditions that must be satisfied before a corresponding operation takes place. You can specify dynamic conditions in relation to:

- The hardware installed on a workstation
- The software installed on a workstation
- The operating system running on a workstation
- The directory the software is to be installed in, or already resides in if the software is being uninstalled or removed
- Procedures to be run at the workstation before or after operations are performed.

Installing the Same Change File on Different Workstations

For example, you can take advantage of the flexibility offered by dynamic change files to:

- Install software on two types of workstation: small machines or big machines. To do so you would include these conditions:

```
DYNAMIC SECTION: big_machine
   CONDITION:                DRIVE_SIZE >= 180M
   PRE-INSTALL:              /usr/lpp/netwiewdm/pre-install

DYNAMIC SECTION: small_machine
   CONDITION:                DRIVE_SIZE < 180M
   PRE-INSTALL:              /usr/lpp/netwiewdm/pre-install-small

OBJECT:
   DYNAMIC SECTION:      small_machine
   SOURCE NAME:         user/lpp/netviewdm/file_small
   TYPE:                FILE
   ACTION:              COPY
```

- Mount different remote directories for installations on workstations with different operating systems: one for UNIX-based operating systems, one for Intel-based operating systems, and one for Windows NT. To do so you would include these dynamic sections:

```
GLOBAL NAME:                DIN.CF.REF.1
CHANGE FILE TYPE:           GEN
COMPRESSION TYPE:           LZW
REBOOT REQUIRED:            NO
DEFAULT TOKEN:              SRCINST(AIX) = /mnt/
DEFAULT TOKEN:              TRGINST(AIX) = /instago/
DEFAULT TOKEN:              TRGINST(ALL) = D:\INSTAGO\
DEFAULT TOKEN:              SRCINST(ALL) = Z:\
REMOVABLE:                  YES
ACTIVABLE:                  YES
INTERACTIVE:                NO
AUTHORIZE:                  NONE
SW HISTORY RESET:           NO
INSTALLATION DURATION:      00:00:00
COST:                       0
PACK FILES:                 NO
SECURE PACKAGE:             YES

DYNAMIC SECTION:  intel_mount
CONDITION: OPS.OPERATING_SYSTEM=OS/2
REMOTE DIRECTORY:
   SERVER NAME:             pwpc1
   EXPORTED DIRECTORY:      /alfi/newbld/client_2
   EXPORTED DIRECTORY:      C:\newbld\client_2
   MOUNTED FILE SYSTEM:     Z:
```

```
|         MOUNT OPTIONS:               -u -g
|         PREREQ COMMAND:              dir C:\ > C:\dir.out

|         DYNAMIC SECTION:  unix_mount
|         CONDITION: OPS.OPERATING_SYSTEM=AIX
|         REMOTE DIRECTORY:
|           SERVER NAME:               pwpc1
|           EXPORTED DIRECTORY:        /alfi/newbld/client_2
|           MOUNTED FILE SYSTEM:       /mnt
|         PREREQ COMMAND:              ls -l / > /ls.out

|         DYNAMIC SECTION:  nt_mount
|         CONDITION: OPS.OPERATING_SYSTEM=WINDOWS_NT
|         REMOTE DIRECTORY:
|           SERVER NAME:               650
|           EXPORTED DIRECTORY:        source.nt
|           MOUNTED FILE SYSTEM:       Z:
|           MOUNT OPTIONS:             /user:guest ""
|         PREREQ COMMAND:              dir D:\ > D:\dir.out

|         OBJECT:
|             DYNAMIC SECTION:         intel_mount
|             SOURCE NAME:             /mnt1/client_2/fnd4mbs.dll
|             SOURCE NAME:             C:\newbld\client_2\fnd4mbs.dll
|             SOURCE NAME AT INSTALL:  $(SRCINST)fnd4mbs.dll
|             TARGET NAME:             $(TRGINST)fnd4mbs.dll
|             TYPE:                    REMOTE_FILE
|             ACTION:                  COPY
|             INCLUDE SUBDIRS:         NO

|         OBJECT:
|             DYNAMIC SECTION:         unix_mount
|             SOURCE NAME:             /mnt1/client_2/fnd4mbs.dll
|             SOURCE NAME AT INSTALL:  $(SRCINST)fnd4mbs.dll
|             TARGET NAME:             $(TRGINST)fnd4mbs.dll
|             TYPE:                    REMOTE_FILE
|             ACTION:                  COPY
|             INCLUDE SUBDIRS:         NO
```

With the graphical interface ...

Create dynamic change files in the Catalog window. Select:

- ► Change file
- ► Create new
- ► Refresh (or Update, or Fix)
- ► In the Change File window select the **Condition** and the **Dynamic Sections** push buttons

Installing the Same Change File on Different Workstations

With the command line interface ...

Build change files using the **bld** command. This command builds a change file based on a *change file profile*, in which you can include conditions and dynamic sections. Change file profiles are described with the reference information provided with the product.

Chapter 4. Using Change Control Operations

This chapter describes change control operations made available through TME 10 Software Distribution, and how to schedule their use in your network.

Change Control Operations

TME 10 Software Distribution offers you a fully automated method for distributing and installing software. You can use it to schedule and install software and updates to previously installed software, as well as to uninstall updates and restore the software to its previous state. TME 10 Software Distribution performs the following functions in a network:

- **Distributes software packages**

  Software packages can be distributed to remote servers. From the server the software in the package can then be installed on the clients in its domain.

  Alternatively, software packages distributed to a server can include commands to mount remote directories where software files reside. The remote files are accessed from the client during the install operation.

- **Installs software packages on workstations**

  A new software package can be installed automatically, with no user intervention, on all workstations in the network. Installed software is available for use immediately after a client is restarted. This is done using the *install* function for a *refresh* change file.

  Software changes can also be distributed over a network so that software files are either updated or removed automatically on all workstations on which the software was originally installed. An installation request can also check that prerequisite and corequisite software is present on a target before installing software packages.

- **Applies updates to a current software level**

  When a new level of a software product is released, it can be applied to existing software as an update.

- **Applies fixes to software**

  When fixes to installed software are released, these can be applied to all existing installed copies. This is done using the *install* function for a *fix* change file.

- **Installs software from remote workstations**

  Software files to be installed can be stored in a remote directory that is mounted at a target during the installation process. When you do not mount remote directories for installations, files are sent to a target, stored in the work area during installation, and then deleted when the installation is complete. This means that a target must have double the amount of disk space required by a product.

  When you take advantage of the remote mount capabilities offered by TME 10 Software Distribution, you can redirect installations:

Using Change Control Operations

- To directories located on a server.

  You do so by specifying the name of the directory to be exported and the options required for the remote mount using specific shared tokens (SERVERREPOS, SEVEREXPOP, REMOTEREPOS, REMREPMNTOP). You specify them when you define target parameters using the **addpm** command.

- To directories located on another workstation in the domain.

  You do so by specifying the directory to be exported and the remote mount information when you build a change file using either the graphical interface or the **bld** command.

- *Installs software on unattended workstations*

  You can use TME 10 Software Distribution to send a command to a client workstation to install, update, or delete changes to its software while it is unattended and has no user logged on.

- *Rolls back (backs up) to the previous level of software*

  An installation operation can also include a request to back up the version of software currently installed on a workstation. Any files deleted or modified by the installation process are automatically saved, and if the installation is unsuccessful, the previous version can be restored, or rolled back. The current level of the application must be *installed* with the *removable* option in order for the *remove* function to perform a roll-back.

- *Makes changes permanent*

  If the software package was installed on one or a group of clients with the backup option, a single *accept* request to make this version permanent can be submitted to all the clients. After software has been accepted, the previous version or level cannot be restored.

- *Uninstalls one or more software packages installed previously by TME 10 Software Distribution*

  The *uninstall* function removes, in a single operation, selected software packages from one or more clients. The software packages must have been installed originally by TME 10 Software Distribution.

- *Activates software on one or a group of workstations*

  The *activate* function restarts one or a group of workstations in a single operation after TME 10 Software Distribution has been used to install software packages on them.

With the graphical interface ...

Specify change control operations in the Catalog window. Select the change file you require, then select:

▶ Selected
▶ Accept, Activate, Install, Remove or Uninstall

```
┌─ With the command line interface ... ──────────────────────────┐
│                                                                │
│  Specify change control operations using the acc, act, inst,   │
│  rem, and uninst commands.                                     │
│                                                                │
└────────────────────────────────────────────────────────────────┘
```

Scheduling Change Control Operations in the Network

You can schedule the distribution of data and software and change control operations to occur at specific times, according to when it is both necessary and most convenient. Using appropriate schedules you can achieve network-wide synchronization of the installation of particular software. The scheduling facilities provided by TME 10 Software Distribution are flexible, and provide you with alternative methods to express when and how often operations should take place. You can:

- *Specify change control windows for targets*

  Each client workstation that is a target for software installation can be configured with a *change control time window*, a period of time during which installation of the software can take place. Installation of software that is scheduled for times outside the change control window for a client workstation is suspended until the change control window is open.

  You specify change control windows when you define targets to the network.

- *Specify origin or destination operation times*

  You can schedule operation times for either the workstation where the operation originates, or the workstation where the operation is to take place. This makes it possible to schedule requests that are to take place in time zones different from the zone of the origin workstation. You can also schedule a period of time *before* and *after* which an operation cannot take place.

- *Specify high-priority operations*

  Change control operations that are urgent can be transmitted as high-priority, meaning that they will take place as soon as the currently executing operation has completed.

- *Specify recursive operations*

  A *recursive* operation is one that is performed repeatedly at specified intervals. Intervals can be specified on a monthly, weekly, daily, or hourly basis, depending on business requirements.

  The execution of occurrences of recursive plans can be conditioned on the success of previous occurrences. This means that a subsequent transmission can take place only if the previous one produced the expected result. If Monday's sales data was not sent successfully, Tuesday's occurrence of the same operation will not be generated and transmitted.

Scheduling Requests Together

- *Hold operations at targets*

  A user at a client workstation can temporarily suspend or prevent the execution of change control activities requested by the administrator by holding change control activity or by turning the workstation off. The functions are held and are executed when the workstation is switched on or the workstation is released.

With the graphical interface ...

Schedule change control operations in the Catalog window. Select the change file you require, then select:

- ► Selected
- ► Accept, Activate, Install, Remove or Uninstall
- ► The Schedule and Frequency push buttons

With the command line interface ...

Schedule change control operations using the-d, -t, and -r parameters for the **acc**, **act**, **inst**, **rem**, and **uninst** commands.

Scheduling Requests Together

If the plan option is installed on your system, you have to facilities to create *transmission plans*. A transmission plan groups change control and distribution operations into a single request that is submitted for execution. When you group entries in a plan you can manage complex change control activities automatically by:

- Establishing the sequence in which requests are to be executed.

- Executing the same request at multiple targets.

- Specifying conditions so that a request is executed only if a previous request in the same plan completes successfully. Conditions can be expressed at the individual target, the domain, and the request level.

- Executing requests repeatedly according to a set schedule.

The following sample plan gives you an idea of how you can take advantage of transmission plans to automate activities. In Figure 5 on page 37, a change file (excel.ver40.ref.1) is retrieved from a preparation site, sent to the servers in a network, and then installed on all the clients in the servers' domains.

```
Global Name:          plan.install2

Request:
        ID:           rtrv1
        Function:     rtrv excel.ver40.ref.1 C11
        Destination:  R1

Request:
        ID:           send1
        Function:     send excel.ver40.ref.1
        Destination:  S1 S2 S3
        Condition:    EE(rtrv1) = NORMAL

Request:
        ID:           install1
        Function:     inst excel.ver40.ref.1 -n -va
        Destination:  c11 c12 c21 c22 c31 c32
        Condition:    ED(send1) = NORMAL
```

Figure 5. A Transmission Plan

With the graphical interface ...

Define plans in the Catalog window. Select:

- ▶ Catalog
- ▶ Plan
- ▶ Create new

With the command line interface ...

Create plans using the **addpln** command. This command creates a plan based on a *profile* that you create using a test editor. Plan profiles are described in online reference documentation.

Distributing Data Files

The term *data distribution* refers to the exchange of data files between client and server workstations within and across domains. You can use TME 10 Software Distribution facilities to distribute or retrieve data files such as:

- User data files
- Software packages and documentation
- Problem management data files such as dumps, log files, and trace files

TME 10 Software Distribution can distribute data files whether they have been cataloged or not. In addition, it provides facilities to compress and translate data during distribution.

Distributing Data Files

Compressing Data Files

When large volumes of data are transmitted across a network, transmission efficiency is a significant factor in reducing time and cost.

TME 10 Software Distribution can send or retrieve data in compressed form. The compression format used can be either SNA, LZW, or based on a user-supplied algorithm. Compression substantially reduces both line transmission time and storage space requirements. After transmission, data can be expanded into its original form.

Data compression facilities are available for data exchange within and across domains when your network uses the server-to-server communication protocol.

Translating Data

When different environments in your network use different data encoding techniques, TME 10 Software Distribution can perform data translation on files containing character data. A typical example of this requirement is transmitting data between systems that use ASCII encoding and those that use EBCDIC encoding.

Data translation facilities are available for data exchange within and across domains.

Data Distribution Functions

The distribution functions of TME 10 Software Distribution are:

Send   The send function is used to send a file from a source target, which must be either the server or a local client workstation, to a destination target or group of targets, which can be either local or remote. The send function can be used from both server and client workstations. The action can be immediate or deferred until a specified time and date. It can also be issued with the replace option.

Retrieve   The retrieve function is used to retrieve a file from another target, which can be local or remote, onto a local server or client workstation. The retrieve function can be used from both server and client workstations. The action can be immediate or deferred until a specified time and date.

Delete   The delete function is used to delete files at local or remote targets.

With the graphical interface ...

Distribute files in the Catalog window. If files are cataloged, select the file you want to send, retrieve, or delete and then select:

- ► Selected
- ► Send, retrieve, or delete

If files are not cataloged, in the Catalog window select:

- ► Distribution
- ► Send file, Retrieve file, or Delete file

> **With the command line interface ...**
>
> Distribute cataloged files using the **send**, **rtrv**, or **del**, commands.
>
> Distribute non-cataloged files using the **sendf**, **rtrvf**, or **delf**, commands.

Installing on Pristine Workstations

You can use TME 10 Software Distribution in conjunction with a network file server to install software on pristine workstations which have no software installed on them. In the AIX/6000 environment, you can use specific facilities provided with the TME 10 Software Distribution tool option to:

- Perform the first-time installation of AIX/6000, of TCP/IP (including NFS) and of TME 10 Software Distribution Client or TME 10 Software Distribution, Version 3.1.5 for AIX in unattended mode.

- Clone an AIX Version 4.1 workstation using a backup image.

- Clone an AIX Version 3.2.5 workstation using a backup image or using a high performance data transfer method.

Scenarios that describe the step-by-step procedure you must follow to perform pristine installations are included in the DynaText *Installation Scenarios* file.

After a pristine workstation is installed, you use TME 10 Software Distribution to:

- Define the RISC System/6000 workstation to the network

- Perform change control and distribution operations to the workstation

- Maintain all the software and applications that are installed on the workstation

You can also use TME 10 Software Distribution to maintain the system software products stored on the network file server so that they are ready to be installed on future pristine workstations.

Chapter 5. Tracking Network Operations

This chapter describes the facilities provided by TME 10 Software Distribution to track the hardware and software configuration of the workstations in your network.

Keeping Track of Workstation Hardware Configurations

You can use TME 10 Software Distribution facilities to keep track of the hardware configuration of the workstations in your network. This information is referred to as hardware inventory.

You run an inventory discovery procedure to detect what hardware is present on a workstation. The output of the procedure is stored in a file in the server catalog. You can then make use of this hardware inventory information when you formulate the conditions for creating dynamic groups and change files, and for change control and distribution requests. For instance, you can specify that a change file be installed only on workstations that have at least 10 MB of disk space available. Or you can define a dynamic group that contains only workstations with a graphic adapter installed, and install a graphics program only on those workstations.

Inventory discovery is not a mandatory part of TME 10 Software Distribution. You can specify the hardware present on a target when you define it by specifying hardware parameters. However, the inventory discovery process is a way of catching this information and keeping it up to date automatically.

You can run inventory discovery on remote targets if the targets involved in the operation are connected by the server-to-server (STS) transmission protocol.

Several inventory discovery programs are provided for your use. Some can be used on AIX targets, while another is available for OS/2 and Windows targets:

AIX inventory discovery

Hardware discovery processes for AIX are installed as part of TME 10 Software Distribution, Version 3.1.5 for AIX under /usr/lpp/netviewdm/bin. Three programs are available:

fndinvhs   Discovers software and hardware
fndinvhw   Discovers hardware
fndinv   Discovers software

The discovered hardware and software data is collected and stored at the target in these files:

fndtkinv   Stores discovered installation parameters
fndswinv   Stores discovered software
fndhwinv   Stores discovered hardware

From these files the data is then transferred to the server's database.

OS/2 and Windows inventory discovery

A program that takes advantage of NetFinity inventory capabilities on workstations that run OS/2 and Windows TME 10 Software Distribution products is provided. It is called `fndinv`, and it creates at each target output files (`fndswinv`, `fndhwinv`, and `fndtkinv`) that are then used by TME 10 Software Distribution.

Inventory data is transferred from clients to the server when you do either of the following:

- From the command line interface, submit the **nvdm inv** command.
- From the graphical interface, select **Inventory** from the **Selected** pull-down menu in the Targets window.

Note that for fndinv to work, the OS/2 or Windows client must have the Hw/Sw Discovery Tool component installed.

For other operating systems, you can either purchase an inventory discovery program from a third party, or write one in yourself as described in the inventory discovery documentation provided online.

Keeping Track of Workstation Software Configurations

You can keep track of the software installed on the workstations in your network using two methods:

- Running software inventory procedures
- Referring to the change control status information in the catalog.

Software Discovery Information

You can run inventory discovery procedures to detect the software installed on workstations. These procedures are similar to those described for hardware inventory in "Keeping Track of Workstation Hardware Configurations" on page 41.

Change Control Status Information

Change control status is stored in the server catalog for each software package installed on each client workstation in a domain. The catalog also includes the change control history for the software installed on the server itself. When a change control operation is executed on a client workstation, a report of the operation results is generated and routed to the workstation in the network that originated the operation. This workstation can in turn be connected in a hierarchical arrangement to a central site, which acts as a repository for reports. This means that all change control reports can be stored in a central TME 10 Software Distribution catalog (see "Tracking Hardware and Software at Central Sites" on page 45 for more information).

The status information in the catalog can report a change file as being:

Active/Inactive: The change file is installed either in the active area of the target file storage or only in the service area.

Available/Not authorized: The change file is stored on the server and is available for use by the target, or the target has not been authorized to use it.

Back level: The change file is a previous version of the software that was backed up after a more recent version was installed, but is reactivated because the more recent version was removed.

Discovered: The change file was discovered by an inventory discovery procedure. It is installed on the workstation, but was not installed by a TME 10 Software Distribution procedure.

Distributed: The change file was distributed to this target from another target.

Distribution pending: The change file is currently being distributed.

In error: A change control operation using the change file has failed without recovery, leaving the installation in an unpredictable state.

Installed/Not installed/In progress: The change file is either installed, is not installed, or the installation is currently in progress.

Reboot required: When the change file was installed, REBOOT REQUIRED=YES was specified. To use the new software, restart the workstation so that the changes in the file become operative.

Removable/Not removable: The change file is installed on the target and can be removed and replaced with an earlier version (rolled back), or cannot be removed.

Before sending change files to clients, you can check the status of the software on the client. The operations that you can schedule for a client depend on the status of software, as shown in Table 6 on page 44. You can use change control status information to specify conditions in dynamic change files and dynamic groups (see "How Grouping Targets Can Facilitate Change Control Operations" on page 22 and "Installing the Same Change File on Different Workstations" on page 29). For example, you can schedule a remove operation only on those workstations in your network where the status of a change file is *in error*.

Tracking the Progress of Change Control Operations

Status	Install	Accept	Activate	Remove	Uninstall
Table 6. Change File Status and Associated Functions					
Available	Yes	No	No	No	No
Distributed	Yes	No	No	No	No
In error	Yes	Yes	Yes	Yes	Yes
In progress	No	No	No	No	No
Installed, Removable, Active	No	Yes	No	Yes	Yes
Installed, Not removable, Active	No	No	No	No	Yes
Installed, Removable, Inactive	No	Yes	Yes	Yes	Yes
Installed, Not removable, Inactive	No	No	Yes	No	Yes

With the graphical interface ...

Check change file history in the Catalog window. Select the change file and then select:

- ► Selected
- ► Open
- ► History

You can also check change file from the Target window. Select a target and then select:

- ► Selected
- ► Open
- ► History

With the command line interface ...

List change file statuses using the **lscm** command.

Tracking the Progress of Change Control Operations

Change control and distribution operations that have been submitted to the system for execution are referred to as *requests*. You can display requests to follow their progress and take any necessary remedial action. Detailed filtering mechanisms allow you to display only those requests you are immediately interested in. You can view the progress of requests according to:

- • The servers they were submitted from
- • The request queue sequence number
- • The global name of the files included in them
- • The type of operation being performed (install, remove, accept, activate)
- • Their status (in progress, successful, waiting, held, deleted, failed, not started)
- • The severity level returned for errors (failed, warning, severe, hardware failure)

- The scheduled date and time
- The domains or the targets they address

You can then act upon a request by:

- Holding or releasing its execution
- Deleting it
- Rescheduling it
- Restarting it

With the graphical interface ...

Check change file history in the Catalog window. Select the change file and then select:

- ► Selected
- ► Open
- ► History

You can also check change file from the Target window. Select a target and then select:

- ► Selected
- ► Open
- ► History

With the command line interface ...

List change file statuses using the **lscm** command.

Tracking Hardware and Software at Central Sites

TME 10 Software Distribution allows you to route change control reports to one or more central sites in a network, usually referred to as *focal points*. This capability is available when the remote administrator option is installed on servers.

When remote servers are linked to each other in a hierarchical arrangement (meaning that a downstream workstation is subordinate to all workstations located upstream to it) reports can be sent from lower to higher levels in the hierarchy. For example, in Figure 6 on page 46, a hierarchy is formed by Focal, Manager1, and Server1. Manager1 is subordinate to Focal, and Server1 is subordinate to both Manager1 and Focal. Higher levels can perform change control and distribution operations on lower levels, and reports can be routed from lower levels to higher levels according to the *change management modes* defined at each server.

Tracking Hardware and Software at Central Sites
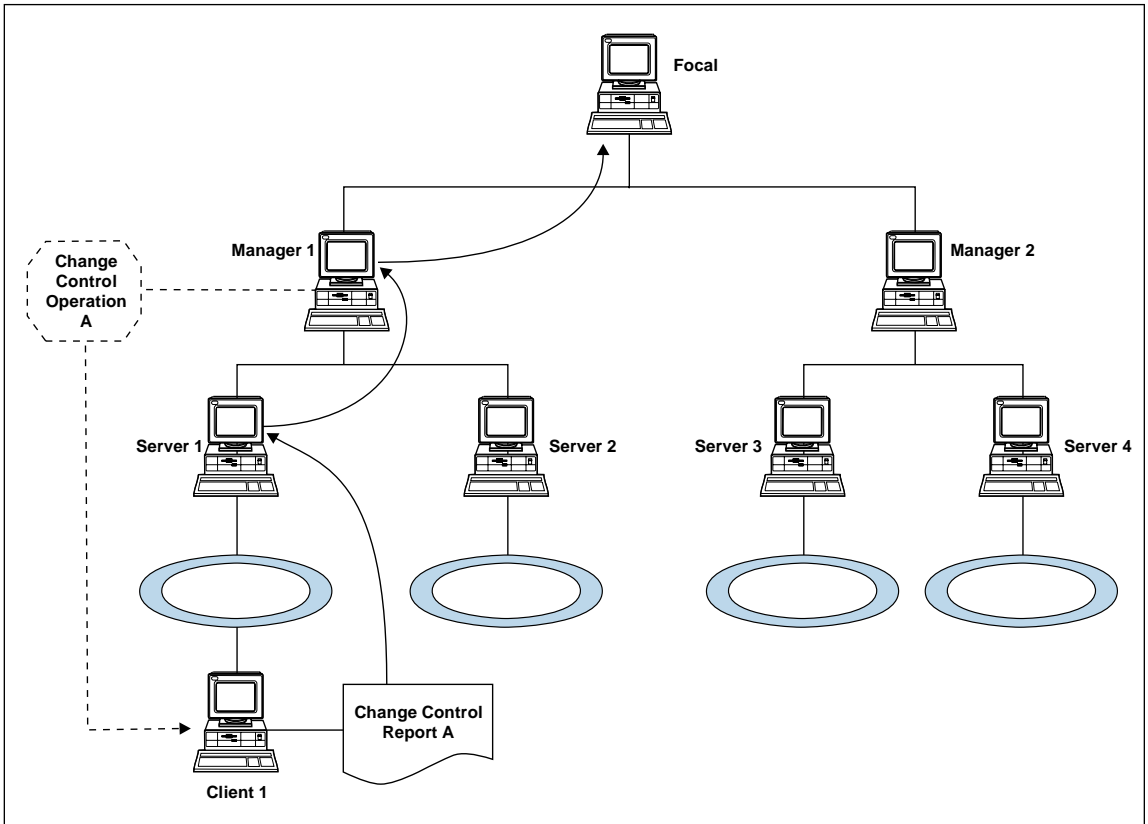
Figure 6. Change Control reporting to a central site

You must be careful when configuring the mode attribute of your server targets, to make sure the correct network role is assigned to each one. The mode can be either of the following:

Manager   Managers are servers which must have the remote administrator option installed. When a server is defined as another server's manager, it can perform change control operations on it and consequently on the clients in its domain. Thus in Figure 6, Server1 and Server2 must define Focal as their manager to allow it to perform operations on them. They will define Manager1, the server directly adjacent to them in the hierarchy, as their focal point.

Focal point   A focal point must have the remote administrator option installed. It can perform the same change control and distribution functions as a manager target. In addition, it receives all change control reports from downstream servers that have defined it as their focal point. Servers can have more than one focal point. Thus in Figure 6, Server1 and Server2 must define Manager1 as their focal point, and Manager1 must specify Focal as its focal point. Consequently, those reports routed to Manager1 from Server1 and Server2 are routed to Focal as well. Focal

thus holds the central catalog for this hierarchy, as it stores change control history records for all its subordinate workstations.

A focal point can also be configured on a single-node target, in which case it can act solely as a report repository for a network without performing change control and distribution operations.

Using Multiple Managers

The change control functions described in the previous sections can also be carried out using several servers configured as managers in different parts of an organization, all of which could be supervised from central focal points. Managers can be linked to more than one focal point, according to your network needs. For example, it is often wise to configure a *backup focal point* that stores the same information as your main focal point, and which can be put into active use should any problems occur at your main focal point site.

You can handle a large network as a number of smaller networks, each with a manager, and all of which are connected in a hierarchy to a central focal point. Each site can perform operations on downstream TME 10 Software Distribution domains in the network, and store change control history records for them. After a change control operation is processed on a client, a report is routed back through the client's manager to the upstream focal point, where complete history records are kept for the entire network. Figure 6 on page 46 shows a very simple example of such a network.

With a hierarchy of TME 10 Software Distribution manager sites, focal points, and intermediate nodes you can optimize distribution across a network. Change control activities can be staged and, if required, part of the administration can be delegated to regional or local levels while central control is maintained.

Chapter 6. Using Security Functions

TME 10 Software Distribution provides security mechanisms that you can use to prevent unauthorized use of product functions and safeguard the privacy of sensitive data. You can define:

- User profiles that limit the functions a user can perform
- User access to targets
- User access to data
- Target authorizations for specific change files
- Security checks to be performed on the files in change files before they are installed on workstations.

TME 10 Software Distribution users must have a user ID to access the product. You can provide additional security by assigning passwords to users as well.

The following sections describe the authorizations contained in user profiles, and how to define user access to data and to targets.

Defining User Authorization Profiles

User authorization profiles define the functions that each user can perform with the product. You can define profiles to be used by groups of users who perform similar tasks, or you can define individual profiles for each user.

You can take advantage of three default authorization profiles, which are created during installation and stored at the server.

Administrators (FNDADMN)

Administrators have access to *all* operations, including the administrative and configuration functions.

When you define user authorizations, keep in mind that all FNDADMN users can perform operating system specific operations that require root and bin authorizations.

Table 7 on page 50 lists the values set for this profile.

Defining User Authorization Profiles

Table 7. Administrator Profile (FNDADMN)	
Function	**Authorization**
Change management	Install, Remove, Accept, Uninstall Execute Activate Authorize, Unauthorize, Delete History All targets
Distribution	Send Retrieve, Delete, Replace
Preparation	Build, Unbuild, Catalog, Delete, Create, View
Queues	Manage
System administration	Modify
Configuration	Modify
Erase requests	Authorize
Manage requests	All

Builders (FNDBLD)

Builders are authorized to perform change control preparation functions. That is, they can prepare and build change files. Table 8 lists the values set for this profile.

Table 8. Builder Profile (FNDBLD)	
Function	**Authorization**
Change management	Install, Remove, Accept, Uninstall Execute Activate All targets
Distribution	Send Retrieve, Delete, Replace
Preparation	Build, Unbuild, Catalog, Delete, Create, View
Queues	View
System administration	View
Configuration	View
Erase requests	Authorize
Manage requests	All

Users (FNDUSER)

Users can distribute files and display the configuration. Table 9 lists the values set for this profile.

Table 9. User Profile (FNDUSER)	
Function	**Authorization**
Change management	Install, Remove, Accept, Uninstall Execute Activate Only on target where the user is working
Distribution	None
Preparation	None
Queues	View
System administration	None
Configuration	None
Erase requests	No authorization
Manage requests	None

With the graphical interface ...

Define user profiles in the Catalog window. Select:

- ► System
- ► Authorization
- ► Authorization profile

With the command line interface ...

Define user profiles using the **addprf** command. Modify profiles using the **updprf** command, and delete profiles using the **delprf** command.

Defining User Access to Data and Targets

Define data access keys (DAKs) to determine the objects in the catalog that a user can work with, and define target access keys (TAKs) to determine the targets a user can access. You define them in the following way:

- When you build or catalog a change file, a data file, or a plan, you specify the DAK associated with it. A catalog entry can have only one DAK, but you do not have to assign it one at all if it is not necessary. No DAK is the default.

  In the same way, when you define a target you specify the TAK associated with it.

- When you define a user, you determine the catalog entries the user can access by specifying the DAKs the user is associated with, and the targets the user can access by specifying the user's TAKs. A user can have up to 32 DAKs and TAKs (which is the maximum number defined to the system). The default is no DAKs or TAKs, in which case the user can work only with objects or targets that do not have DAKs or TAKs.

  When a user issues a command against a catalog entry or for a target, the system checks to verify that the object or target and the user have the same DAK or TAK.

With the graphical interface ...

Define users and the DAKs and TAKs associated with them in the Catalog window. Select:

- ► System
- ► User

Define DAKs and TAKs that you can then associate with data or targets in the Catalog window. Select:

- ► System
- ► Authorization
- ► Data Access Key (or Target Access Key)

With the command line interface ...

Define users using the **addusr** command. Modify users using the **updusr** command, list user information using the **lsusr** command, and delete users using the **delusr** command.

Define DAKs and TAKs using the **updak** command. List access keys using the **lsak** command.

Defining Data Security

Since change files are usually prepared at preparation site workstations, then stored on servers before being distributed around a network and installed, it is important to ensure that their contents are not tampered with before an actual installation takes place. This is especially critical when a change file contains sensitive data.

TME 10 Software Distribution provides the following mechanisms:

- When you are building a remote source change file, specify **Secure package** in order to:

  - Verify that the remote files specified are accessible, when the change file is built.

  - Verify that the files are *the same* as those specified when the change file was built, when the change file is installed.

- In addition, you can activate user exits that perform the following functions in connection with the secure package attribute:

  - When the change file is built, assign a *secure key* to it by calculating its CRC (Cyclic Redundancy Check) number.

  - When the change file is installed at a client, the client product proceeds with the installation only if the secure key is the same as the one originally assigned, and only if the secure key identifies the workstation where the change file was built.

Chapter 7. Finding and Using TME 10 Software Distribution Information

This chapter tells you where you can find more information about TME 10 Software Distribution functions and how to use them.

It makes reference to files that can be found on the media on which the product is provided. TME 10 Software Distribution is provided on two CD-ROMs:

CD Number	CD Name	Product Number
LK3T-5087-00	TME 10 Software Distribution, Version 3.1.5 for OS/2, Windows 2000, Windows NT, Windows 95/98, Windows 3.1x, NetWare	5698-SWD
LK3T-5088-00	TME 10 Software Distribution, Version 3.1.5 for AIX	5698-SWD

Where to Find Installation and Configuration Information

You can find information about how to install and configure the product as follows:

Installing the product
Part 2, "Installing and Configuring TME 10 Software Distribution" on page 61 lists the hardware and software prerequisites for TME 10 Software Distribution, Version 3.1.5 for AIX and tells you how to install the products. It also describes how to configure the base configuration files at servers and clients, and how to configure communication protocols.

Configuration scenarios
The *Installation Scenarios* file (see "On-Line and Printable Documentation" on page 58) contains a series of scenarios that help you to configure various networks topologies such as:

- Single-server networks
- Multidomain networks
- Centrally managed multidomain networks
- Server-to-server links
- Networks with NetView DM for MVS
- Intermediate nodes in networks

Pristine workstation scenarios
The *Installation Scenarios* file (see "On-Line and Printable Documentation" on page 58) contains scenarios that describe how to use TME 10 Software Distribution to:

- Clone an AIX Version 4.1 workstation using a backup image

- Migrate from AIX Version 3.2.5 to AIX Version 4.1

- Clone an AIX Version 3.2.5 workstation using a backup image or using a high performance data transfer method

Finding and Using TME 10 Software Distribution Information

Installing the TME 10 Software Distribution Clients

The *TME 10 Software Distribution Clients Installation and Configuration* manual provides the information to install and configure clients for the OS/2, AIX, Windows 2000, Windows NT, Windows 98, Windows 95, and Windows 3.11 platforms.

Graphical Interface Help

When you are using the graphical interface, help is always available. Five categories of help can be requested:

General help

Explains how to use a window. To access it, use one of the following methods:

- Press F1 while on a main window
- Select **General help** from the **Help** menu on a main window
- Select the **Help** push button on a window
- Press the F2 key while displaying contextual help
- Press the F1 key while the input focus is on the **Help** push button on a window
- Select an item from the Help index
- Press the F2 key while displaying Help for help.

Contextual help

Explains how to use a particular field or control on a window. To view this type of help, press the F1 key while the input focus is on the relevant field or control.

Menu item help

Displays help on the action that is taken when you choose a particular menu item. Access it by pressing the F1 key while the menu item is highlighted. Select the menu item using the keyboard, because doing so with the mouse causes the requested action to be processed.

Help index

Displays a list of all available general help topics. You can view the help associated with a topic by selecting that topic.

Access the Help Index by selecting **Help index** from the **Help** menu on a main window, or by pressing F11 while viewing another help window.

Help for help

Displays help about how to use this help system. To view this information, select **Using help** from the **Help** menu on a main window.

Help is displayed in a new primary window. If the help information is more than can fit on one window, you can scroll Up and Down by using either the up or down arrow keys or the scroll bar. You can also resize this window. When you have finished reading the information, use the Windows menu or click on another window to return to the operation you were performing. You can also exit by pressing Alt+F4 or clicking on the bar to the left of the window.

Command Line Interface Information

When you are using the command line interface information you can look for information:

For all commands

In the *TME 10 Software Distribution Reference* (see "PostScript Format" on page 58). This online book contains a general description of command syntax and how to enter commands, the syntax for all product commands, and a description of how to edit change file profiles.

For individual commands

You can display just the parameters for a command by entering the `help` command followed by the name of the command itself. For example, to display the parameters that are accepted for the `inst` command, from a command prompt enter:

`nvdm help inst`

You can display the complete syntax for a command and a full description of all its parameters by entering the `man` command. For example, to obtain information about the `addtg` command, enter:

`man nvdm_addtg`

Information about Messages and Error Codes

All TME 10 Software Distribution messages are logged in the message log. You can view it in the Message Log window of the graphical interface, or by entering the `nvdm log` command from a command prompt.

Other User Information Is Available

Appendixes in this manual document how to use the following functions:

- Implementing inventory discovery
- Writing change control scripts
- Replacing the quiesce check
- Adding user-defined compression methods
- Writing user exits
- Using TME 10 Software Distribution from NetView/6000

On-Line and Printable Documentation

TME 10 Software Distribution documentation is available on the product CD in PostScript and PDF formats.

PostScript Format

Files of the TME 10 Software Distribution documentation formatted for a PostScript printer are supplied with the PS option. If this option is installed, the `/usr/lpp/netviewdm6000en_US/ps` directory contains the following files:

Filename	Document name
FNDNTMST.PS	TME 10 Software Distribution for Windows NT Quick Beginnings
FNDOSMST.PS	TME 10 Software Distribution for OS/2 Quick Beginnings
FNDNWMST.PS	TME 10 Software Distribution for NetWare Quick Beginnings
FNDAXMST.PS	TME 10 Software Distribution for AIX Quick Beginnings
FNDR6MST.PS	TME 10 Software Distribution for Windows NT and OS/2 Command Reference
FNDA6MST.PS	TME 10 Software Distribution for AIX Reference
FNDNRMST.PS	TME 10 Software Distribution for NetWare Command Reference
FNDCIMST.PS	TME 10 Software Distribution Clients Installation and Configuration
FNDM6MST.PS	TME 10 Software Distribution Message Reference
FNDS6MST.PS	TME 10 Software Distribution Pristine and Migration Scenarios
FNDI6MST.PS	TME 10 Software Distribution for AIX Installation Scenarios
README.PS	TME 10 Software Distribution README

You can print them before or after you install TME 10 Software Distribution, Version 3.1.5 for AIX, using whatever method you have set up for printing to a PostScript printer. Because the files are large, use of a high-speed PostScript printer is recommended.

PDF Format

The TME 10 Software Distribution manuals are also available in PDF format, allowing them to be viewed or printed using Adobe Acrobat Reader, which can be downloaded free of charge from Adobe's Internet site (www.adobe.com); see the site for full details of platforms supported by Acrobat. The directory which holds the PostScript files also holds the same manuals in PDF format, where the filename of the manual is the same as that of the PostScript version but the extension is changed to **.**PDF.

The manuals are fully hyperlinked, so that you link directly to pages from the Table of Contents and the Index, as well as from specific references in the text.

Calling IBM Service

The IBM Support Center provides telephone assistance in problem diagnosis and resolution in the United States and Puerto Rico. You can call the IBM Support Center at any time; you will receive a return call within eight business hours (Monday through Friday, 8 a.m. to 5 p.m., your local time). The number to call is (800) 237-5511.

Outside the United States and Puerto Rico, contact your local IBM representative or your authorized IBM supplier.

Part 2. Installing and Configuring TME 10 Software Distribution

This part describes what you must do to install the product and configure the network it will be running in. It includes these chapters:

Information about how to install and configure OS/2 and Windows clients is in the *Installation and Configuration* book for the client products.

Chapter 8. Installation Requirements

This chapter introduces TME 10 Software Distribution, Version 3.1.5 for AIX product options, and lists the software and hardware requirements that must be met before installing the product.

Identifying Your Network Configuration

Before installing TME 10 Software Distribution, Version 3.1.5 for AIX, identify the network configuration it will be running under, and the options and transmission protocols required for that environment. The *Installation Scenarios* book provided in PostScript and PDF formats illustrates many typical TME 10 Software Distribution, Version 3.1.5 for AIX networks, and you may want to consult it to help you identify your network configuration.

In addition to the types of servers, clients, and perhaps intermediate nodes included in your network, you must also decide what kind of transmission protocol connects the various entities. See "Communication Protocols that Can Be Used to Link Nodes" on page 21 for information.

TME 10 Software Distribution, Version 3.1.5 for AIX Packages and Options

TME 10 Software Distribution, Version 3.1.5 for AIX is distributed as three separately orderable packages, each of which includes different product options. When you install a package, you can select which options to install on your system. Each package responds to specific operational needs, as described in the following sections.

When you order any of these packages, you are also provided with an additional package containing national language support options. It includes product messages, graphical interface text files, and product manuals in DynaText and PostScript formats.

Single Node Package

The single node package provides the functions typically used at a single node or an intermediate node in a network. A single node can be used as a change file preparation site. It can initiate distribution activities on other network nodes, and can initiate change control activities on itself (in pull mode). Intermediate nodes are used to store and fan out change control and distribution requests.

The product options in this package are:

- Base
- Commmunication
- Graphical interface
- Tool

63

TME 10 Software Distribution Packaging

Server Package

The server package provides server functions.

The product options in it are:

- Base
- Server
- Communication
- Graphical interface
- Tool

Plan Package

The plan package provides functions for creating and managing transmission plans, as well as functions required for focal point targets. The product options in it are:

- Remote administrator
- Plan

You must purchase either the single-node package or the server package together with the plan package to have a functioning system.

The following sections describe the product options that compose the packages described above.

Base Option

You must install the base option on all TME 10 Software Distribution, Version 3.1.5 for AIX workstations. It provides the core TME 10 Software Distribution, Version 3.1.5 for AIX functions.

All change control and distribution is performed by the base option. This includes support for file and command exchange with remote nodes, storing and cataloging received files, routing commands within the NetView DM for MVS system, and scheduling change control and file distribution commands.

The base option on its own does not provide a runnable system. You must also install at least the communication option or the server option to have a working system.

Communication Option

You must install the communication option on all servers or single nodes that require remote communication.

The base option is a prerequisite to the communication option. It also requires either TCP/IP or SNA.

Server Option

The server option provides server functions to servers that communicate with clients. Without the server option, the server cannot recognize clients, and is not able to route change control commands to them or receive commands from them.

The base option is a prerequisite to the server option. It also requires TCP/IP or SNA.

Remote Administrator Option

When this option is installed on a TME 10 Software Distribution, Version 3.1.5 for AIX workstation, it can issue requests for change control operations to remote targets. A workstation with this option installed can also act as a focal point node, which can store change control reports for all targets under its control at lower levels in a network.

The base and communication options are prerequisites to the remote administrator option. They are included in the single node package and the server package.

Plan Option

This option provides the functions for creating transmission plans. Plans group numerous distribution and change control operations into a single request.

The base option is a prerequisite to the plan option. It is included in the single node and the server package.

Tool Option

This option contains utilities and files that aid you in performing some more complex operations. It includes:

- Profiles and scripts that you can use to install TME 10 Software Distribution, Version 3.1.5 for AIX on workstations in a network, using TME 10 Software Distribution, Version 3.1.5 for AIX itself
- Scripts that integrate TME 10 Software Distribution, Version 3.1.5 for AIX with NetView/6000
- Utilities that automatically create change file profiles from installp images, and extract prerequisite information from installp images
- A script that can be used to switch from one language to another to view online man pages.
- A compression utility that recovers unused space in some TME 10 Software Distribution files.

Graphical Interface Option

This option installs the graphical interface. You do not need to install this option to use the product because all operations can be performed from the command line interface which is part of the base option.

The base option is a prerequisite to the graphical interface option. If you want the graphical interface to be displayed in a language different from US English, which is the default, then you should install the `gitext` option from the NLS package.

NLS Package Options

The NLS package contains all forms of documentation associated with TME 10 Software Distribution, Version 3.1.5 for AIX, in the national language it is requested in. This package is provided when you purchase any other package. It contains the following options:

Installation Requirements

Message   This file contains product messages and files used for routing messages to NetView/6000. The base option is a prerequisite.

Gitext   This file contains the text displayed on graphical interface windows and graphical interface online help. The graphical interface option is a prerequisite. If Gitext is not installed, the graphical interface is displayed in the default language (US English).

Man   This file contains man (manual) pages for the product, which enable information related to command line commands to be displayed online. The base option is a prerequisite.

| **PDF**   This option contains the TME 10 Software Distribution, Version 3.1.5 for
| AIX manuals in Adobe Acrobat's PDF format.

PS   This option contains the TME 10 Software Distribution, Version 3.1.5 for AIX manuals in printable PostScript format.

Installation Requirements

This section lists the installation requirements for TME 10 Software Distribution, Version 3.1.5 for AIX.

Hardware Requirements

TME 10 Software Distribution, Version 3.1.5 for AIX can be installed and used on RISC System/6000, POWERstations, POWERservers, and PowerPCs. The product can also run on symmetric multiprocessor (SMP) workstations. If you install the X/Motif graphical interface, you also need a graphics monitor and adapter.

To support SNA or TCP/IP communication, the RISC System/6000 workstation must be equipped with one of the following communication adapters:

- Ethernet
- Multiprotocol
- Portmaster
- Serial (SLIP)
- Token-Ring
- X.25 coprocessor

Disk Space and Memory Requirements

| The TME 10 Software Distribution for AIX server occupies approximately 36MB; the
| documentation in Postscript and PDF files will occupy another 75MB.

Software Requirements

The following sections describe other software needed to run the various TME 10 Software Distribution, Version 3.1.5 for AIX product options.

AIX Base Operating System (BOS)

AIX Version 3.2.5 Extended or higher must be installed on a workstation before installing any TME 10 Software Distribution, Version 3.1.5 for AIX options. Later versions must be backward compatible with Version 3.2.5

Extended. Version 4.3.3 is the latest version of the AIX operating system supported by TME 10 Software Distribution, Version 3.1.5 for AIX.

AIX BOS Network Facilities (BOSNET) TCP/IP Option

This facility must be installed on a workstation before installing the base option with the server option. More precisely, the `bosnet.tcpip` and the `bos.net.tcp.client` filesets must be installed.

Data link control drivers

You need suitable data link control drivers installed so that you can achieve remote communication. These drivers are supplied with the basic AIX operating system, as part of the BOSNET extension.

The drivers that you need depend upon the physical link that you are using. There are drivers for Token Ring and SDLC.

SNA Server/6000

You need SNA Server/6000 Version 2.1 installed if your configuration includes an SNA network. SNA Services Version 1.2 at fix level 366 or higher can also be used for the same purpose.

For AIX Version 4.3.3, Communications Server Version 5.0 should be used.

NetView/6000

You need NetView/6000 Version 2 or later, together with IBM AIX Service Point installed in the network if you intend to make use of the facility that routes SNA alerts from TME 10 Software Distribution, Version 3.1.5 for AIX to NetView/6000 and then to NetView DM on the mainframe. To use this facility you must also have the BOSNET `bosnet.snmpd` fileset installed on TME 10 Software Distribution, Version 3.1.5 for AIX.

AIXwindows

You need AIXwindows Version 1.2, Motif 1.2, and the appropriate font support if you intend to install the graphical interface. For Motif 1.2, install PTF U424846, PTF U435138, and any other prerequisite PTFs.

In addition, you must also have the following filesets installed:

- **AIX 3.2.5**
  - x11rte.obj
  - x11rte.motif1.2.obj.1.2.3

- **AIX 4.x**
  - x11.base.rte
  - x11.motif.lib
  - x11.motif.mwm

Communications Protocols

Communications protocols are required for server and client communication and for connecting a NetWare distribution server to an MVS or AIX focal point.

Protocols to Connect to Another Server

The following protocols are supported for an AIX Server communicating with another TME 10 Software Distribution Server.

Server-to-Server (STS) Protocols

Table 10. STS Protocols to Connect to Another Server

Platform	Communications Potocol
OS/2	APPC TCP/IP
NetWare	TCP/IP
Windows NT or Windows 2000	TCP/IP
AIX	APPC TCP/IP

SNA/DS Protocols: Both APPC and TCP/IP protocols can be used to connect to servers using SNA/DS.

Protocols to Connect to a Client

The following protocols are supported for an AIX server communicating with a TME 10 Software Distribution Client.

Table 11. Protocols to Connect to A Client

Platform	Communications Potocol
OS/2	APPC TCP/IP
NetWare	TCP/IP
Windows NT or Windows 2000	TCP/IP
Windows 95 or Windows 98	TCP/IP
Windows 3.11	TCP/IP
AIX	APPC TCP/IP

From a TME 10 Software Distribution for AIX server it is also possible to make an ASYNCHRONOUS connection with Windows 3.11 and OS/2 clients.

Protocols to connect to a focal point

For communication with an MVS focal point the APPC-TCP/IP protocol should be used.

Prerequisites on the Host System

To perform software distribution from an MVS system to TME 10 Software Distribution, the MVS system must have NetView DM for MVS Release 6.2 or 7 installed; however,

| if you wish to enable the host to delete pending requests at the server, it must be
| Release 7.

| To perform TME 10 Software Distribution from an AIX system to TME 10 Software
| Distribution, the AIX system must have TME 10 Software Distribution 3.1.4 for AIX, or
| later, installed.

Using TME 10 Software Distribution, Version 3.1.5 for AIX in Multi-Language Environments

Your network may include servers and clients on which the feature is installed in
different languages. For example, a server in San Francisco on which the feature is
installed in English may communicate with a server in Tokyo on which the feature is
installed in Japanese. The following sections describe how to work in these
environments.

National Languages in a Domain

The clients and the server in a domain should use the feature in the same language,
otherwise information stored on one workstation may not be legible on another. For
example, messages stored in the message log on the server workstation in Japanese
cannot be displayed correctly on a client running in English.

Using TME 10 Software Distribution for AIX in Multi-Language Environments

AIX Language Variable

The feature is available in the English language only. If your AIX operating system is not in English, this fact may cause some confusion, because prompts from TME 10 Software Distribution, Version 3.1.5 for AIX in English will not correspond to actual responses that you must enter in the language your AIX environment is installed in.

For example, if you are working with AIX in Italian, and receive the following message in English from TME 10 Software Distribution, Version 3.1.5 for AIX:

```
uncatalog flat.data.res  y/n?
```

you must respond in Italian using either:

```
s (si) or n (no)
```

To ensure full message capability, the AIX LANG environment variable should be set to the language appropriate to your system, not to C.

Entering Data

In product documentation, restrictions about the length of fields are made with respect to single byte character set (SBCS) languages. For double and multiple byte character set (DBCS and MBCS) languages, those restrictions are calculated in terms of bytes, not characters.

For example, a field that can have up to eight characters in an SBCS language is limited to eight bytes in DBCS and MBCS languages.

To avoid unpredictable results when exchanging data between workstations that use different character sets, it is suggested that you use alphabetic characters (lowercase and uppercase), digits, and the characters listed in Table 12 on page 71, unless otherwise stated.

Table 12. SBCS and MBCS Supported Characters			
Symbol	**Description**	**Symbol**	**Description**
blank	space	!	exclamation mark
?	question mark	#	number sign
$	dollar sign	%	percent
&	ampersand	'	apostrophe
(	left parenthesis	)	right parenthesis
*	asterisk	+	plus sign
,	comma	-	hyphen
.	period	/	slash
:	colon	;	semicolon
<	less than	=	equal sign
>	greater than	"	quotation mark
@	commercial at	[	left bracket
\	backslash	]	right bracket
^	circumflex	_	underscore
`	grave accent	{	left brace
|	vertical line	}	right brace
~	tilde		

Chapter 9. Installing TME 10 Software Distribution, Version 3.1.5 for AIX

TME 10 Software Distribution, Version 3.1.5 for AIX is distributed as a refresh that you
| can use to install the product from scratch or to upgrade from TME 10 Software
| Distribution, Version 3.1.x.

| It is also distributed as PTF U469697 *TME 10 Software Distribution update* which you
| install on top of TME 10 Software Distribution Version 3.1.X for AIX, where X is equal
| to or greater than 0. The PTF is provided in the English language only and can be
| installed on top of the English version only of the product. To install it, follow the
| instructions given in the Memo to Users provided with the PTF itself (called
| **FNDRMMST.TXT**, with equivalent versions in PostScript and PDF formats).

| This chapter describes how to install TME 10 Software Distribution, Version 3.1.5 for
| AIX. It can be used either from scratch, i.e. on a computer which does not have a
| previous version of TME 10 Software Distribution for AIX installed, or, if you already
| have installed TME 10 Software Distribution for AIX 3.1.x, to update the program files
| to the new version without changing your data or configuration.

Before installing the product you must:

1. Determine your environment.

2. Know the AIX name of the device from which the installp package is to be loaded.
   If the package is already on your hard disk, you must know the name of the
   directory where the installp package is stored. For example, a high-density tape is
   often accessed as /dev/rmt0.1, and packages are usually stored on the hard disk
   in the directory /usr/sys/inst.images.

3. Have root privileges.

The options of TME 10 Software Distribution can be installed separately; the base
option is a prerequisite for the other components. Use either of the following
installation procedures:

SMIT   The System Management Interface Tool (SMIT) provides a menu based
              interface for installing and removing options.

installp The AIX command installp can be used to install the options. This
              command is very flexible. If you are using this command, see:

              • "Installing TME 10 Software Distribution Using installp" on page 76.

              • The description of the command in the *Reference* manual.

Note: You can also install minor enhancements to TME 10 Software Distribution,
          Version 3.1.5 for AIX using the change control facilities of TME 10 Software
          Distribution, Version 3.1.5 for AIX itself. Treat TME 10 Software Distribution as
          you would any other software package, and install a change file.

Installing TME 10 Software Distribution Using SMIT

To install the TME 10 Software Distribution options using SMIT, perform the following steps:

1. Insert the TME 10 Software Distribution CD-ROM (number `LK3T-5088-00`)

2. Enter the following command:

   ```
   smitty install_selectable_all
   ```

   Note: If you start SMIT in a graphical interface, you cannot use the Tab key or PF keys described in this section. You must use either a mouse or *smitty* to perform the same operations.

3. Press F4 to see list of devices or directories from which the options are to be installed, or perform step 4.

4. In the `INPUT device / directory for software` field, type the input device or directory from which you are installing the package. You must use the full path name of the device or directory.

5. In the `SOFTWARE to install` field, press F4 to see a list of the options that can be installed, or perform step 6.

6. Select the names of the options to be installed. The options available depend on the TME 10 Software Distribution, Version 3.1.5 for AIX package you are installing. Packages and options are described in detail in "TME 10 Software Distribution, Version 3.1.5 for AIX Packages and Options" on page 63. The options can be from among the following:

Table 13 (Page 1 of 2). TME 10 Software Distribution Product Options and Their Prerequisites

Option Name	Descriptive Name	Prerequisites
netviewdm6000.base.obj	Base option	AIX Operating System 3.2 or higher
netviewdm6000.comms.obj	Communication option	Base SNA Services 1.2 or higher TCP/IP 3.2 or higher
netviewdm6000.server.obj	Server option	Base TCP/IP 3.2 or higher
netviewdm6000.gi.obj	Graphical interface	Base Gitext x11rte filesets
netviewdm6000.remoteadmin.obj	Remote administrator option	Base Communication
netviewdm6000.plan.obj	Plan feature	Base
netviewdm6000.tool.obj	Tool feature	Base
netviewdm6000.msg.en_US	Product messages in US English for the ISO 8859-1 code set	Base

Table 13 (Page 2 of 2). TME 10 Software Distribution Product Options and Their Prerequisites

Option Name	Descriptive Name	Prerequisites
netviewdm6000.gitext.en_US	Graphical interface text in US English for the ISO 8859-1 code set	Graphical interface
netviewdm6000.man.en_US	Man pages in US English for the ISO 8859-1 code set	Base
netviewdm6000.pdf.en_US	Adobe Acrobat PDF Format books in US English	None
netviewdm6000.ps.en_US	PostScript books in US English	None
netviewdm6000.msg.En_US	Product messages in US English for the IBM-850 code set	Base
netviewdm6000.gitext.En_US	Graphical interface text in US English for the IBM-850 code set	Graphical interface
netviewdm6000.man.En_US	Man pages in US English for the IBM-850 code set	Base

7. Set the following options on the Install/Update from All Available Software Packages menu:

```
 * INPUT device / directory for software

 * SOFTWARE to install                           .[No]

   PREVIEW only? (install operation will NOT occur)    no

   COMMIT software updates?                        yes

   SAVE replaced files?                            no

   AUTOMATICALLY install requisite software?       yes

   EXTEND file systems if space needed?            yes

   OVERWRITE same or newer versions?               no

   VERIFY install and check file sizes?            no

   DETAILED output?                                no

   Process multiple volumes?                       yes
```

Note: This is *one* example of how SMIT installation options can be set. Other settings are possible.

Reinstalling TME 10 Software Distribution

When the installation is complete, **OK** is displayed in the command field. TME 10 Software Distribution has been installed successfully.

Installing TME 10 Software Distribution Using installp

To install the TME 10 Software Distribution options using installp, enter the following command:

```
installp -qa -d <device> -X <option>
```

where:

<device>   Is the name of the device, or local path, on which the installp image is stored.

<option>   Identifies the option to be installed. Enter this parameter once for each option that you want to install. TME 10 Software Distribution product options are listed "Installing TME 10 Software Distribution Using SMIT" on page 74.

Journaled File System

The TME 10 Software Distribution install operation creates a journaled file system named `/usr/lpp/netviewdm`. The default size of the journaled file system is 24MB. If this file system already exists then it is used when an installation is performed.

To install TME 10 Software Distribution on an external disk, you must create the journaled file system `/usr/lpp/netviewdm` on the volume group related to that external disk.

Reinstalling TME 10 Software Distribution

You can reinstall any of the TME 10 Software Distribution product options. You do this, for example, if you alter your existing installation by deleting an executable file.

Existing configuration files are not overwritten when TME 10 Software Distribution is reinstalled. The following sections describe the reinstall procedures using either SMIT or installp.

Reinstalling TME 10 Software Distribution Using SMIT

To reinstall an option using SMIT you need root privileges. From the command line, enter:

```
smit
```

Proceed with the installation on the Install Software Without Updates screen.

Note: If your current product has been installed with the `Commit software` option on the Install Software With Updates menu set to Yes, then you must set the `Overwrite existing version` option to Yes before reinstalling the installp package. If your current product has been installed with the `Commit software` option set to No, you must remove it before reinstallation, or commit it.

Reinstalling TME 10 Software Distribution Using installp

To reinstall the TME 10 Software Distribution options, enter the following command for each of the options to be installed:

```
installp -a -F -X -d <device> <option>
```

The parameters are the same as those for the installation procedure, except that -F is used for forcing the installation on the existing code of the same level, when the existing level is in the committed state.

Saving a Configuration

Take note of the following when using the reinstallation procedure:

- The save and restore operations can only be performed on products that have the same level.

- Reinstallation of the server option does not affect the existing configuration.

- Reinstallation of the communication option overwrites all SNA communication configuration files.

- Reinstallation of the base option overwrites all configuration files (base, server, and communication) and deletes all log files in `/usr/lpp/netviewdm`.

For these reasons you should save your configuration before reinstallation. To do this, copy the configuration to a safe location beforehand, and restore it afterwards by performing the following steps. You must have root privileges to perform these tasks.

1. Find a volume where you can save the configuration files. Use, for example, your home directory (for example, `usr/john`).

2. Copy the TME 10 Software Distribution database to your directory using the command:

   ```
   cp -R /usr/lpp/netviewdm/db /usr/john/config
   ```

3. Reinstall your options as described in "Reinstalling TME 10 Software Distribution" on page 76.

4. Restore the TME 10 Software Distribution database using the command:

   ```
   cp -R /usr/john/config/* /usr/lpp/netviewdm/db
   ```

Removing TME 10 Software Distribution Options

If you ever want to remove TME 10 Software Distribution options, keep the following points in mind:

- Stop TME 10 Software Distribution by issuing the command:

   ```
   nvdm stop -x -k
   ```

   before removing the product.

Removing TME 10 Software Distribution Options

- Under AIX Version 3.2.5, if the options are installed with the `Commit software` parameter set to Yes, they cannot be removed. You must install them again as Applied to be able to remove them.

- The remove operation will fail if you select *all* options at once and set the `Automatically remove dependent software` parameter to No.

- To remove all software options, set `Automatically remove dependent software` to Yes, and select the `netviewdm.base` option only.

- It the status of the product is COMMITTED, to remove the product the `OVERWRITE existing version` flag must be set to Yes.

Chapter 10. Editing the Base Configuration File

The first configuration activities you perform on a newly installed TME 10 Software Distribution, Version 3.1.5 for AIX system are at the server. First, you must define it as a target. You then work from it to perform the configuration activities required to set up your network.

Basic server configuration is performed during the installation process itself and stored in the base configuration file. This section describes the format of this file and the values that you can enter in its fields.

Base Configuration File Parameters

The base configuration file, `/usr/lpp/netviewdm/db/nvdm.cfg`, contains the system parameters for controlling TME 10 Software Distribution, Version 3.1.5 for AIX. The `nvdm.cfg` file generated at installation does not contain all possible parameters, only a default subset of them. You can add or remove parameters and their values by editing the file using any text editor.

The file is stored in a fixed text format. Each line starts with one of the keywords described in Table 14 on page 80. Keywords must be entered in uppercase and ended with a colon (:). Each keyword can be used only once with the exception of the **SERVER** keyword, which can be used up to five times. The order of keywords in the file is not important, and blank and comment lines can be included. Comment lines begin with a hash character (#).

To make modifications to the file operative, you must stop and then restart the product using the commands:

```
nvdm stop -x -k
nvdm start
```

Note that if incorrect entries are entered in the base configuration file, the product may not start.

Editing the Base Configuration File - Parameters

Table 14 (Page 1 of 10). Base Configuration File Parameters	
Keyword	**Description**
API TRACE FILE SIZE	The maximum size of the TME 10 Software Distribution, Version 3.1.5 for AIX API trace files in bytes. When the trace file is full it is automatically backed up, and a new trace file is started.
	You can change this value but it should not normally be necessary. A large value uses more disk space. A small value degrades performance very slightly, because the trace is backed up more often.
	Refer to the *Messages and Codes* for information about the API trace file.
AUTHORIZE	This keyword authorizes change control operations on change files and the execution of data files at a workstation. It is applicable only to:
	• Operations that are NOT performed by FNDADMN users (because the FNDADMN user profile allows them to perform these operations regardless).
	• Operations performed on change files that do NOT already have a change control status.
	Specify either:
	NONE No target is authorized to install a change file or to execute a data file unless it is authorized via the `auth` command.
	ALL Any target is authorized to install a change file or to execute a data file unless it is not authorized via the `unauth` command.
	The default is NONE.
	Set the AUTHORIZE keyword as soon as you have installed TME 10 Software Distribution, Version 3.1.5 for AIX. When you change the value of this keyword, all the authorized targets become unauthorized and vice versa.
AUTOMATIC TARGET INVENTORY	This keyword causes clients, when they are added via command line or via GUI, to automatically perform a hardware and software inventory and store the results in the server database.
	Valid values are:
	YES The target inventory is performed when the target is added via command line or via GUI
	NO The target inventory is performed only when an automatic registration occurs. See the AUTOMATIC TARGET REGISTRATION keyword.
	The default is NO.

Table 14 (Page 2 of 10). Base Configuration File Parameters	
Keyword	**Description**
AUTOMATIC TARGET REGISTRATION	This keyword enables clients to automatically configure themselves, or to autoregister, in the server database as one of the server's local targets. The configuration is performed the first time a client connects to the server, if the client is not already configured. The client is also registered at all servers at higher levels in the network hierarchy, if AUTOMATIC TARGET REGISTRATION is set to Yes in their base configuration files as well.
	Valid values are:
	YES Enables targets to be registered at the server.
	NO Does not enable targets to be registered at the server.
	The default is NO.
	You must specify the TARGET ADDRESS and TARGET MODE keywords in the client's base configuration file if automatic registration is to be performed. In addition, set the CONFIGURATION keyword in the client base configuration file to CLIENT.
	When an automatic target registration is performed an inventory is automatically scheduled.
AUTOMATIC TARGET INFO UPDATE	This keyword specifies whether or not the focal point target is to update its database whenever a target configuration is updated at its server. Any information updated for the target is updated at the focal point target, except for the target name, target address, domain address, and server name.
	Valid values are:
	YES The focal point target updates its database with the target's new information.
	NO The focal point target does not update its database with the target's new information.
	The default is NO.
AUTOMATIC TARGET DELETION	This keyword specifies whether or not the focal point target is to update its database whenever a target is removed from its server's domain.
	Valid values are:
	YES The focal point target updates its database with the removed target's information.
	NO The focal point target does not update its database with the removed target's information.
	The default is NO.

Editing the Base Configuration File - Parameters

Table 14 (Page 3 of 10). Base Configuration File Parameters	
Keyword	**Description**
BACKUP AREA	The name of the directory that change files are stored in when they are installed as removable on a workstation. The default is /usr/lpp/netviewdm/backup. If you change the default path, you must copy all files into the new path, otherwise change control history information is lost. If the default path for the backup area is changed, it is not removed automatically if the product is uninstalled.

CONFIGURATION	The configuration of your TME 10 Software Distribution, Version 3.1.5 for AIX node. The name is set up during installation and cannot be changed. It can be one of the following:
	
	INSTALL
	    Base option installed. No functions can be performed with this system.
	
	BASE
	    Base and communication options installed.
	
	SERVER_NO_COMMS
	    Base and server options installed.
	
	SERVER_WITH_COMMS
	    Server and communication options installed.
	
	REMOTE_ADMIN_BASE
	    Base and remote administrator options installed.
	
	REMOTE_ADMIN_SERVER
	    Server and remote administrator options installed.
	
	The value of the field in the server base configuration file can be displayed using the **lsbs** command. (See *TME 10 Software Distribution, Version 3.1.5 for AIX Reference.*)

CONNECTION WINDOW DURATION	The value you specify on this parameter becomes the default value for the **d** option on the **connect** command. It represents the amount of time, in minutes, that the connection window from the server to the client is to remain open. The default value is 60 minutes.
DACA IDLE TIME	The time in seconds after which an idle DACA connection is considered failed. Enter a value from 0 to 32767. The default is 300.

Table 14 (Page 4 of 10). Base Configuration File Parameters	
Keyword	**Description**
DACA RETRY TIME	The time in seconds before a failed DACA connection is retried.
	Enter a value from 0 to 32767. The default is 600.
DIAL RETRY PERIOD	The time period to wait before attempting the next MAX DIAL RETRIES attempts to dial the client.
MAX DIAL RETRIES	The maximum number of dial attempts made for each DIAL RETRY PERIOD.
EXTEND FILE SYSTEM ON STORE	This keyword specifies whether the product is enabled to extend the files system at the destination workstation if there is not enough disk space to store a file during a send or a retrieve operation.
	Valid values are:
	YES If there is not enough space on the workstation, TME 10 Software Distribution, Version 3.1.5 for AIX extends the destination file system.
	NO The file system cannot be extended at the destination workstation.
	The default is NO.
FILE SYSTEM	The name of the file system that the operating system supports. Based on this keyword, TME 10 Software Distribution chooses the local name when a request provides multiple local names.
	The default is one of the following numeric values, depending on the operating system:
	1 AIX **2** OS/2, Windows 3.11, DOS **3** NetWare **5** Windows 2000, Windows NT, Windows 98, Windows 95
FORWARD OFFLINE LOGS	This keyword determines whether log messages issued while a mobile client is disconnected are to be forwarded to the server. You can specify either YES or N0.
INVENTORY PROGRAM	The name (with its complete path) of the program invoked when the **inv** command is issued. It creates the fndswinv file.

Editing the Base Configuration File - Parameters

Keyword	Description
Table 14 (Page 5 of 10). Base Configuration File Parameters	
Keyword	**Description**
LAN AUTHORIZATION	Valid entries are: **0** LAN address authorization is not required. **1** TME 10 Software Distribution, Version 3.1.5 for AIX validates LAN addresses on all LAN messages received by the server. If you specify LAN address authorization, you must supply the LAN address of each local target. The LAN authorization level can be set for individual work sessions using the **updbs** command at the command line interface or using the graphical interface. However, if you use one of the interfaces to change the value, it is not changed in the base configuration file, but is only modified for your current work session. This LAN Authorization level has meaning in the server base configuration file only.
LOG FILE SIZE	The maximum size of the TME 10 Software Distribution, Version 3.1.5 for AIX message log file in bytes. When the log file is full it is automatically backed up, and a new log file is started. You can change this value but it should not normally be necessary. A large value uses more disk space. A small value degrades performance slightly, because the log is backed up more often. Refer to the *Messages and Codes* for information about the message log.
MACHINE TYPE	The operating system in use on your TME 10 Software Distribution, Version 3.1.5 for AIX node. The name is set up during installation and cannot be changed.
MAX ATTEMPTS	The maximum number of failed attempts to connect to the server before the system temporarily revokes the access to the server. The default is *5.*
MAX CONNECTIONS	The maximum number of simultaneously connected local targets allowed, where 'connected' means that the target is in the process of performing a D&CC request. You can specify up to 100 connections. The default is 50.
MAX LOCAL TARGETS	The total number of local targets that can be configured for a server. The maximum number is 2000. The default is 512. If the single-node package is installed, this parameter must be set to 1.

Table 14 (Page 6 of 10). Base Configuration File Parameters

Keyword	Description
MAX REQUESTS	The maximum number of requests that can be active simultaneously.
	Enter a value from 1 to 65536. The default is 8192.
MAX SERVER CONNS	The maximum number of connections to remote server DACAs that be open simultaneously. These STS connections receive requests from remote servers to perform operations. The MAX STS keyword, on the other hand, defines those STS connections that initiate operations to remote servers.
	Enter a value from 0 to 256. The default is 10.
MAX SERVER TARGETS	The total number of non-adjacent server targets that can be configured for a server. The total includes remote servers and single node targets. The MAX SNADS ROUTES and MAX STS ROUTES keywords are used to specify the number of these targets connected across either SNA/DS or STS connections.
	The maximum number is 2048. The default is 10.
MAX SNA CONNS	The maximum number of connections to remote server DACAs that be open simultaneously. These SNA connections receive requests from remote servers to perform operations.
	Enter a value from 0 to 500. The default is 100.
MAX SNADS ROUTES	The maximum number of SNA/DS connections that can be defined.
	The maximum number is 800. The default is 20.
MAX STS	The maximum number of STS processes that can be simultaneously handling connections to remote servers. These STS connections send requests for remote servers to perform operations. The MAX SERVER CONNS keyword, on the other hand, defines those STS connections that receive requests *from* remote servers.
	Enter a value from 0 to 256. The default is 10.
MAX STS ROUTES	The maximum number of STS connections that can be defined.
	The maximum number is 1600 minus the number of SNA/DS routes that have been defined (MAX SNA/DS ROUTES). The default is 30.

Editing the Base Configuration File - Parameters

Table 14 (Page 7 of 10). Base Configuration File Parameters	
Keyword	**Description**
MAX TARGETS	The total number of local and/or remote targets that can be configured. The maximum number of targets is 40000.
	This value must be greater than the sum of MAX LOCAL TARGETS and MAX SERVER TARGETS.
	The formula MAX TARGETS minus MAX LOCAL TARGETS minus MAX SERVER targets determines the number of remote targets that can be configured at a server, for which there is no keyword in the base configuration file.
	The default for MAX TARGETS is 600.
MAX USER INTERFACES	The maximum number of simultaneously connected user interfaces. The default is 20.
MESSAGE LOG LEVEL	This field defines the log level that should be used by clients before they establish a connection to the server and discover the level configured for them there.
	Three log levels are available:
	M Minimal **N** Normal **D** Diagnostic
OS USER	The name of a user of the operating system. This keyword is checked whenever a send, retrieve, or delete operation has been requested for a file, to make sure that the OS USER is authorized to perform the operation. If * (asterisk) is specified for OS USER, then this check is performed on the originator of the request.
	The default is root.
PLAN FEATURE	Specifies whether the users logged on to this server are allowed to manage plans or not. Valid values are:
	YES Users can work with plans.
	NO Users cannot work with plans.
	The default is NO.

Table 14 (Page 8 of 10). Base Configuration File Parameters

Keyword	Description

PORT	This Keyword defines the port and the modem initialization string used to communicate with the server. It is used to both dial to the server, and receive communication from the server. It can be specified only once, and has the following format:
	
	`PORT:    <port name>`
	`    INIT: <init string>`
	
	Where:
	
	`<PORT>`
	    Is the name of the communication port
	
	`<INIT>`
	    Contains the Hayes AT command set string to send to the modem to initialize it
	
	For example, specify:
	
	`PORT:    COM3`
	`    INIT: AT`

REPOSITORY	The name of the repository that TME 10 Software Distribution, Version 3.1.5 for AIX stores its objects in. The default repository is `/usr/lpp/netviewdm/repos`, unless explicitly overridden. If the default repository path is changed, the repository is not removed automatically if the product is uninstalled.
SERVER	The name of the server that serves your TME 10 Software Distribution, Version 3.1.5 for AIX node. It is set up automatically on the server during installation. The server name is the text string that identifies the workstation that supports the server. If you do not know this name, you can find it by looking at the server base configuration file. If you want to attach a target to other servers, so that you are able to administer and configure them, you must name those servers here. This keyword can be used up to five times. The first instance defines the server that provides the server function for the client being defined in this table, the other instances define servers that can be administered from this client.

Editing the Base Configuration File - Parameters

Table 14 (Page 9 of 10). Base Configuration File Parameters	
Keyword	**Description**
SERVICE AREA	The name of the directory that files and information are stored in when change files that require activation are installed. The directory should belong to a local file system mounted during startup. The default is `/usr/lpp/netviewdm/service`. If you change the default path, you must copy all files into the new path, otherwise change control history information is lost. If the default path for the service area is changed, it is not removed automatically if the product is uninstalled.
STS IDLE TIME	The time in seconds after which an idle STS connection is considered to have failed.
	Enter a value from 0 to 32767. The default is 60.
STS RETRY TIME	The time in seconds before a failed STS connection is retried by the Scheduler.
	Enter a value from 0 to 32767. The default is 300.
TARGET PASSWORD AUTHENTICATION	This keyword activates the target authentication feature.
	Valid values are:
	• YES
	• NO
	The default is NO.
TCP/IP PORT	The TCP/IP port number of your TME 10 Software Distribution, Version 3.1.5 for AIX server. It must be a decimal number. This is set up automatically during the installation process and has the value 729. You need to change this field if the TCP/IP port number of your server is changed. Otherwise, the field should not be altered.
TRACE FILE SIZE	The maximum size of the TME 10 Software Distribution, Version 3.1.5 for AIX internal trace files in bytes. Two trace files are used and when one is full, tracing automatically switches to the other. Internal tracing is only activated for diagnostic purposes.
	You can change this value but it should not normally be necessary. A large value uses more disk space. A small value degrades performance very slightly, because the log is backed up more often.
	Refer to the *Messages and Codes* for information about the TME 10 Software Distribution, Version 3.1.5 for AIX internal tracing.

Table 14 (Page 10 of 10). Base Configuration File Parameters	
Keyword	**Description**
WORK AREA	The name of the directory that temporary work files are created and used in, and then deleted from. The default is /usr/lpp/netviewdm/work. If the default path for the work area is changed, it is not removed automatically if the product is uninstalled.
WORKSTATION NAME	The name of the workstation that is running your TME 10 Software Distribution, Version 3.1.5 for AIX node. The name is set up during installation and cannot be changed.

The following is an example of the server base configuration file.

```
# BASE CONFIGURATION FILE
#
# This file should be stored as /usr/lpp/netviewdm/db/nvdm.cfg

WORKSTATION NAME:            nvdmsvr
MESSAGE LOG LEVEL:           N
LAN AUTHORIZATION:           1
CONFIGURATION:              SERVER_WITH_COMMS
MACHINE TYPE:               AIX
LOG FILE SIZE:              50000
TRACE FILE SIZE:            1000000
API TRACE FILE SIZE:        500000
TCP/IP PORT:                729
MAX TARGETS:                600
MAX CONNECTIONS:            50
MAX USER INTERFACES:        20
SERVER:                     nvdmsvr
SERVER:                     nvdmbak
REPOSITORY:                 /usr/lpp/netviewdm/repos
SERVICE AREA:               /usr/lpp/netviewdm/service
BACKUP AREA:                /usr/lpp/netviewdm/backup
WORK AREA:                  /usr/lpp/netviewdm/work
PLAN FEATURE:                Y
```

Figure 7. Example of a server Base Configuration file

Changing TCP/IP Ports

TME 10 Software Distribution, Version 3.1.5 for AIX servers listen at a TCP/IP port for communication from clients and from other servers. The following port numbers have been assigned to the product by Internet:

729   Local server to client communication

Configuring Files at the Server

730, 731 Remote server to server communication

If in the unlikely event one of the above ports is being used by another application, and you decide to change TME 10 Software Distribution, Version 3.1.5 for AIX ports, the following applies:

- The local communication port, 729, is stored in the base configuration file. The port number must be changed in the base configuration file of the server and the base configuration files of all its clients.

- The remote communication ports, 730 and 731, are stored in `/etc/services`. They have to be modified on all connected servers.

If NFS occupies the port numbers reserved for TME 10 Software Distribution, Version 3.1.5 for AIX, issue the following commands:

```
nvdm stop -k -x
stopsrc -g nfs
nvdm start
startsrc -g nfs
```

Changing the Server Name

When TME 10 Software Distribution, Version 3.1.5 for AIX is installed, a default server name and server short name, `SERVER.SERVER` is automatically defined. You must change these values to the names used in your network.

Perform the following steps:

1. Log on as *root*.

2. Stop all activity on the server and on clients by entering the following command on the workstations:

   ```
   nvdm stop -x
   ```

3. Edit the `WORKSTATION NAME` and `SERVER` keywords in the `/usr/lpp/netviewdm/db/nvdm.cfg` file to reflect the server's new name. See "Base Configuration File Parameters" on page 79.

4. Edit the name of the server in `nvdm.cfg` on each client to reflect the new name of the server.

5. Add the new TCP/IP name to the TCP/IP configuration on each of the clients. For AIX clients, this involves editing the `/etc/hosts` file.

6. Rename the server target using the `nvdm rentg` command at the command line interface (see the *Reference*).

7. Erase all of the configuration files located in the `/usr/lpp/netviewdm/uicfg` directory.

8. Rename the physical node and add the new TCP/IP name to the TCP/IP configuration on the server using standard AIX facilities.

9. Start the server and the clients by entering the following command at the workstations:

   ```
   nvdm start
   ```

Setting the LAN Authorization Level

You can change the level of checking that is applied to LAN messages received from clients.

Messages received by the server identify their origin using TCP/IP name and LAN address information. The messages also carry the burned-in LAN address for the LAN adapter on the client. You can ensure that the server checks this burned-in address with a value configured at the server for the client. This is an extra security measure that helps to prevent unauthorized use of the LAN.

Use the graphical interface to set the level of LAN adapter checking applied by TME 10 Software Distribution, Version 3.1.5 for AIX as follows:

1. Access the main Catalog window of the graphical interface.
2. Choose **System** from the menu bar.
3. Select **Validate LAN** from the menu.

A cascade menu is displayed showing **On** and **Off**. The current setting cannot be selected. To change the setting, select the entry that is available. Cancel the menu to retain the current setting.

You can also set the LAN authorization level for the current session using the **updbs** command from the command line interface (see *TME 10 Software Distribution, Version 3.1.5 for AIX Reference*).

Chapter 11. Defining Users and Targets

This chapter provides the information needed to define users and targets in a network. You can find a complete description of how to define them using the command line interface in the online Command Reference for the product.

Defining Users

Users are defined at a database at the TME 10 Software Distribution server. A user definition includes the user's name and password, a description of the user, and security mechanisms that you can use to prevent unauthorized use of product functions and safeguard the privacy of sensitive data. They are:

- The definition of user profiles that enable the user to perform TME 10 Software Distribution, Version 3.1.5 for AIX functions. The system administrator can define profiles to be used by groups of users, or individual profiles for users. Three default authorization profiles are installed with the product; FNDADMN for system administrators, FNDBLD for change file builders, and FNDUSER for users.

- The definition of target access keys (TAK) that enable a user to work with specific targets.

- The definition of data access keys (DAK) that enable a user to work with specific objects, or data.

- The definition of the targets the user is authorized to login at

Defining Targets

In TME 10 Software Distribution, Version 3.1.5 for AIX, the term *target* refers to all workstations in a network to which change control and distribution activities are directed. The following table contains a description of the parameters you use to define targets in a network.

Table 15 (Page 1 of 7). Target Definition Parameters	
Parameter	**Description**
Name	Each target in a network has a unique name. When you communicate with a target, you use that unique name to identify it. A target name can contain up to 63 characters, excluding *, ?, and \.
Description	Text that describes the target. Up to 59 characters can be included in a description.

Defining Targets

Table 15 (Page 2 of 7). Target Definition Parameters	
Parameter	**Description**
Change management	This parameter defines a target's *mode*. Targets can be configured in one of the following modes: **Push mode** Change control operations on a push mode target are controlled by the administrator at a TME 10 Software Distribution server or from a focal or manager target. **Pull mode** Change control operations on a pull mode target are controlled by the target's user. **Manager** A target configured in manager mode can perform change control operations on local targets and on remote targets. The remote administrator option must be installed on managers. Only targets whose type is server or single can be configured in manager mode. More than one target can be defined as a manager in a network. **Focal** A target configured in focal mode is a focal point for targets that define it as such. This means that the defining target routes all change control operation reports (both local and remote) to it. A focal target also acts as a manager. Only targets whose type is server or single be configured in focal mode.

Table 15 (Page 3 of 7). Target Definition Parameters	
Parameter	**Description**
Target address	The address you assign to a target must be unique in its domain, and can be up to 8 alphanumeric characters long. For remote communication, the address of each target must be unique across the entire network. A complete address is composed of two parts, one of which is the target address you define here. The terminology used for the two parts of the address is different depending on the platform it runs on:
	• For TME 10 Software Distribution, Version 3.1.5 for AIX targets the terminology is the same as that used here. A complete target address is expressed as:
	`<server name>.<target address>`
	If a target is a TME 10 Software Distribution server, both its `<server name>` and its `<target address>` must be the same.
	In SNA environments, these terms are also referred to as:
	`<routing group name>.<routing element name>` ► `(RGN.REN)`
	• For NetView DM for MVS and NetView DM/2 targets, the complete name is composed of:
	`<Network ID>.<LU name>`
	Thus the `<target address>` entered in this field must correspond to a target's `LU name`.
	In most organizations, target names are allocated from a central authority (at the host processor) because this is the name to which change control instructions are sent. In a large network, it is often impossible to make this field meaningful. You may need to resort to the use of digits to produce unique names. If you are adding a remote target, its name must already be defined at the remote site.
	The address of a local target forms part of the global file name of files that are cataloged automatically for a target.

Defining Targets

Table 15 (Page 4 of 7). Target Definition Parameters	
Parameter	**Description**
Target type	A target must be one of the following *types*:
	Client
	Targets connected to a TME 10 Software Distribution server which have a TME 10 Software Distribution Client program installed.
	Clients can be either local or remote targets. Those in the same CC domain as their TME 10 Software Distribution server are referred to as *local* targets.
	Clients that belong to a different TME 10 Software Distribution domain are referred to as *remote* targets. Remote targets must be configured at all TME 10 Software Distribution servers they are to exchange files or change control requests with. Change control operations can be performed on remote targets only if the remote administration product option is installed on your system.
	Clients can be defined in push mode or pull mode.
	Clients can also be configured, or registered automatically, the first time they connect to a server. See "Automatic Client Registration" on page 99.
	Server
	Targets running TME 10 Software Distribution, Version 3.1.5 for AIX and which have the server option installed. TME 10 Software Distribution servers can perform change control and distribution operations on TME 10 Software Distribution clients in their software distribution domain. If they have the remote administrator product option installed, they can perform operations on remote targets.
	Server targets can be configured in push, pull, manager or focal mode.

Table 15 (Page 5 of 7). Target Definition Parameters	
Parameter	**Description**
Target type (cont.)	**Single** Targets running TME 10 Software Distribution, Version 3.1.5 for AIX configured as a base system. Single-node targets can be used as preparation sites for software or as focal points to receive reports of change control operations. They can be configured in pull, push, manager or focal mode. Single-node targets can be accessed over TCP/IP and SNA/DS (APPC) networks. Change control operations can be performed on them if the remote administration option is installed on your system. Use this type also to define a NetView DM for MVS target. **User Interface only (UI only) target** Targets that can be used to run the TME 10 Software Distribution, Version 3.1.5 for AIX user interfaces. This type of target is useful when you have an environment in which more than one TME 10 Software Distribution server exists. It allows an administrator (a user belonging to the FNDADMN user group) to access all TME 10 Software Distribution servers from the same target either to perform administrative tasks or to schedule distributions to targets. A workstation configured in this manner is used to initiate change control on *other* targets, or to request distributions to and from the TME 10 Software Distribution server. User interface only targets cannot receive change control instructions from the same TME 10 Software Distribution server for which they are defined. You cannot define change control modes for UI only targets.

Defining Targets

Table 15 (Page 6 of 7). Target Definition Parameters	
Parameter	**Description**
Target OS	The type of operating system installed on the target. It can be one of the following: AIX DOS HP_UX MAC NCR NETWARE OS/2 SCO SINIX SOLARIS SUNOS WINDOWS (for Windows 3.11) WIN95 (for Windows 95) WIN98 (for Windows 98) WINDOWS_NT WIN2K (for Windows 2000)
Access key	The key that allows users to access the target. A user can access a target only if he or she has been assigned the same access key as the one assigned to the target. Target access keys are defined by the system administrator.
LAN address	The target's burned-in LAN address. This parameter must be defined if the Validate LAN function is set to on. When it is on, if the target attempts to communicate with a TME 10 Software Distribution server, a check is performed to verify whether the LAN address stored in the target's database matches its actual address. The Validate LAN function can be set at either a TME 10 Software Distribution server or a target workstation. To change the setting from a target, the LAN address must already be defined in the database of the target. If you are adding a UI target only, a target's LAN address is automatically captured at the target itself, and stored at the TME 10 Software Distribution server. Leave this field blank, unless you need to change the automatically captured address.
Server name	For a client target, the name of the TME 10 Software Distribution server to which the target is linked.

Table 15 (Page 7 of 7). Target Definition Parameters	
Parameter	**Description**
Domain address	This parameter is required for remote targets. The ID can be up to 8 characters long. The only valid characters are uppercase alphabetics and numerics. The value you define depends on the type of target you are defining: **TME 10 Software Distribution, Version 3.1.5 for AIX Server or Single Node** Specify the target address specified for the target. **Client** Specify the server name of the TME 10 Software Distribution server to which the client is connected. **Non TME 10 Software Distribution, Version 3.1.5 for AIX target** If the target you are defining is *not* a TME 10 Software Distribution, Version 3.1.5 for AIX target, specify, for NetView DM/2 and NetView DM for MVS nodes, the Network ID of the remote network to which this target belongs. This value corresponds to the routing group name (RGN) part of the SNA/DS address of the target. Ask the administrator at the host processor what the value is.
Password	The password used to access the target. It can have from 6 to 8 characters.

Automatic Client Registration

System administrators do not have to configure all client targets in a network individually at a server. They can be configured automatically, or autoregistered, the first time a client target connects to a server. For autoregistration to take place, the AUTOMATIC TARGET REGISTRATION keyword in a server's base configuration file must be set to YES (see Table 14 on page 80), and the TARGET ADDRESS and TARGET MODE keywords must be specified in the client base configuration file (see the *Installation and Configuration* manual for the TME 10 Software Distribution Clients).

When a client is automatically configured, its address and mode are inserted in the server database. Any other parameters for the client target must be specified manually using the graphical interface or the command line interface.

Automatic configuration is performed at all remote servers connected in a linear hierarchy to the first server that registers the target, as long as AUTOMATIC TARGET REGISTRATION is set to YES in each server's database. However, automatic target registration information is not routed from TME 10 Software Distribution, Version 3.1.5 for AIX servers to NetView DM/2 servers or NetView DM for MVS focal points.

Chapter 12. Configuring STS Remote Connection Files

This chapter describes how to configure the connection files required to define an STS network using either APPC or TCP/IP connections. You need to edit this configuration information when you define or update STS connections to remote nodes.

Editing STS Connection Configuration Files

The configuration information for STS connections is held in three different text files, which can be edited using a text editor.

The files are owned by `root` and can be accessed by the administrator, the same user who can make configuration changes to TME 10 Software Distribution, Version 3.1.5 for AIX using the graphical interface.

The files that can be edited are:

STS configuration file

This file contains the system parameters for controlling TME 10 Software Distribution, Version 3.1.5 for AIX's use of the STS network implemented across both APPC and TCP/IP connections.

STS connection configuration files

This file contains the details of an APPC or a TCP/IP connection to an adjacent node.

Routing table

This file instructs the server which APPC or TCP/IP connection should be used use when distributions are sent to remote targets.

If you edit one of these files while TME 10 Software Distribution, Version 3.1.5 for AIX is running, you must use the **rld** command to reload configuration changes. See *TME 10 Software Distribution Reference* for more information about this command.

STS Configuration File

The STS configuration file, `/usr/lpp/netviewdm/db/snadscfg`, contains a parameter for controlling the use that TME 10 Software Distribution, Version 3.1.5 for AIX makes of the STS network, ORIGIN HOP COUNT.

There can be only one configuration file at a server. Your network may include both STS and SNA/DS connections. An SNA/DS connection file can contain other keywords in addition to ORIGIN HOP COUNT (see "SNA/DS Configuration File" on page 111). If you have a mixed network, specify all the keywords in a single file.

You only need to edit this file when performing system tuning, and access is usually restricted to the administrator.

STS Connection Configuration File

The file is stored with a fixed text format. Enter the keyword once, in uppercase characters, and end it with a colon (:). Blank and comment lines can be included. Comment lines begin with a hash character (#).

Table 16. STS Configuration File Parameters	
Keyword	**Description**
ORIGIN HOP COUNT	The default hop count to be used for distributions originated by TME 10 Software Distribution, Version 3.1.5 for AIX when the hop count is omitted or is specified as zero in the routing table. (See "Editing the STS Routing Table" on page 104.) An appropriate value for this field is 5.

The following is an example of the STS Configuration File.

```
# STS CONFIGURATION FILE
#

# This file should be stored as /usr/lpp/netviewdm/db/snadscfg

ORIGIN HOP COUNT:                 10
```

Figure 8. Example of an STS Configuration file

STS Connection Configuration File

STS connection configuration files define the details of APPC or TCP/IP connections to adjacent nodes. Each file is given the name of the connection it defines. Up to 800 connection configuration files can be defined, while up to 100 connections can be active simultaneously.

The files are stored in the directory `/usr/lpp/netviewdm/db/snadscon`. Two sample files are provided for STS connections:

- `connstsappc`, for APPC connections
- `connststcp`, for TCP/IP connections

Access is usually restricted to the administrator. Sample connection configuration files are provided in the same directory.

Each file is stored with a fixed text format. Each line starts with one of the keywords described in Table 17 on page 103. Enter each keyword in uppercase and end it with a colon (:).

Each keyword can be used only once. The order of the keywords is not important, and blank and comment lines can be included. Comment lines begin with a hash character (#).

Table 17. STS Connection Configuration File Parameters

Keyword	Description
PROTOCOL	Specify either:
	APPC The APPC protocol is used. This is the default. Use this value to specify the APPN protocol as well.
	TCP/IP The TCP/IP protocol is used.
TYPE	Specify STS.
REMOTE SERVER NAME	The name specified for the remote server when it was defined with the graphical interface add target operation or the **addtg** command. It is the name of the target inserted in the product database.

The following is an example of an STS connection configuration file for the APPC connection protocol. This sample file, called CONNSTSAPPC, is installed during product installation.

```
# SNA/DS CONNECTION CONFIGURATION FILE FOR CONNECTION connstsappc
#
# This connection is used to handle transmissions between
# TME 10 Software Distribution, Version 3.1.5 for AIX servers using APPC across STS.
#
# This file should be stored as /usr/lpp/netviewdm/db/snadscon/connstsappc

PROTOCOL:                       APPC
TYPE:                           STS
REMOTE SERVER NAME:             sts_server
```

Figure 9. Example of an STS Connection Configuration file for APPC

The following is an example of an STS connection configuration file for the TCP/IP connection protocol. This sample file, called CONNSTSTCP, is installed during product installation.

```
# SNA/DS CONNECTION CONFIGURATION FILE FOR CONNECTION connststcp
#
# This connection is used to handle transmissions between
# TME 10 Software Distribution, Version 3.1.5 for AIX servers using STS across TCP/IP.
#
# This file should be stored as /usr/lpp/netviewdm/db/snadscon/connststcp

PROTOCOL:                       TCP/IP
TYPE:                           STS
REMOTE SERVER NAME:             sts_server
```

Figure 10. Example of an STS Connection Configuration file for TCP/IP

Editing the STS Routing Table

This section describes how to define the connections in the STS routing table.

The routing table is an editable text file that tells the TME 10 Software Distribution, Version 3.1.5 for AIX server which APPC or TCP/IP connection to use when sending a distribution to a remote target. Set up this table to reference the connections that you have configured for your system. A routing table can contain up to 1600 entries.

You do not need to configure a routing table if you do not have remote communication installed. You only need to edit it when you add a new remote connection, or when you want to tune a complex network by careful matching of routes and connections.

A sample routing table is set up for you when you install the communication option. The file is /usr/lpp/netviewdm/db/routetab. You need to change this table to define *routes* in it.

The routing table contains the definition of the network ID for the server and for each of the clients in a network. It also defines a number of routes. Each route has the following data associated with it:

- Protocol type: APPC, TCP/IP, or BOTH
- Addresses of the targets reached by this route
- The name of the connection to use
- The hop count to use on distributions sent on this route

Planning a Simple Routing Table

In a simple configuration, your server is attached to just one remote node. Only one connection is defined and this takes all distributions to and from the server.

You need only configure a single route. The SNA/DS address should be given as *.* to allow any SNA/DS name to be routed. If included, set the connection service parameters to show that any value is supported on the connection.

Planning a Complex Routing Table

Plan a complex routing table with care. Remember that a distribution is sent on the first route in the table that meets the requirements of the distribution. You should, therefore, put specific SNA/DS addresses before generic ones, and restricted connections before all-purpose ones.

You may find it helpful to draw a diagram of your connections to adjacent NetView DM nodes before you start. In your diagram, you can put the SNA/DS addresses of the remote targets reached through each node, and the services offered by each connection.

The diagram helps you ensure that distributions are routed to the correct destinations, and that no distribution is unexpectedly rejected because there is no connection capable of carrying it.

If you have a busy system, you can reserve some connections for handling small, high-priority distributions. This prevents them from being held up until active, large distributions complete. You can do this by setting the priority and capacity service parameters on the reserved connections.

Determining Destination Addresses

Routes in your routing table must identify the destination target of the link being defined. A target is identified by two values:

- Its domain address
- Its target address

These two values are separated by a period (.) and are expressed in this format in routes in the routing table:

`<domain address>.<target address>`

The `domain address` is the same as a target's `server name`.

In SNA terminology, these values correspond to:

`<routing group name>.<routing element name>`

which is commonly expressed as:

`<RGN>.<REN>`

The two values are referred to using different terms on the various platforms that can be included in a TME 10 Software Distribution, Version 3.1.5 for AIX network. To identify a target's `<domain address>` and `<target address>` look for these corresponding terms:

Table 18. Target Identification Terms		
Type of Target	**Domain address**	**Target address**
NetView DM for MVS	Network ID *	LU name *
NetView DM/2	Network ID	LU name
AS/400	Network ID	LU name
NetView DM for NetWare	domain address	target address
TME 10 Software Distribution, Version 3.1.5 for AIX	domain address	target address
Note: * In a NetView DM for MVS-controlled network, the network ID is the same as the network ID defined for the NetView DM for MVS system it is part of. The LU name uniquely identifies the node within that network. You can find out what this is by contacting the administrator at that site. You set the local network ID just once. This value is picked up for the RGN of all local nodes.		

To define a target using the command line interface, see the *TME 10 Software Distribution Reference*.

Editing the STS Routing Table

Defining Routes

Before defining the routes in a routing table, the type of network you are defining must be specified. Specify one of the following values for the NETWORK PROTOCOL keyword at the beginning of the routing table:

APPC   APPC routes are being defined. Use APPC to specify APPN connections as well.

TCP/IP   TCP/IP routes are being defined. This is the default value, even if this keyword is not included in the routing table.

BOTH   Both APPC and TCP/IP routes are being defined in the routing table.

Each route is defined by exactly one line in the routing table. Lines in a routing table can be any length. Blank lines are permitted between routes in the routing table. Comment lines begin with a hash character (#). You should specify data using the characters listed in "Entering Data" on page 70.

Enter the following information in the order shown to define a route.

Parameter	Description
Destination Address	The destination nodes that this route serves. TME 10 Software Distribution, Version 3.1.5 for AIX searches the table for an entry matching the destination of each distribution that it sends. If no match is found, a distribution report is generated and returned to the originator of the distribution.
	Enter the address in the form:
	`<domain address>.<target address>`
	Each value can be up to eight characters long. The characters must be either uppercase letters, numbers, or the special characters @, #, and $. For example `NETWK1.LU0001`.
	You can use the asterisk (*) and question mark (?) as wildcard characters.
	See "Determining Destination Addresses" on page 105 for additional information.
Connection Name	The connection name for this route. This name relates to the STS connection configuration file that you create under the `/usr/lpp/netviewdm/db/snadscon` directory (see "Editing STS Connection Configuration Files" on page 101). For consistency this name can be the host name of the system you are connecting to.

Parameter	Description
Hop Count	This parameter defines the hop count for all distributions sent out using this route. It indicates the maximum number of nodes that the distribution can legitimately pass through before reaching its destination. The hop count prevents distributions from looping between nodes in the network with contradictory routing tables.
	Enter the hop count as a decimal digit. Set the value to one if the next node on this route is an end node. If you are unsure of the topology of your network, set the field to 5. Hop count can be omitted from a routing table. If it is not defined, the value defaults to the one defined for Origin Hop Count in the SNA/DS configuration file (see "STS Configuration File" on page 101).

Sample routing tables follow. They are supplied with TME 10 Software Distribution, Version 3.1.5 for AIX in the /usr/lpp/netviewdm/db/routetab file. The comment line containing the column headings for the configuration information is included to facilitate reading.

```
# STS ROUTING TABLE
# This table provides STS routing information for
# APPC routes.
# This file should be stored as /usr/lpp/netviewdm/db/routetab

NETWORK PROTOCOL:   APPC
#
# Destination    Connection     Hop
# address                       Count
#

SRVSNA1.*        CONNSNA1       5
SRVSNA2.*        CONNSNA2       5
*.*              CONNSNA3       5
```

Figure 11. Example of an STS Routing Table for APPC routes

An example of an STS routing table for TCP/IP routes follows.

Editing the STS Routing Table

```
# STS ROUTING TABLE
# This table provides STS routing information for
# TCP/IP routes.
# This file should be stored as /usr/lpp/netviewdm/db/routetab

NETWORK PROTOCOL:   TCP/IP

#
# Destination    Connection    Hop
# Address                      Count
#

SRVTCP1.*        CONNTCP1      10
SRVTCP2.*        CONNTCP2      10
*.*              CONNTCP3      10

```

Figure 12. Example of an STS Routing Table for TCP/IP routes

An example of an STS routing table for both APPC and TCP/IP routes follows.

```
# STS ROUTING TABLE
# This table provides STS routing information for
# TCP/IP and APPC routes.
# This file should be stored as /usr/lpp/netviewdm/db/routetab

NETWORK PROTOCOL:   BOTH

#
# Destination    Connection      Hop
# Address                        Count
#

SRVSNA1.*         CONNSNA1         10
SRVSNA2.*         CONNSNA2         10
*.*               CONNSNA3         10
NETWK1.HOST       CONNSNA4         10

SRVTCP1.*         CONNTCP1         10
SRVTCP2.*         CONNTCP2         10
*.*               CONNTCP3         10

```

Figure 13. Example of an STS Routing Table for APPC and TCP/IP protocols

Defining STS Targets

Target name and address information is specified by way of various parameters, which have been described in the *TME 10 Software Distribution Reference*. This section is a brief summary of this information in relation to target definition for STS communication. For each target you define:

- **Server name and target address**

  You provide this information when you define a target using the graphical interface or the **addtg** line command. The combination of the two (`<server name>`.`<target address>`) uniquely identifies the target in a network.

- **Protocol type and Side Information Profile name**

  Additional target identification information is needed for STS connections. A target's protocol type can be either TCP/IP or APPC. The default is TCP/IP, and the TCP/IP host name for the target is located automatically. If it is APPC, then you must also supply either the name of the Side Information Profile defined for the target when configuring its SNA Server/6000 Version 2.1 parameters (see "Defining the LU 6.2 Side Information Profiles" on page 149), or a target's network ID, LU name, and LU mode.

Configuration Line Commands

You can list, load, and update TME 10 Software Distribution, Version 3.1.5 for AIX configuration parameters using these commands that are described in detail in the *TME 10 Software Distribution Reference.*

Chapter 13. Configuring SNA/DS Remote Connection Files

This chapter describes how to configure configuration files for an SNA/DS network that uses either APPC or TCP/IP connections. You need to edit this configuration information to define or update SNA/DS connections to remote nodes.

Editing Connection Configuration Files

The configuration information for SNA/DS connections is held in three different text files, which can be edited using a text editor.

The files are owned by `root` and can be accessed by the administrator, the same user who can make configuration changes to TME 10 Software Distribution, Version 3.1.5 for AIX using the graphical interface.

The files that can be edited are:

SNA/DS configuration file

> This file contains the system parameters for controlling TME 10 Software Distribution, Version 3.1.5 for AIX's use of the SNA/DS network implemented across both APPC and TCP/IP connections.

SNA/DS connection configuration files

> This file contains the details of an APPC or a TCP/IP connection to an adjacent node.

Routing table

> This file instructs the server which APPC or TCP/IP connection should be used use when distributions are sent to remote targets.

If you edit one of these files while TME 10 Software Distribution, Version 3.1.5 for AIX is running, you must use the **rld** command to reload configuration changes. See the *TME 10 Software Distribution Reference* for more information about this command.

SNA/DS Configuration File

The SNA/DS configuration file, `/usr/lpp/netviewdm/db/snadscfg`, contains system parameters for controlling the use that TME 10 Software Distribution, Version 3.1.5 for AIX makes of the SNA/DS network. You only need to edit the SNA/DS configuration file when performing system tuning, and access is usually restricted to the administrator.

The file is stored with a fixed text format. Each line starts with one of the keywords described in Table 19 on page 112. Enter each keyword in uppercase and end it with a colon. You should specify data using the characters listed in "Entering Data" on page 70.

Each keyword can be used only once. The order of the keywords is not important, and blank and comment lines can be included. Comment lines begin with a hash character (#).

SNA/DS Configuration File

Table 19. SNA/DS Configuration File Parameters

Keyword	Description
ORIGIN HOP COUNT	The default hop count to be used for distributions originated by TME 10 Software Distribution, Version 3.1.5 for AIX when the hop count is omitted or is specified as zero in the routing table. (See "Editing the SNA/DS Routing Table" on page 116.) An appropriate value for this field is 5.
MAX CRMU	The maximum number of Completion Report Message Units (CRMU) that TME 10 Software Distribution, Version 3.1.5 for AIX sends in one go when it detects that Message Unit ID (MU_ID) registries are out of synchronization. You do not normally need to change the value of this field. It is used for performance tuning. If the value is low then resynchronization of registries take longer. If the value is high, then system resources are consumed by the resynchronization and are not available for normal processing. Because resynchronization should be rare, and it is unlikely that a connection would have many MU_IDs outstanding at any time, 5 is a reasonable value.
TRANSMISSION HOLD TIME	The time for which a connection should be held by TME 10 Software Distribution, Version 3.1.5 for AIX after a severe transitory error or an error requiring operator intervention has been detected. The value you enter should be in seconds. An appropriate value is 1000 seconds (16 minutes).
ALLOCATION FAILURE RETRY TIME	The number of seconds that TME 10 Software Distribution, Version 3.1.5 for AIX is to wait before attempting to reestablish a conversation on this connection after a previous conversation has been unsuccessful. This prevents thrashing when a connection is unavailable. The value entered should be a decimal number. A reasonable value is 300 seconds (five minutes).

The following is an example of the SNA/DS Configuration File.

```
# SNA/DS CONFIGURATION FILE
#

# This file should be stored as /usr/lpp/netviewdm/db/snadscfg

ORIGIN HOP COUNT:                10
MAX CRMU:                        5
TRANSMISSION HOLD TIME:          3600
ALLOCATION FAILURE RETRY TIME:   1000
```

Figure 14. Example of an SNA/DS Configuration file

SNA/DS Connection Configuration Files

SNA/DS connection configuration files define the details of APPC or TCP/IP connections to adjacent nodes. Each file is given the name of the connection it defines. Up to 800 connection configuration files can be defined, while up to 100 connections can be active simultaneously.

The files are stored in the directory /usr/lpp/netviewdm/db/snadscon. Two sample files are provided for SNA/DS connections:

- connsna, for APPC connections
- conntcp, for TCP/IP connections

Access is usually restricted to the administrator. Sample connection configuration files are provided in the same directory.

Each file is stored with a fixed text format. Each line starts with one of the keywords described in Table 20. Enter each keyword in uppercase and end it with a colon. You should specify data using the characters listed in "Entering Data" on page 70.

Each keyword can be used only once. The order of the keywords is not important, and blank and comment lines can be included. Comment lines begin with a hash character (#).

Table 20 (Page 1 of 3). SNA/DS Connection Configuration File Parameters	
Keyword	**Description**
PROTOCOL	Specify either:
	APPC The APPC protocol is used. This is the default. Use this value to specify the APPN protocol as well.
	TCP/IP. The TCP/IP protocol is used.
TYPE	

SNA/DS Connection Configuration File

Table 20 (Page 2 of 3). SNA/DS Connection Configuration File Parameters	
Keyword	**Description**
SEND TP SYMBOLIC DESTINATION	The name of the SNA Services/6000 Version 1.2 or SNA Server/6000 Version 2.1 Side Information Profile that is used when TME 10 Software Distribution, Version 3.1.5 for AIX starts a conversation to receive on this connection. That is, the profile to use when starting a conversation with the Send TP at the remote node. The name of the profile is assigned when you configure SNA profiles (see "Defining the LU 6.2 Side Information Profiles" on page 149). The entry can be up to eight characters long.
	This keyword is valid only if the PROTOCOL specified is APPC.
RECEIVE TP SYMBOLIC DESTINATION	The name of the SNA Services/6000 Version 1.2 or SNA Server/6000 Version 2.1 Side Information Profile that is to be used when TME 10 Software Distribution, Version 3.1.5 for AIX starts a conversation to Send on this connection. That is, the profile to use when starting a conversation with the Receive TP at the remote node. The name of the profile is assigned when you configure SNA profiles (see "Defining the LU 6.2 Side Information Profiles" on page 149). The entry can be up to eight characters long.
	This keyword is valid only if the PROTOCOL specified is APPC.
REMOTE SERVER NAME	The name specified for the remote target when it was defined with the graphical interface add target operation or the **addtg** command. It is the name of the target inserted in the product databse.
TCP/IP TIME-OUT	The number of seconds that TME 10 Software Distribution, Version 3.1.5 for AIX waits for a message on a TCP/IP connection before deciding there is a communication error and thus terminates the distribution.
	This keyword is valid only if the PROTOCOL specified is TCP/IP. The value entered should be a decimal number. A reasonable value is 300 (five minutes).
NEXT DSU	The RGN and REN of the SNA/DS node at the other end of this connection. You must find out this information from the system administrator at the other node. If the PROTOCOL specified is TCP/IP, this field must specify the target address and the domain address of the remote server.
	The RGN and REN must be entered as consecutive strings separated by a period.

Table 20 (Page 3 of 3). SNA/DS Connection Configuration File Parameters	
Keyword	**Description**
TRANSMISSION TIME-OUT	The number of seconds that TME 10 Software Distribution, Version 3.1.5 for AIX is to wait before retrying the transmission of an MU_ID that failed or was interrupted. This prevents thrashing when a transitory error occurs.
	The value entered should be a decimal number. A reasonable value is 60 seconds.
RETRY LIMIT	The number of times that TME 10 Software Distribution, Version 3.1.5 for AIX is to retry the transmission of a distribution (file or change management command) before reporting an error.
	The value entered should be a decimal number. A reasonable value is 3.
SEND MU_ID TIME-OUT	The number of seconds that TME 10 Software Distribution, Version 3.1.5 for AIX is to wait before retrying the transmission of a distribution. This value prevents thrashing if the remote node has detected a transitory error.
	The value entered should be a decimal number. A reasonable value is 60.
RECEIVE MU_ID TIME-OUT	The number of seconds that TME 10 Software Distribution, Version 3.1.5 for AIX is to wait before retrying the receipt of a distribution. This value prevents thrashing if a transitory error has been detected.
	The value entered should be a decimal number. A reasonable value is 120.

The following is an example of an SNA/DS connection configuration file for the APPC protocol. This sample file, called CONNSNA, is installed during product installation.

```
# SNA/DS CONNECTION CONFIGURATION FILE FOR CONNECTION connsna
#
# This connection is used to handle transmissions between
# TME 10 Software Distribution, Version 3.1.5 for AIX servers.
#
# This file should be stored as /usr/lpp/netviewdm/db/snadscon/connsna

PROTOCOL:                       APPC
TYPE:                           SNA
SEND TP SYMBOLIC DESTINATION:   NVDMSIDS
RECEIVE TP SYMBOLIC DESTINATION: NVDMSIDR
NEXT DSU:                       NTWRK1.LU0001
TRANSMISSION TIME-OUT:          60
RETRY LIMIT:                    3
SEND MU_ID TIME-OUT:            60
RECEIVE MU_ID TIME-OUT:         120
```

Figure 15. Example of an SNA/DS Connection Configuration file for APPC

The following is an example of an SNA/DS connection configuration file for the TCP/IP protocol. This sample file, called CONNTCP, is installed during product installation.

```
# SNA/DS CONNECTION CONFIGURATION FILE FOR CONNECTION conntcp
#
# This connection is used to handle transmissions between
# two servers om a TME 10 Software Distribution, Version 3.1.5 for AIX network.
#
# This file should be stored as /usr/lpp/netviewdm/db/snadscon/conntcp

PROTOCOL:                       TCP/IP
TYPE:                           SNA
REMOTE SERVER NAME:             SERVER
TCP/IP TIME-OUT:                300
NEXT DSU:                       NTWRK2.SERVER
TRANSMISSION TIME-OUT:          60
RETRY LIMIT:                    3
SEND MU_ID TIME-OUT:            60
RECEIVE MU_ID TIME-OUT:         120
```

Figure 16. Example of an SNA/DS Connection Configuration file for TCP/IP

Editing the SNA/DS Routing Table

This section describes how to define the connections in the SNA/DS routing table.

The routing table is an editable text file that tells the TME 10 Software Distribution, Version 3.1.5 for AIX server which APPC or TCP/IP connection to use when sending a distribution to a remote target. Set up this table to reference the connections that you have configured for your system. A routing table can contain up to 1600 entries.

You do not need to configure a routing table if you do not have remote communication installed. You only need to edit it when you add a new remote connection, or when you want to tune a complex network by careful matching of routes and connections.

A sample routing table is set up for you when you install the communication option. The file is /usr/lpp/netviewdm/db/routetab. You need to change this table to define *routes* in it.

The routing table contains the definition of the network ID for the server and for each of the clients in a network. It also defines a number of routes. Each route has the following data associated with it:

- Protocol type: APPC, TCP/IP, or BOTH
- SNA/DS address of the targets reached by this route
- The name of the connection to use
- The SNA/DS hop count to use on distributions sent on this route

The following parameters can also be included in a routing table for APPC routes:

- Service parameter thresholds specifying:
    - The minimum distribution priority allowed on the route
    - The maximum size distribution allowed on this route
    - The SNA/DS security provided by this route
    - The SNA/DS protection provided by this route

Planning a Simple Routing Table

In a simple configuration, your server is attached to just one remote node. Only one connection is defined and this takes all distributions to and from the server.

You need only configure a single route. The SNA/DS address should be given as *.* to allow any SNA/DS name to be routed. If included, set the connection service parameters to show that any value is supported on the connection.

Planning a Complex Routing Table

Plan a complex routing table with care. Remember that a distribution is sent on the first route in the table that meets the requirements of the distribution. You should, therefore, put specific SNA/DS addresses before generic ones, and restricted connections before all-purpose ones.

You may find it helpful to draw a diagram of your connections to adjacent NetView DM nodes before you start. In your diagram, you can put the SNA/DS addresses of the remote targets reached through each node, and the services offered by each connection.

The diagram helps you ensure that distributions are routed to the correct destinations, and that no distribution is unexpectedly rejected because there is no connection capable of carrying it.

If you have a busy system, you can reserve some connections for handling small, high-priority distributions. This prevents them from being held up until active, large

Editing the SNA/DS Routing Table

distributions complete. You can do this by setting the priority and capacity service parameters on the reserved connections.

Determining Destination Addresses

Routes in your routing table must identify the destination target of the link being defined. A target is identified by two values:

- Its domain address (the same as its server name)
- Its target address

These two values are separated by a period (.) and are expressed in this format in routes in the routing table:

```
<domain address>.<target address>
```

The `domain address` is the same as a target's `server name`.

In SNA terminology, these values correspond to:

```
<routing group name>.<routing element name>
```

which is commonly expressed as:

```
<RGN>.<REN>
```

The two values are referred to using different terms on the various platforms that can be included in a TME 10 Software Distribution, Version 3.1.5 for AIX network. To identify a target's `<domain address>` and `<target address>` look for these corresponding terms:

Table 21. Target Identification Terms

Type of Target	Domain address	Target address
NetView DM for MVS	Network ID *	LU name *
NetView DM/2	Network ID	LU name
AS/400	Network ID	LU name
NetView DM for NetWare	domain address	target address
TME 10 Software Distribution, Version 3.1.5 for AIX	domain address	target address

Note: * In a NetView DM for MVS-controlled network, the network ID is the same as the network ID defined for the NetView DM for MVS system it is part of. The LU name uniquely identifies the node within that network. You can find out what this is by contacting the administrator at that site. You set the local network ID just once. This value is picked up for the RGN of all local nodes.

To define a target using the command line interface, see the *TME 10 Software Distribution Reference*.

Defining Routes

Before defining the routes in a routing table, the type of network you are defining must be specified. Specify one of the following values for the NETWORK PROTOCOL keyword at the beginning of the routing table:

APPC APPC routes are being defined. Use APPC to specify APPN connections as well.

TCP/IP TCP/IP routes are being defined. This is the default value, even if this keyword is not included in the routing table.

BOTH Both APPC and TCP/IP routes are being defined in the routing table.

Each route is defined by exactly one line in the routing table. Lines in a routing table can be any length. Blank lines are permitted between routes in the routing table. Comment lines begin with a hash character (#). You should specify data using the characters listed in "Entering Data" on page 70.

Enter the following information in the order shown to define a route.

Parameter	Description
Destination Address	The destination SNA/DS nodes that this route serves, as an SNA RGN.REN address. TME 10 Software Distribution, Version 3.1.5 for AIX searches the table for an entry matching the destination of each distribution that it sends. If no match is found, a distribution report is generated and returned to the originator of the distribution.
	Enter the address in the form:
	`<domain address>.<target address>`
	Each value can be up to eight characters long. The characters must be either uppercase letters, numbers, or the special characters @, #, and $. For example `NETWK1.LU0001`.
	You can use the asterisk (*) and question mark (?) as wildcard characters.
	See "Determining Destination Addresses" on page 118 for additional information.

Editing the SNA/DS Routing Table

Parameter	Description
Minimum Distribution Priority	The minimum distribution priority that this route supports. This parameter is optional, and can only be used for APPC routes.
	When TME 10 Software Distribution, Version 3.1.5 for AIX has matched an SNA/DS address, it checks the priority available. If the distribution does not have high enough priority to use this route, TME 10 Software Distribution, Version 3.1.5 for AIX continues to search for another one.
	The priority values are:
	FAST Allows only distributions with priority FAST on this route.
	CONTROL Allows only distributions with priority FAST or CONTROL on this route.
	DATA16 Allows only distributions with priority FAST, CONTROL, or DATA16 on this route.
	DATA15 Allows any distribution with priority DATA15 or higher on this route.
	DATA14 ... DATA1 Allow any distribution with priority DATAn or higher on this route, where n is any number in the range 1 through 14.
	ANY Allows any priority distribution on this route.
Distribution Protection	The distribution protection that this route provides. The protection parameter specifies whether the distribution must be stored on nonvolatile storage while a DSU has responsibility for it. If a distribution requires more protection than a route can provide, the route is not used. This parameter is optional, and can only be used for APPC routes.
	The protection values are:
	LEVEL2 Level-2 protection is provided on this route, which indicates that the distribution is safe-stored in nonvolatile storage.
	ANY Any requested level of protection is supported on this route.

Parameter	Description
Maximum Size Distribution	The maximum size distribution that this route supports. This is the capacity of the route. If a distribution is larger than this size, the route is not used. This parameter is optional, and can only be used for APPC routes.
	The capacity values are:
	0DATA
	No distribution carrying data is allowed on this connection. Only reports are sent.
	1MEGABYTE
	Only distributions with less than 1MB of data are sent on this connection.
	4MEGABYTES
	Only distributions with less than 4MB of data are sent on this connection.
	16MEGABYTES
	All distributions can use this connection.
	ANY
	All distributions can use this connection regardless of the amount of data being carried.
Distribution Security	The distribution security that this route provides. The security parameter specifies that the distribution is to be safeguarded from unauthorized access while it is being sent through the DS network. If a distribution requests higher security than a route can provide, the route is not used. This parameter is optional, and can only be used for APPC routes.
	The security values are:
	LEVEL2
	Level-2 security is provided on this route, which indicates that DS should route the distribution on sessions that are designated secure.
	ANY
	Any requested level of security is supported on this route.
Connection Name	The connection name for this route. This name relates to the SNA/DS connection file that you create under the `/usr/lpp/netviewdm/db/snadscon` directory. For consistency this name can be the host name of the system you are connecting to. For information about defining the SNA/DS connection, refer to Chapter 15, "Configuring APPC Communication" on page 127.

Editing the SNA/DS Routing Table

Parameter	Description
Hop Count	This parameter defines the hop count for all distributions sent out using this route. It indicates the maximum number of SNA/DS nodes that the distribution can legitimately pass through before reaching its destination. The hop count prevents distributions from looping between SNA/DS nodes in the network with contradictory routing tables.
	Enter the hop count as a decimal digit. Set the value to one if the next node on this route is an end node. If you are unsure of the topology of your network, set the field to 5. Hop count can be omitted from a routing table. If it is not defined, the value defaults to the one defined for Origin Hop Count in the SNA/DS configuration file (see "SNA/DS Configuration File" on page 111).

Sample routing tables follow. They are supplied with TME 10 Software Distribution, Version 3.1.5 for AIX in the `/usr/lpp/netviewdm/db/routetab` file. The comment line containing the column headings for the configuration information is included to facilitate reading.

```
# SNA/DS ROUTING TABLE
# This table provides SNA/DS routing information for a
# TME 10 Software Distribution, Version 3.1.5 for AIX SNA network.
# This file should be stored as /usr/lpp/netviewdm/db/routetab

NETWORK PROTOCOL:   APPC
#
# Destination  Priority Protection   Capacity  Security  Connection    Hop
# Address
#

SRVSNA1.*        DATA16    ANY         1MEGABYTE ANY         CONNSNA1
SRVSNA2.*        ANY       ANY         ANY       ANY         CONNSNA2
*.*              ANY       ANY         ANY       ANY         CONNSNA3
```

Figure 17. Example of an SNA/DS Routing Table for APPC routes

An example of an SNA/DS routing table for a TCP/IP connection follows.

```
# SNA/DS ROUTING TABLE
# This table provides SNA/DS routing information for
# a TME 10 Software Distribution, Version 3.1.5 for AIX TCP/IP network.
# This file should be stored as /usr/lpp/netviewdm/db/routetab

NETWORK PROTOCOL:   TCP/IP

#
# Destination          Connection
# address
#

SRVTCP1.*          CONNTCP1
SRVTCP2.*          CONNTCP2
*.*                CONNTCP3
```

Figure 18. Example of an SNA/DS Routing Table for TCP/IP routes

An example of an SNA/DS routing table for both APPC and TCP/IP connection
protocols follows.

```
# SNA/DS ROUTING TABLE
# This table provides SNA/DS routing information for
# a TME 10 Software Distribution for AIX combination TCP/IP and APPC network.
# This file should be stored as /usr/lpp/netviewdm/db/routetab

NETWORK PROTOCOL:   BOTH

#
# SNA/DS         Priority Protection   Capacity  Security Connection      Hop
# Destination
# address
#

SRVSNA1.*       DATA16   ANY          1MEGABYTE ANY      CONNSNA1
SRVSNA2.*       ANY      ANY          ANY       ANY      CONNSNA2
*.*             ANY      ANY          ANY       ANY      CONNSNA3
NETWK1.HOST     ANY      ANY          ANY       ANY      CONNSNA4

SRVTCP1.*                                                CONNTCP1
SRVTCP2.*                                                CONNTCP2
*.*                                                      CONNTCP3
```

Figure 19. Example of an SNA/DS Routing Table for both APPC and TCP/IP protocols

Configuration Line Commands

You can list, load, and update TME 10 Software Distribution, Version 3.1.5 for AIX
configuration parameters using these commands that are described in detail in the
TME 10 Software Distribution Reference.

Chapter 14. Configuring TCP/IP Communication

Many of the TCP/IP network configuration tasks that you have to perform involve editing configuration files, and have already been described in Chapter 12, "Configuring STS Remote Connection Files" on page 101 and Chapter 13, "Configuring SNA/DS Remote Connection Files" on page 111. They are:

- Determine the network ID of the STS or SNA/DS network that your servers are attached to and the LU names (short names) that provide access to the servers. Use these to set the addresses of the clients and servers in the domains you are defining.

- Configure the STS or SNA/DS connections which the CC servers use to send and receive distributions.

- Construct the STS or SNA/DS routing table that TME 10 Software Distribution, Version 3.1.5 for AIX uses to route distributions to remote targets.

- Customize the STS or SNA/DS communication used by TCP/IP, providing data that influences how TME 10 Software Distribution, Version 3.1.5 for AIX sends and receives distributions on all connections.

This section describes how to complete TCP/IP configuration on your RISC System/6000 to allow communication within the same TCP/IP domain or across different TCP/IP domains. You need to perform these tasks only once, when TCP/IP is installed, and then only if your system has the server or the remote administration option installed.

You must provide the following information:

- The name of the network interface you are using, for example, token ring or Ethernet.

- The host name for the server that you are configuring. This is the TCP/IP name by which this node is known.

- The Internet address of the node.

- The Internet address mask to define subnetworks if appropriate.

If you are in doubt about values for these fields, contact your network administrator and read the corresponding TCP/IP configuration manuals.

You can define the time in seconds after which, when the server attempts to connect to a target, a TCP/IP communication is considered failed. Set the `CONN_TMO` environment variable by entering the following command at an AIX command line:

`export CONN_TMO=number_of_seconds`

where `number_of_seconds` can be a value from 0 to 75. The default is 10.

Note: In many cases, TME 10 Software Distribution, Version 3.1.5 for AIX is installed on a working system. If the server is already using TCP/IP on the LAN, you do not need to perform these configuration steps.

Configuring TCP/IP Communication

This section describes the steps that you follow to configure TCP/IP so that your TME 10 Software Distribution, Version 3.1.5 for AIX server can communicate with TME 10 Software Distribution, Version 3.1.5 for AIX clients on the LAN. The section assumes that you have already installed TCP/IP on your server.

TCP/IP is configured using smit. Smit displays a series of menus and when you select appropriate values, menus appear for you to define profiles for TCP/IP.

Using SMIT to Configure TCP/IP

You need root privileges to perform these operations.

To configure TCP/IP, perform the following steps:

1. To load smit, on the AIX command line type `smit`.

   The System Management menu is displayed.

2. From the System Management menu, select **Communications Applications and Services**.

3. From the Communications Applications and Services menu, select **TCP/IP**.

4. From the TCP/IP menu, select **Minimum Configuration & Startup**.

5. You are prompted to select the network interface on which you want TCP/IP to run over (for example, token-ring network interface).

   The Minimum Configuration & Startup screen for the network interface that you have chosen is displayed. There are many fields that you can set; however, most of them have defaults. You need only set the following fields:

HOSTNAME	The name used to refer to the server. This is often called the TCP/IP name of the server. Enter a unique, easily remembered name, up to 12 characters long.
INTERNET Address	The Internet address of the server. Enter four decimal numbers separated by periods. This address is unique and is assigned by your network administrator
Network MASK	The network mask defines the subnetworks that your server can see. Enter a mask in the form of an Internet address. Ask your network administrator for an appropriate value.
Ring Speed	Specify the speed of the token ring (either 4 or 16) you are attaching to, or the speed desired if you are establishing a new ring.
	Note: Consult your network administrator and the appropriate IBM manuals if you are in any doubt about how to set the fields. You can press F1 for contextual help at any time.

6. Press Enter to complete the configuration procedure.

Chapter 15. Configuring APPC Communication

| This chapter describes how to configure APPC for LU 6.2 communication using SNA
| Server/6000 Version 2.1 (for computers running versions of AIX prior to version 4.3.3)
| or Communications Server, Version 5 (for computers running AIX version 4.3.3. See
"Protocols to Connect to Another Server" on page 68 for details of those types of
targets that can communicate across APPC.

As you configure TME 10 Software Distribution, Version 3.1.5 for AIX for remote
communication, you come into contact with technical terms associated with SNA and
SNA/DS. You must have some understanding of these terms to complete a successful
configuration.

For more information about SNA and SNA/DS, refer to the *Systems Network
Architecture Formats*, GA27-3136, and the *Systems Network Architecture Distribution
Services*, SC30-3098. If you need more information about AIX SNA Server Version 2.1
| refer to *AIX SNA Server/6000: Configuration Reference*, SC31-7014. If you need more
| information about AIX Communications Server, Version 5 refer to *AIX Communications
| Server, Version 5 Configuration Reference*, SC31-7014.

Overview of APPC Configuration Tasks

Perform the following steps to configure your TME 10 Software Distribution, Version
3.1.5 for AIX system for remote communication:

- Determine the network ID of the SNA network your server is connected to and the
  name of the LU that provides access to your server. Use these values to set the
  addresses of your clients and server.

- Configure the APPC connections that your server uses to send and receive
| distributions. These connections must be defined to TME 10 Software
| Distribution, Version 3.1.5 for AIX and to SNA Server/6000 Version 2.1 or
| Communications Server, Version 5, as appropriate, which provides the underlying
  support.

- Construct the STS or SNA/DS routing table that TME 10 Software Distribution,
  Version 3.1.5 for AIX uses to route distributions to remote targets.

- Customize the STS or SNA/DS communication, providing data that determines how
  TME 10 Software Distribution, Version 3.1.5 for AIX sends and receives
  distributions on all connections.

- Configure the remote NetView DM for MVS nodes to accept communication from
  TME 10 Software Distribution, Version 3.1.5 for AIX.

The configuration tasks you perform depend on the communication links you are
creating. You must configure STS or SNA/DS communication at the TME 10 Software
Distribution, Version 3.1.5 for AIX server.

Configuring APPC Links

When you configure TME 10 Software Distribution, Version 3.1.5 for AIX for remote communication, the main task is the configuration of APPC at the TME 10 Software Distribution, Version 3.1.5 for AIX server itself.

Configuring the AIX Communication Device

This section describes the steps you must perform to add the device driver support for the Data Link Controls (DLC) of the specific communication adapter that you are using.

You can use SMIT for this configuration task.

> **Fields May Differ**
>
> The fields described in the following SMIT dialogs refer to a configuration for ethernet, token-ring and SDLC connections. Parameters for different types of connections may differ.
>
> Parameters may also differ according to SNA levels. Any parameters displayed on your screen and not described in this manual should be left at their default values.

1. Type `smit` on the command line to load it. The System Management menu is displayed.

2. From the System Management menu, select:

   → Devices
   → Communication

3. From the Communication menu, select the adapter that corresponds with your SNA communication adapter (for example, Ethernet, Token Ring or SDLC).

4. From the Adapter menu, select:

   → Services
   → Data Link Controls
   → Add the Data Link Control that corresponds with your adapter

5. The name of the data link control is displayed. Press Enter to add the control to your configuration.

Configuring SNA Server/6000 Version 2.1 on AIX Versions Prior to Version 4.3.3

This section only refers to configuring APPC on a computer running versions of AIX prior to version 4.3.3 with SNA Server/6000 Version 2.1; the APPC configuration for computers running AIX Version 4.3.3 with Communications Server, Version 5 installed are described in "Configuring Communications Server, Version 5 on AIX Version 4.3.3" on page 151.

This section describes the steps that you must perform to configure SNA Server/6000 Version 2.1 so that TME 10 Software Distribution, Version 3.1.5 for AIX can

communicate with remote nodes. It assumes that you have already installed SNA Server/6000 Version 2.1 on your server.

You use SMIT to perform these configuration activities. You need to be logged on with root privileges to do so. SMIT provides a series of menus for defining SNA Server/6000 Version 2.1 *profiles*. Each profile contains the configuration details for one logical component of the configuration.

You configure profiles to define the following:

- SNA node
- SNA control point
- Data link control
- Link station
- Local logical unit session
- LU mode
- LU transaction
- Partner LU
- Partner LU location
- CPIC side information

Perform the following steps to configure SNA Server/6000 Version 2.1:

1. Type smit on the command line to load it. The System Management menu is displayed.

2. From the System Management menu, select:

   → Communications Applications and Services
   → SNA Server/6000
   → Configure SNA Profiles (*fastpath name: sna*)
   → Advanced SNA Configuration

This is the Advanced SNA Configuration menu. You are now ready to configure profiles to define the SNA/DS network.

Defining an SNA Node

You can use the default SNA profile when you are defining an SNA node.

1. From the Advanced SNA Configuration menu, select:

   → SNA System Defaults
   → Change/Show a Profile (*fastpath name: _snasnach*)

   You must set the following fields on the Change/Show SNA Node Profile screen.

Table 22. Change/Show SNA Node Profile	
Field	**Description**
PROFILE name	The name used to refer to this profile. SNA is often used as a name.
All other fields	Accept the default values.

Configuring APPC Communication Using SNA Server/6000 Version 2.1

2. Press Enter after making these changes.

3. Return to the Advanced SNA Configuration menu.

Defining an SNA Control Point

1. From the Advanced SNA Configuration menu, select:

   → Control Point
   → Change/Show a Profile (*fastpath name: _snacpmch*)

   Enter the following information on the Change/Show Control Point Profile screen.

Field	Description
Table 23 (Page 1 of 3). Change/Show SNA Control Point Profile	
PROFILE name	This field displays the name of the Control Point Profile, node_cp, which you cannot change. The system uses this name to refer to the set of characteristics that you describe in this profile. You must have only one Control Point Profile in SNA Server/6000 Version 2.1.
XID node ID	Enter the XID that identifies your physical unit (PU). The node ID can be supplied by the system administrator. It is an eight-digit hexadecimal number. This value is used only for switched links (token ring, Ethernet, X.25, and switched SDLC). The eight digits are a concatenation of two values:
	Block number
	The first three hexadecimal digits of the XID node ID provide a block number, or product identifier, for the PU. The value 071 is the IBM product identifier for the RISC System/6000 workstation.
	ID number
	The last five hexadecimal digits of the XID node ID are the ID number assigned to the PU. This number is used to distinguish each piece of equipment from any other similar pieces of equipment on the network.
	If the link station connects to a VTAM (MVS) host, the ID number is the IDNUM= parameter in the VTAM PU definition statement (see "Physical and Logical Unit Definitions" on page 161).

Table 23 (Page 2 of 3). Change/Show SNA Control Point Profile	
Field	**Description**
XID node ID (continued)	Enter one of the following values:
	• For a peer network using independent LU 6.2, use the default wildcard value, * (asterisk).
	• For dependent communication with a host system, you can specify a unique XID node ID. This value must uniquely identify the local node in the subarea network.
	If you enter a value in this field, the control point can use that value to support communication with a single host PU. To support additional host PUs, you must define Link Station Profiles with different XID node IDs. All XID node IDs specified in the Control Point Profile and Link Station Profiles for the local system must be unique.
Network name	Enter the name of the network to which this TME 10 Software Distribution, Version 3.1.5 for AIX node is connected. Combined with the control point name (below), the network name forms a unique identifier (the fully qualified name) for this control point. The name can be supplied by the system administrator.
	If you are connected to a NetView DM for MVS system, then this name is the same as the NETID parameter in the VTAM start statement (see "Physical and Logical Unit Definitions" on page 161).
Control Point (CP) name	The control point name. Combined with the network name (previous field), the control point name forms a unique identifier (the fully qualified name) for this control point. The name can be supplied by the system administrator.
	If you are connecting to a NetView DM for MVS system without APPN™ support, this field is ignored. If your system has APPN support (for example if it is using VTAM 4.1 or SNA Server Version 1.2), this field is required, and must match the local control point name. It corresponds to the CPNAME in the VTAM PU statement.

| Table 23 (Page 3 of 3). Change/Show SNA Control Point Profile ||
Field	Description
Control Point alias	The name of an alias that can be used on the local system in place of the control point name. The default is the value supplied for the control point name.
Control point type	Indicates whether the SNA Server/6000 Version 2.1 control point functions as an APPN network node (enter `appn_network_node`), or an APPN end node (enter `appn_end_node`). If the control point provides intermediate session routing, enter `appn_network_node`. (Intermediate session routing is not related to the TME 10 Software Distribution, Version 3.1.5 for AIX intermediate node capability.)
All other fields	Accept the default values.

2. Press Enter and return to the Advanced SNA Configuration menu.

Defining a Token Ring SNA DLC Profile

Perform the following steps to define a data link control profile for a Token Ring connection:

1. From the Advanced SNA Configuration menu, select:

   → Links
   → Token Ring
   → Token Ring SNA DLC
   → Change/Show a Profile (*fastpath: _snatoklinkmk*)

   Enter the following information on the Change/Show Token Ring SNA DLC Profile screen.

| Table 24. Change/Show Token Ring SNA DLC Profile ||
Field	Description
PROFILE name	The name of the profile. `tok0` is suggested for TME 10 Software Distribution, Version 3.1.5 for AIX.
All other fields	Accept the default values.

2. Press Enter and return to the Advanced SNA Configuration menu.

Defining a Token Ring Link Station Profile

Perform the following steps to define a link station profile for a Token Ring connection:

1. From the Advanced SNA Configuration menu, select:

   → Links
   → Token Ring
   → Token Ring Link Station
   → Change/Show a Profile (*fastpath: _snatokattcmk*)

   Enter the following information on the Change/Show Token Ring Link Station Profile screen.

Table 25 (Page 1 of 6). Change/Show Token Ring Link Station Profile	
Field	**Description**
PROFILE name	Enter the same name used for the Physical Unit (PU) in VTAM (see "Configuring NCP and VTAM" on page 161). It is not mandatory to use the VTAM PU name in this field, but its use reflects the relationship between the link station and PU on a mainframe connection.
Use APPN Control Point's XID node ID?	Select one of the following values for this field to indicate whether or not the link station is controlled by the APPN CP of the node:
	yes This value associates the link station with the XID node ID of the APPN CP. Selecting this value places the link station under the control of the APPN CP, and provides it with full APPN support.
	no This value excludes the link station from the control of the APPN control point and places it under the control of a PU you designate. If you select this value, specify the XID node ID of the PU represented by this link station.
XID node ID	If you entered no in the previous field, enter the XID node ID of the link station's controlling PU in this field. The value is 071xxxxx. For more details on how to compose this ID, see Table 23 on page 130.
SNA DLC Profile name	Enter the name of the data link control profile. tok0 was defined in the PROFILE name field in Table 24 on page 132.

Table 25 (Page 2 of 6). Change/Show Token Ring Link Station Profile	
Field	**Description**
Stop link station on inactivity?	Set this field to either yes or no according to resource availability and desired connection establishment time. It is suggested that you accept the default value of no, which specifies that the link station remains permanently active whether it is in use or not.
Inactivity time-out	Enter the number of minutes the link station can remain active before being stopped. It is suggested that you accept the default value of 0.
LU address registration?	LU address registration is not generally required. Set this field to no.
Trace link?	Select yes or no to make the system save a sequential log of events that occur on the link used by this link station.
Trace size	Change this field only when "Trace link?" is set to yes. It specifies the type of trace that will be logged.
Access Routing	Select one of the following values to determine how the remote station is contacted.
	link_address The remote station is contacted by address. If you select this value, enter the addresses in the "Remote link address" and the "Remote SAP address" fields.
	link_name The remote station is contacted by name. If you select this value, enter the link name in the "Remote link name" field.
Remote link name	Enter the same name used for this profile. It must also be the same name used for the Physical Unit (PU) in the VTAM definition (see "Configuring NCP and VTAM" on page 161).
Remote link address	This address must match the TIC parameter in the NCP definition. See "Configuring NCP and VTAM" on page 161.
Remote SAP address	The services access point (SAP) address for the remote system. The value entered here must be the same as the value defined for the "Local SAP address" field in the SNA DLC Profile for the remote station.

Table 25 (Page 3 of 6). Change/Show Token Ring Link Station Profile	
Field	**Description**
Verify adjacent node?	Select yes or no to verify the adjacent node's identification parameters against information obtained during link activation. It is suggested that you accept the default value of no, to facilitate the activation of the link. If you specify yes, then you *must* specify exact values for the next four fields on the menu: • Network ID of adjacent node • CP name of adjacent node • XID node ID of adjacent node • Node type of adjacent node
Network ID of adjacent node	The network ID of the adjacent network node. It corresponds to the NETID parameter in VTAM. If you specified yes for Verify adjacent node? above, then you must enter the exact value for this field. If the adjacent node does not supply a control point name in XID exchanges, you must enter a value in this field and define a Partner LU 6.2 Location Profile to define adjacent node resources.
CP name of adjacent node	Enter the CP name for the adjacent node. It is the name used for the Physical Unit (PU) in VTAM, or the SSCP_ID if the node is NetView DM/MVS. If you specified yes for Verify adjacent node? above, then you must enter the exact value for this field. If the adjacent node does not supply a control point name in XID exchanges, you must enter a value in this field and define a Partner LU 6.2 Location Profile to define adjacent node resources.
XID node ID of adjacent node	The XID node ID of the adjacent node to enable the system to verify that the correct physical units are being connected during XID exchanges. Enter either 071xxxxx, or leave the default value if the node is NetView DM/MVS.

Table 25 (Page 4 of 6). Change/Show Token Ring Link Station Profile	
Field	**Description**
Node type of adjacent node	Select one of the following values to specify the node type of the adjacent node: **learn** The adjacent node type is learned through link activation exchanges. **appn_network_node** The adjacent node is an APPN network node. If the adjacent node is *not* an APPN network node and the "Verify adjacent node?" field is set to yes, link activation fails. **appn_end_node** The adjacent node is an APPN end node. If the adjacent node is *not* an APPN end node and the "Verify adjacent node?" field is set to yes, link activation fails. If you specified yes for Verify adjacent node? above, then you must enter the exact value for this field.
Solicit SSCP sessions?	Select yes or no to specify whether the RISC System/6000 local node uses this link station to request SSCP sessions from the adjacent SNA node. It is suggested that you accept the system default of yes.
Initiate call when link station is activated?	Select yes or no to indicate whether this link station is a calling link station or a listening link station. It is suggested that you accept the system default of yes, which indicates that this is a calling link station.
Activate link station at SNA start up?	Select yes or no to indicate whether the link station is activated when the SNA Server/6000 system is started. It is suggested that you accept the system default of no.
Activate on demand?	Select yes or no to specify whether the link station can dynamically activate its associated link when required by the APPN CP.
CP-CP sessions supported?	Select yes or no to indicate whether the link station can support CP-CP sessions.

Table 25 (Page 5 of 6). Change/Show Token Ring Link Station Profile

Field	Description
Adjacent network node preferred server?	If you set `CP-CP sessions supported` to `yes`, select one of the following values: **yes** The link station connects to an adjacent node that is the preferred network server. **no** If no preferred network node server is specified, SNA Server/6000 chooses the first available link station that supports CP-CP sessions.
Partner required to support CP-CP sessions?	Select `yes` or `no` to specify whether the adjacent node must or is not required to support CP-CP sessions.
Initial TG number	Accept the default value of `0`, that specifies a negotiable TG number.
Restart on normal deactivation	Accept the default value of `no`, so that the link station is not restarted when normal link deactivation procedures are concluded.
Restart on abnormal deactivation?	Accept the default value of `no`, so that the link station is not restarted after an abnormal link deactivation.
Restart on activation?	Accept the default value of `no`, so that when this link station is activated, no additional instance of the nonselective listening link station is created to service inbound link activation requests.
Effective capacity	Accept the default value of `4300800` which is required for Token Ring and Ethernet connections.
Cost per connect time	Accept the default value of `0`, the standard session-cost-per-time-unit for the communications adapter specified.
Cost per byte	Accept the default value of `0`, the standard per-byte session cost for the communications adapter specified.
Security	Accept the default value of `nonsecure`. This value indicates that none of the other values apply, or that you do not know what security level applies.
Propagation delay	Accept the default value `lan`, which specifies a propagation delay of 384 μ.

Configuring APPC Communication Using SNA Server/6000 Version 2.1

Table 25 (Page 6 of 6). Change/Show Token Ring Link Station Profile	
Field	**Description**
User-defined 1	Accept the default value of 128.
User-defined 2	Accept the default value of 128.
User-defined 3	Accept the default value of 128.
Comments	You can enter comments regarding the profile.

   2. Press Enter and return to the Advanced SNA Configuration menu.

Defining an SDLC EIA232D SNA DLC Profile

Perform the following steps to define a data link control profile for an SDLC connection:

   1. From the Advanced SNA Configuration menu, select:

   ➡ Links
   ➡ SDLC
   ➡ EIA232D
   ➡ EIA232D SNA DLC
   ➡ Change/Show a Profile (*fastpath: _sna232dlc*)

   Enter the following information on the Change/Show EIA232D SNA DLC Profile screen.

Table 26 (Page 1 of 4). Change/Show EIA232D SNA DLC Profile	
Field	**Description**
PROFILE name	The name of the profile. mpq0 is suggested for TME 10 Software Distribution, Version 3.1.5 for AIX.
Data link device name	It is suggested that you accept the default value mpq0, which is the name that the local system uses for the SDLC physical link adapter. The name must match the AIX device name for the adapter to be used.
Force disconnect time-out (1-600 seconds)	Specify the number of seconds that the system should wait for a response after requesting a disconnect from the link before closing the link station. It is suggested that you accept the default value of 120.
User-defined maximum I-Field size?	Accept the default value of no, which indicates that the maximum I-field size is system-defined.
If yes, Max. I-field size (265-30729)	Accept the default value of 265.

Table 26 (Page 2 of 4). Change/Show EIA232D SNA DLC Profile	
Field	**Description**
Max. num of active link stations (1-255)	Specify the maximum number of active link stations allowed on this port. A suggested value is 100.
Number reserved for inbound activation	Accept the default value of 0.
Number reserved for outbound activation	Accept the default value of 0.
Serial encoding	Select the type of data encoding to use for transmitting data over the link. If you are configuring a link to an existing network, use the encoding type that the rest of the network uses. It is suggested that you accept the default value of nrzi (non-return-to-zero inverted recording). The alternative is nrz (non-return-to-zero recording).
Request to send (RTS)	Accept the default value of controlled, which indicates that the system activates the RTS signal when required prior to each transmission on the link. The alternative is continuous.
DTR control	Accept the default value of DTR, which means that the data terminal ready signal indicates whether the data terminal equipment (DTR) is ready.
Bit clocking	Accept the default value of external, which indicates that the modem provides its own data clocking. The alternative is internal, meaning that SNA Server/6000 must provide for bit clocking.
Transmit rate	Accept the default value of 1200, since this field is only required when Bit clocking is set to internal.
Network type	Select whether the type of transmission medium used by the network is switched or nonswitched. It is suggested you accept the default value of switched.
Answer mode	Select whether the modem answer mode is automatic or manual. It is suggested you accept the default value of manual.

Table 26 (Page 3 of 4). Change/Show EIA232D SNA DLC Profile	
Field	**Description**
Transmit window count	Specify the number of information SDLC I-frames to send to the remote station before waiting for a response. It is suggested you accept the default value of 7.
Retransmit count	Specify the number of retransmissions the local system should attempt before declaring that a permanent transmission error has occurred. It is suggested you accept the default value of 10.
Retransmit threshold	Specifies the number of SDLC I-frame retransmissions allowed as a percentage of the total number of SDLC I-frame transmissions. It is suggested you accept the default value of 10.
Secondary inactivity time-out	Specifies the number of seconds that secondary stations using this link wait for a transmission from the primary station. It is suggested you accept the default value of 30.
Primary repoll frequency (1-250, of .1 second)	Specifies the length of time (in units of 0.1 second) that the primary station should wait for a response after having polled the secondary station. It is suggested you accept the default value of 30.
Primary repoll threshold (1-100 %)	Specifies the number of repolls as a percentage of the total polls sent to the secondary station. It is suggested you accept the default value of 10.
Primary repoll count (1-50 repolls)	Specifies the number of times that the primary station will unsuccessfully poll a secondary station before marking the secondary station as not working. It is suggested you accept the default value of 15.
Link type	Select one of the following values: **point-to-point** The local system is connected to the secondary station with a point-to-point link. A point-to-point link connects only two stations. This value is the default. **multipoint** Select this value if the local system is to act as the primary station on a multipoint line. A multipoint line enables the primary station to communicate with several secondary stations simultaneously.

Table 26 (Page 4 of 4). Change/Show EIA232D SNA DLC Profile	
Field	**Description**
Primary idle list poll frequency (30-180 seconds)	This field specifies the amount of time, in seconds, that the primary station should wait between polls of stations on the idle list. It is suggested you accept the default value of 60.
Primary slow list poll frequency (1-60 seconds)	This field specifies the amount of time, in seconds, that the primary station should wait between polls of stations on the slow list. It is suggested you accept the default value of 1.
Retry interval (1-1000 seconds)	Specifies the number of seconds that should pass between attempts to restart the link station. It is suggested you accept the default value of 60.
Retry limit (0 or 1-500 seconds)	Specifies the number of times that SNA Server/6000 should try to restart the link station. It is suggested you accept the default value of 20.
Comments	You can enter a comment as part of the profile.

2. Press Enter and return to the Advanced SNA Configuration menu.

Defining an SDLC EIA232D Link Station Profile

Perform the following steps to define a link station profile for an SDLC connection:

1. From the Advanced SNA Configuration menu, select:

   ➡ Links
   ➡ SDLC
   ➡ EIA232D
   ➡ EIA232D Link Station
   ➡ Change/Show a Profile (*fastpath: _sna232lnk*)

   Enter the following information on the Change/Show SDLC EIA232D Link Station Profile screen.

Table 27 (Page 1 of 3). Change/Show SDLC EIA232D Link Station Profile	
Field	**Description**
PROFILE name	Enter the same name used for the Physical Unit (PU) in VTAM (see "Configuring NCP and VTAM" on page 161). It is not mandatory to use the VTAM PU name in this field, but its use reflects the relationship between link station and PU on a mainframe connection.

Table 27 (Page 2 of 3). Change/Show SDLC EIA232D Link Station Profile	
Field	**Description**
Use APPN Control Point's XID node ID?	Select one of the following values for this field to indicate whether the link station is controlled by the node's APPN CP:
	yes This value associates the link station with the XID node ID of the APPN CP.
	no This value excludes the link station from the control of the APPN control point and places it under the control of a PU you designate. If you select this value, you must specify the XID node ID of the PU represented by this link station.
XID node ID	If you entered no in the previous field, you must enter the XID node ID of the link station's controlling PU in this field. The value is 071xxxxx. For more details on how to compose this ID, see Table 23 on page 130.
SNA DLC Profile name	Enter the name of the data link control profile. mpq0 was defined above.
Stop link station on inactivity?	Set this field to either yes or no according to resource availability and desired connection establishment time. It is suggested you accept the default value of no.
Inactivity time-out	Enter the number of minutes the link station can remain active before being stopped. It is suggested you accept the default value of 0.
LU address registration?	LU address registration is not generally required. Set this field to no.
Trace link?	Select yes or no to specify whether you want the system to save a sequential log of events that occur on the link used by this link station.
Trace size	Change this field only when Trace link? is set to yes. It specifies the type of trace that will be logged.
Local secondary station address	Specify the SDLC local secondary station address of this link station. It is suggested that you accept the default value of 1.
Station type	Describes the operation of the local station with respect to other stations in the network. It is suggested that you accept the default value of secondary.

Table 27 (Page 3 of 3). Change/Show SDLC EIA232D Link Station Profile	
Field	**Description**
Remote secondary station address	Specify a value in this field only if the "Station type" is primary.
Remaining fields	All other fields are the same as those for a Token Ring Link Station profile. See Table 25 on page 133.

2. Press Enter and return to the Advanced SNA Configuration menu.

Defining the Local Logical Unit Profile

1. From the Advanced SNA Configuration menu, select:

   → Sessions
   → LU 6.2
   → LU 6.2 Local LU
   → Add a Profile (*fastpath: _snalocalu6mk*)

   Enter the following information on the Change/Show LU 6.2 Local LU Profile screen.

Table 28. Change/Show LU 6.2 Local LU Profile	
Field	**Description**
Profile name	It is suggested to give the profile the same name as the logical unit, to make it easy to identify.
Local LU name	This field must match the preparation system logical unit name in the VTAM listing. See "Configuring NCP and VTAM" on page 161.
Local LU alias	It is suggested that you use the same name given to the local LU name.
Local LU is dependent?	Select one of the following values:
	yes The local LU 6.2 is dependent and cannot initiate LU-LU sessions.
	no The local LU6.2 is independent; sessions are activated without SSCP control.
All other fields	Accept the default values.

Defining Conversation Modes

The characteristics of a conversation on a connection are defined by a conversation mode profile. A simple installation has just one mode defined.

Configuring APPC Communication Using SNA Server/6000 Version 2.1

For this example, define a default conversation mode profile called LU62 as follows. You only need to add other mode profiles if you have a complex system.

1. From the Advanced Configuration menu, select:

   → Sessions
   → LU 6.2
   → LU6.2 Mode
   → Add a Profile (*fastpath: _snamodemk*)

   The Add SNA LU6.2 Mode Profile screen is displayed. Enter:

Table 29 (Page 1 of 2). Add SNA LU6.2 Mode Profile	
Field	**Description**
PROFILE name	The name of this profile. Normal practice is to give the profile the same name as the mode itself. LU62 is suggested.
MODE name	The name of the mode. This is usually some descriptive name such as LU62. The same name must be configured at the remote site as well.
Maximum number of SESSIONS	This field specifies the maximum number of conversations that can exist simultaneously on a connection using this mode. If you are connected to a NetView DM for MVS or a NetView DM/2 system, or if you specify no in the Parallel sessions supported? field (see Table 31 on page 148), this field must be set to 1.
	For connections to TME 10 Software Distribution, Version 3.1.5 for AIX and NetView DM for NetWare nodes, you can set this field to a higher number (and set Parallel sessions supported? to yes) to get better throughput. It would, however, be unusual to set this value higher than about 5, even on a connection with very high bandwidth.
Minimum contention WINNERS	The number of connections reserved for conversations initiated by your server. Set this field to 0 unless you have a specific reason to do otherwise.
Minimum contention LOSERS	The number of connections reserved for conversations initiated by the remote node. Set this field to 0 (zero) unless you have a specific reason to do otherwise.

Table 29 (Page 2 of 2). Add SNA LU6.2 Mode Profile

Field	Description
Auto activate limit	This field specifies the number of LU-LU sessions that will be activated for a particular local LU, partner LU, and mode combination after mode limits have been negotiated. It is suggested that you accept the default value of 0.
Upper bound for adaptive receive pacing window	Enter a value to specify the maximum number of request units that can be allocated to accommodate inbound session traffic. It is suggested that you accept the default value of 16.
Receive pacing window	Pacing controls the exchange of data on a session. Receive pacing restricts the amount of data that a remote node can send to your server without the server sending an acknowledgment. A low value limits throughput. A high value risks data being received faster than it can be processed. In practice a value of 7 is suitable. If throughput is an issue on a particular session, you can change this value.
Maximum RU size	This field specifies the maximum number of bytes in each request and response unit that are sent and received in sessions using this mode. It also specifies the preferred RU size for sessions using this mode. You should select the largest size that will work well on your physical network. For a LAN, select at least 2816. For a switched SDLC link, select 1024. If you are in any doubt, use the default value and adjust it if performance is a problem. Refer to "Logon Mode Table Definition" on page 170 for details of how this data is set at the NetView DM for MVS system.
Minimum RU size	Accept the default value of 256, which specifies the minimum size for request units that can be sent and received on sessions using this mode.
Class of Service (COS) name	Accept the default value of #CONNECT, which specifies a standard, nonsecure COS with medium transmission priority.

2. Press Enter to save changes.

3. Return to the LU6.2 menu.

Defining Local Transaction Program Names

The values you specifiy for some of the fields in this profile are different for STS and for SNA/DS communication.

For SNA/DS you must add the following two profiles:

NVDMSND   Defines the Send TP for TME 10 Software Distribution, Version 3.1.5 for AIX.

NVDMRCV   Defines the Receive TP for TME 10 Software Distribution, Version 3.1.5 for AIX.

For STS communication you define only one profile. It can be called fndrbas.

To add these profiles:

1. From the LU6.2 menu, select:

   → LU6.2 Transaction Program Name (TPN)
   → Add a Profile (*fastpath: _snatpnmk*)

   The Add SNA LU6.2 TPN Profile screen is displayed. Enter:

Table 30 (Page 1 of 2). Add SNA LU6.2 TPN Profile	
Field	**Description**
Profile name	Suggested values for SNA/DS are: NVDMSND, for Send TPN NVDMRCV, for Receive TPN The suggested value for STS is fndrbas.
Transaction program name	For SNA/DS, enter the following values for the hexadecimal representation of the transaction program name. 21F0F0F7, for Send TPN 21F0F0F8, for Receive TPN For STS communication, enter NVDM.
Transaction program name (TPN) is in hexadecimal?	For SNA/DS communication, select yes to indicate that the transaction program name that follows is in hexadecimal format. For STS communication, select no.
PIP data?	Select no to indicate that no program initialization parameters are used.
If yes, Subfields	This field only applies if "PIP data?" is set to yes. You do not need to set this field.

Field	Description
Table 30 (Page 2 of 2). Add SNA LU6.2 TPN Profile	
Conversation type	Toggle the value of this field to `Basic` to indicate that basic conversations should be used.
Sync level	Accept the default value of `none` which indicates that the transaction programs do not issue or respond to CONFIRM requests.
Resource security level	Accept the default value of `none` which indicates that not access-security information is required to access these transaction programs.
Full path to TP executable	For SNA/DS communication, set this field to: For Send TPN: `/usr/lpp/netviewdm/bin/fndts` For Receive TPN: `/usr/lpp/netviewdm/bin/fndtr` For STS communication, set this field to: `/usr/lpp/netviewdm/bin/fndrbas`
Multiple instances supported?	Select `yes` to allow multiple instances of the transaction program to run at once.
User ID	Set this field to the user instance number of root. This is `0` (zero) on all systems.
Server synonym name	This field is not used. You can leave it blank.
Restart action	Select `once` to indicate that the system resource controller should not restart the TPNs.
Communication type	Toggle the value of this field to `signals` to make the system resource controller use operating system signals to communicate with the TP.
If IPC, Communication IPC queue key	This field applies only if "Communication type" is set to IPC. You do not need to set it.
Standard input file/device	Use the default value `/dev/console`.
Standard output file/device	Use the default value `/dev/console`.
Standard error file/device	Use the default value `/dev/console`.
Comments	You can enter any comments that apply to the profiles.

2. Press Enter to save the information.

3. Return to the LU6.2 menu.

Defining the LU 6.2 Partner LU Profile

Perform the following steps to add this profile:

1. From the LU6.2 menu, select:

   ➡ LU 6.2 Partner LU

   ➡ Add a Profile (*fastpath: _snapartmk*)

   The Add SNA LU 6.2 Partner LU Profile is displayed. Enter:

Table 31. Add SNA LU 6.2 Partner LU Profile	
Field	**Description**
Profile name	It is suggested that you enter the same name given to the partner LU.
Fully qualified partner LU name	The first qualifier of this name must be the same as the VTAM NETID parameter, which corresponds to the network ID (see "Configuring NCP and VTAM" on page 161). The second qualifier is the name of the partner LU.
Partner LU alias	Enter the same name as the partner LU.
Parallel sessions supported?	Select a value to specify whether multiple concurrently active sessions are supported with the partner LU. Dependent LU 6.2 protocols do not support parallel sessions.
	yes Multiple concurrent sessions are supported with the partner LU.
	no Only one session can be active with the partner LU.
	If you are configuring connections with NetView DM for MVS or NetView DM/2, you must specify no.
All other fields	Accept the default values.

Defining the Partner LU 6.2 Location Profile

This profile is required only if the remote control point is an APPN Low Entry Node (LEN). If it is End Node (EN) or Network Node (NN) do not define it.

Perform the following steps to add this profile:

1. From the LU6.2 menu, select:

   ➡ Partner LU 6.2 Location

➡ Add a Profile (*fastpath: _snalocatmk*)

The Add Partner LU 6.2 Location Profile is displayed. Enter:

Table 32. Add Partner LU 6.2 Location Profile

Field	Description
Profile name	Enter the name of the partner LU.
Fully qualified partner LU name	The first qualifier of this name must be the same as the VTAM NETID parameter, which corresponds to the network ID (see "Configuring NCP and VTAM" on page 161). The second qualifier is the name of the partner LU.
Fully qualified owning Control Point name	The first qualifier of this name must be the same as the VTAM NETID parameter, which corresponds to the network ID. The second qualifier must be the same as the VTAM PU name for this node, or the SSCP_ID if the partner LU is NetView DM/MVS. See "Configuring NCP and VTAM" on page 161.
All other fields	Accept the default values.

Defining the LU 6.2 Side Information Profiles

The remote transaction programs for communication are defined to SNA Server/6000 Version 2.1 in their own profiles. The values you specifiy for some of the fields in this profile are different for STS and for SNA/DS communication.

For SNA/DS you must add the following two profiles:

NVDMSIDS   Defines the Send TP at the remote node.
NVDMSIDR   Defines the Receive TP at the remote node.

For STS communication, you define only one profile. It can be called SERVER01.

To add a profile (you must define two separate profiles for SNA/DS communication):

1. From the LU6.2 menu, select:

   ➡ LU 6.2 Side Information
   ➡ Add a Profile (*fastpath: _snasidemk*)

   The Add LU 6.2 Side Information Profile screen is displayed. Enter:

Configuring APPC Communication Using SNA Server/6000 Version 2.1

Table 33. Add LU 6.2 Side Information Profile	
Field	**Description**
PROFILE name	Suggested values for SNA/DS profiles are: `NVDMSIDS` for the send TPN `NVDMSIDR` for the receive TPN The suggested value for an STS profile is SERVER01.
Local LU or Control Point Alias	Enter the local LU name.
Partner LU alias	Leave blank
Fully qualified partner LU name	The first qualifier of this name must be the same as the VTAM NETID parameter, which corresponds to the network ID (see "Configuring NCP and VTAM" on page 161). The second qualifier is the name of the partner LU. This field is in alternative to the `Local LU alias` name, above. Enter the name in the "Mode" field of the LU 6.2 Mode Profile. See Table 29 on page 144.
Remote transaction program name (RTPN)	For SNA/DS profiles enter: `21F0F0F7`, for Send TPN `21F0F0F8`, for Receive TPN For an STS profile enter `NVDM`.
RTPN in hexadecimal?	For SNA/DS profiles specify `yes` which indicates that the value entered in the `Remote transaction program name` field above in is hexadecimal format. For STS profiles accept the default value of `no`, which indicates that the value is in ASCII format.
Comments	You can enter comments related to the profiles.

2. Press Enter to save changes.

3. Exit from SMIT.

Verifying a SNA Server/6000 Version 2.1 Configuration

In SNA Server/6000 Version 2.1, the profiles do not become effective until they have been successfully verified. Perform the following steps:

1. Go to the Advanced SNA Configuration menu in SMIT.

2. Select the **Verify Configuration Profiles** option.

3. Select `dynamic_update` in the `Update action of verification successful` field. (You can select `normal_update` if SNA Server/6000 Version 2.1 is not active.)

4. Press Enter to make the changes effective.

5. Press F3 to return to the input screen.

Configuring Communications Server, Version 5 on AIX Version 4.3.3

This section only refers to configuring APPC on a computer running AIX version 4.3.3 with Communications Server, Version 5; the APPC configuration for computers running AIX versions prior to version 4.3.3 with SNA Server/6000 Version 2.1 installed are described in "Configuring SNA Server/6000 Version 2.1 on AIX Versions Prior to Version 4.3.3" on page 128. It assumes that you have already installed Communications Server, Version 5 on your server.

You use SMIT to perform these configuration activities. You need to be logged on with root privileges to do so. SMIT provides a series of menus and panels for defining Communications Server, Version 5 *profiles*.

This section shows the profiles that must be created to configure Communications Server, Version 5 so that TME 10 Software Distribution, Version 3.1.5 for AIX can communicate with remote nodes using SNA/DS. APPC Communication can also be defined using STS;, either in a server to server environment or to enable APPC communication between server and client. Full details of the various profiles for SNA/DS and STS are given in the *TME 10 Software Distribution for AIX Installation Scenarios* manual, available in softcopy format on your product CD-ROM.

Each profile contains the configuration details for one logical component of the configuration. The field names in the profiles can easily be matched with the parameters on the panels of Communications Server, Version 5.

The profiles are samples which need to be customized for your specific circumstances. Where TME 10 Software Distribution, Version 3.1.5 for AIX requires a specific value in a parameter, a note following the profile explains the required value. Otherwise, you can accept the default values if you wish, or modify them to suit your requirements. A single example of each profile is provided, except in the case of the transaction programs, where both send and receive program definitions are given.

The profiles are contained in three files, as follows:

Node configurations in `sna_node_cfg`

- Configuration file information (details not shown here)
- Node definitions
- Ethernet data link control definitions
- Ethernet port definitions
- Ethernet link station definitions
- Partner logical unit definitions
- Local logical unit definitions
- Mode definitions

- Transaction program definitions (send and receive)
- Link station routing definitions

Transaction program location definitions in `sna_tps`

- Transaction program location definitions

Domain configurations in `sna_domn_cfg`

- Configuration file information (details not shown here)
- Set global log type (details not shown here)
- CPIC side information definitions

The panels defining these profiles can be accessed as follows:

From the System Management menu, select:

→ Communications applications and services
→ Communications Server for AIX
→ Configure SNA resources

> **Note:** Unlike SNA Server/6000 Version 2.1, you do not need to actively verify the profiles created, as Communications Server, Version 5 does it automatically for you.

Define Node

```
[define_node]
cp_alias = PPE0161
description = ""
fqcp_name = ITIBM0PC.PPE0161
node_type = END_NODE
mode_to_cos_map_supp = NO
mds_supported = YES
node_id = <05d00161>
max_locates = 1500
dir_cache_size = 255
max_dir_entries = 0

locate_timeout = 0
reg_with_nn = YES
reg_with_cds = YES
mds_send_alert_q_size = 100
cos_cache_size = 24
tree_cache_size = 40
tree_cache_use_limit = 40
max_tdm_nodes = 0
max_tdm_tgs = 0
max_isr_sessions = 1000
```

```
|                    isr_sessions_upper_threshold = 900
|                    isr_sessions_lower_threshold = 800
|                    isr_max_ru_size = 16384
|                    isr_rcv_pac_window = 8
|                    store_endpt_rscvs = NO
|                    store_isr_rscvs = NO
|                    store_dlur_rscvs = NO

|                    dlur_support = YES
|                    pu_conc_support = YES
|                    nn_rar = 128
|                    max_ls_exception_events = 0
|                    ms_support = NORMAL
|                    queue_nmvts = NO
|                    send_term_self = NO
|                    ptf_flags = NONE
```

| ### Define Ethernet Data Link Control

```
|                    [define_ethernet_dlc]
|                    dlc_name = ETHER0
|                    description = ""
|                    neg_ls_supp = YES
|                    card_type = GDLC_ETHERNET
|                    initially_active = NO
|                    adapter_number = 0
|                    max_saps = 16
|                    ethernet_type = 802_3
```

| **Note:** adapter_number must be set to the last digit of the adapter name (normally
| zero, of there is only one adapter).

| ### Define Ethernet Port

```
|                    [define_ethernet_port]
|                    port_name = ETSAP0
|                    description = ""
|                    dlc_name = ETHER0
|                    port_type = PORT_SATF
|                    port_number = 0
|                    lsap_address = 0x04
|                    initially_active = YES
|                    implicit_hpr_support = YES
|                    implicit_link_lvl_error = NO

|                    max_rcv_btu_size = 1492
|                    tot_link_act_lim = 255
|                    inb_link_act_lim = 0
|                    out_link_act_lim = 0
|                    ls_role = LS_NEG
|                    implicit_dspu_services = NONE
|                    implicit_dspu_template = ""
|                    implicit_ls_limit = 0
|                    act_xid_exchange_limit = 9
|                    nonact_xid_exchange_limit = 5
```

```
|                      ls_xmit_rcv_cap = LS_TWS
|                      max_ifrm_rcvd = 7
|                      target_pacing_count = 7
|                      max_send_btu_size = 1492
|                      mac_address = <0004ac975928>
|                      implicit_cp_cp_sess_support = YES
|                      implicit_limited_resource = NO
|                      implicit_deact_timer = 30
|                      effect_cap = 3993600
|                      connect_cost = 0
|                      byte_cost = 0
|
|                      security = SEC_NONSECURE
|                      prop_delay = PROP_DELAY_LAN
|                      user_def_parm_1 = 128
|                      user_def_parm_2 = 128
|                      user_def_parm_3 = 128
|                      local_name = ""
|                      xid_timeout = 8
|                      xid_retry_limit = 2
|                      t1_timeout = 50
|                      t1_retry_limit = 2
|                      ack_time = 1
|                      inact_time = 48
|                      force_time = 120
```

| ### Define Ethernet Link Station

```
|                      [define_ethernet_ls]
|                      ls_name = ETHL0
|                      description = ""
|                      port_name = ETSAP0
|                      adj_cp_name = ITIBM0PC.MVSESA31
|                      adj_cp_type = NETWORK_NODE
|                      dspu_services = NONE
|                      dspu_name = <0000000000000000>
|                      dlus_name = <00000000000000000000000000000000000>
|                      bkup_dlus_name = <00000000000000000000000000000000000>
|                      local_node_id = <00000000>
|                      adj_node_id = <00000000>
|                      mac_address = <0200694edd02>
|
|                      lsap_address = 0x04
|                      max_send_btu_size = 1492
|                      ls_attributes = SNA
|                      cp_cp_sess_support = YES
|                      hpr_supported = YES
|                      hpr_link_lvl_error = YES
|                      auto_act_supp = NO
|                      tg_number = 0
|                      limited_resource = NO
|                      solicit_sscp_sessions = NO
|                      pu_name = <0000000000000000>
```

```
disable_remote_act = NO
default_nn_server = YES
link_deact_timer = 30
use_default_tg_chars = YES
effect_cap = 3993600
connect_cost = 0
byte_cost = 0
security = SEC_NONSECURE
prop_delay = PROP_DELAY_LAN
user_def_parm_1 = 128
user_def_parm_2 = 128
user_def_parm_3 = 128

target_pacing_count = 7
ls_role = LS_NEG
max_ifrm_rcvd = 0
dlus_retry_timeout = 0
dlus_retry_limit = 0
need_vrfy_fixup = NO
initially_active = NO
restart_on_normal_deact = NO
react_timer = 30
react_timer_retry = 65535

xid_timeout = 8
xid_retry_limit = 2
t1_timeout = 8
t1_retry_limit = 2
ack_time = 1
inact_time = 48
force_time = 120
```

Define Partner Logical Unit
There should be a Partner LU for every connection with another server using SNA.

```
[define_partner_lu]
plu_alias = LT0182A0
description = ""
fqplu_name = ITIBM0PC.LT0182A0
plu_un_name = LT0182A0
parallel_sess_supp = YES
appcip_routing_preference = NATIVE
max_mc_ll_send_size = 0
conv_security_ver = NO
```

Define Local Logical Unit

```
[define_local_lu]
lu_alias = LT0161A0
list_name = ""
description = ""
lu_name = LT0161A0
lu_session_limit = 0
pu_name = <0000000000000000>
nau_address = 0
default_pool = NO
syncpt_support = NO

lu_attributes = NONE
sscp_id = 0
disable = NO
sys_name = ""
timeout = 60
back_level = NO
```

Define Mode

```
[define_mode]
mode_name = LU62
description = ""
max_neg_sess_lim = 32767
plu_mode_session_limit = 2
min_conwin_src = 1
min_conloser_src = 0

auto_act = 0
receive_pacing_win = 4
max_receive_pacing_win = 0
default_ru_size = YES
max_ru_size_upp = 1024
max_ru_size_low = 0
cos_name = #CONNECT
```

Define Transaction Program

Two transaction programs must be defined: one for the SEND transactions and one for the RECEIVE transactions.

```
|                    [define_tp]
|                    tp_name = <21f0f0f7>
|                    description = ""
|                    list_name = ""
|                    conv_type = BASIC
|                    security_rqd = NO
|                    sync_level = NONE
|                    enabled = YES
|                    pip_allowed = NO
|                    tp_instance_limit = 0
|                    incoming_alloc_timeout = 60
|                    secondary_key = NVDMSND

|                    [define_tp]
|                    tp_name = <21f0f0f8>
|                    description = ""
|                    list_name = ""
|                    conv_type = BASIC
|                    security_rqd = NO
|                    sync_level = NONE
|                    enabled = YES
|                    pip_allowed = NO
|                    tp_instance_limit = 0
|                    incoming_alloc_timeout = 60
|                    secondary_key = NVDMRCV
```

| **Define Link Station Routing**

```
|                    [define_ls_routing]
|                    lu_name = LT0161A0
|                    fq_partner_lu = ITIBM0PC.PPE0014
|                    wildcard_fqplu = NO
|                    description = ""
|                    ls_name = ETHL0

|                    [define_ls_routing]
|                    lu_name = LT0161A0
|                    fq_partner_lu = ITIBM0PC.ND24TCPB
|                    wildcard_fqplu = NO
|                    description = ""
|                    ls_name = ETHL0
```

Define Transaction Program Location

A transaction program location should be defined for each of the two transaction programs.

```
[<21f0f0f7>]
LUALIAS = ""
DESCRIPTION = ""
USERID = root
GROUP = ""
TIMEOUT = -1
TYPE = NON-QUEUED
STYLE = COMPATIBLE
PATH = /usr/lpp/netviewdm/bin/fndts
STDIN = /dev/console
STDOUT = /dev/console
STDERR = /dev/console


[<21f0f0f8>]
LUALIAS = ""
DESCRIPTION = ""
USERID = root
GROUP = ""
TIMEOUT = -1
TYPE = NON-QUEUED
STYLE = COMPATIBLE
PATH = /usr/lpp/netviewdm/bin/fndtr
STDIN = /dev/console
STDOUT = /dev/console
STDERR = /dev/console
```

Define CPIC Side Information

You need to define CPIC side information for the SEND and RECEIVE transaction programs for each server with which you want to communicate using a SNA/DS connection.

For the first or only server you would normally use the default symbolic destination names of NVDMSDS (send) and NVDMSIDR (receive), that are entered in the TME 10 Software Distribution for AIX configuration of the SNA/DS connection. For any other servers you should create your own unique symbolic destination names.

```
|              [define_cpic_side_info]
|              sym_dest_name = NVDMSIDR
|              description = ""
|              lu_alias = LT0161A0
|              partner_lu_name = ITIBM0PC.LT0182A0
|              mode_name = LU62
|              tp_name_type = SNA_SERVICE_TP
|              tp_name = 21F0F0F8
|              conversation_security_type = NONE
|              security_user_id = ""
|              security_password = ""


|              [define_cpic_side_info]
|              sym_dest_name = NVDMSIDS
|              description = ""
|              lu_alias = LT0161A0
|              partner_lu_name = ITIBM0PC.LT0182A0
|              mode_name = LU62
|              tp_name_type = SNA_SERVICE_TP
|              tp_name = 21F0F0F7
|              conversation_security_type = NONE
|              security_user_id = ""
|              security_password = ""
```

Starting SNA Server/6000 Version 2.1 or Communications Server, Version 5 Automatically

TME 10 Software Distribution, Version 3.1.5 for AIX starts automatically when your computer is restarted. If you have the communication option installed and are using it, you need to start SNA Server/6000 Version 2.1 or Communications Server, Version 5, as appropriate, and its link stations at the same time.

When SNA Server/6000 Version 2.1 or Communications Server, Version 5 is installed, the necessary statements are added automatically to the /etc/inittab file. Make sure that the file contains these lines:

```
rcsna:2:2wait:/etc/rc.sna >/dev/console 2>&1 # Start sna daemons
 .
 .
 .
NetViewDM/6000:2:wait: /etc/rc.ndm
```

Figure 20. Part of /etc/inittab

In addition, when you define link stations at a workstation, set the Activate link station at SNA startup? parameter to Yes.

You may also need to edit the file /etc/rc.sna. You need root privileges to do so. The comments in this file tell you what to do. The following line should be

| uncommented so that SNA Server/6000 Version 2.1 or Communications Server,
| Version 5 can start at startup time:

| `#/usr/bin/sna-start`

Chapter 16. Configuring VTAM and NetView DM for MVS

This chapter describes the configuration procedures you must perform to enable communication between TME 10 Software Distribution, Version 3.1.5 for AIX and NetView DM for MVS running under MVS. NetView DM for MVS must be Release 6 or higher. You must perform configuration whether TME 10 Software Distribution, Version 3.1.5 for AIX communicates with NetView DM for MVS directly or through an intermediate node such as NetView DM/2.

Configuring NCP and VTAM

To define a TME 10 Software Distribution, Version 3.1.5 for AIX node to VTAM, you need to work with your NetView DM for MVS system administrators to agree on naming conventions. You need to:

- Define the TME 10 Software Distribution, Version 3.1.5 for AIX physical and logical units (see "Physical and Logical Unit Definitions").

- Define the parameters in a logon mode table for the sessions between the NetView DM for MVS system and TME 10 Software Distribution, Version 3.1.5 for AIX (see "Logon Mode Table Definition" on page 170).

Examples are shown in the following section, but they do not provide an exhaustive definition of how to configure VTAM and NCP. The explanations of macros and operands are provided for your convenience. Consult your *VTAM Resource Definition Reference* where necessary.

Physical and Logical Unit Definitions

Each TME 10 Software Distribution, Version 3.1.5 for AIX that is connected *directly* to NetView DM for MVS (that is, not through an intermediate node) requires an independent LU to be defined to VTAM and also to NCP if it is being used. A physical unit is also required for each adjacent NetView DM node; that is, each adjacent instance of TME 10 Software Distribution, Version 3.1.5 for AIX and each intermediate node that provides a connection to a TME 10 Software Distribution, Version 3.1.5 for AIX node.

The following sections provide three examples of configuring direct connections. They are for the following configurations:

- Connection to a 3745 attached to the TME 10 Software Distribution, Version 3.1.5 for AIX server ring
- Connection through a token-ring gateway
- Direct connection using an SDLC link server

In addition, an example of a logon mode table is shown. You can use this table for any of the three sample configurations.

PU and LU Definition for a Connection to a 3745

This section gives the NCP and VTAM definitions necessary for TME 10 Software Distribution, Version 3.1.5 for AIX to communicate with NetView DM for MVS through a 3745 gateway. See Figure 21.
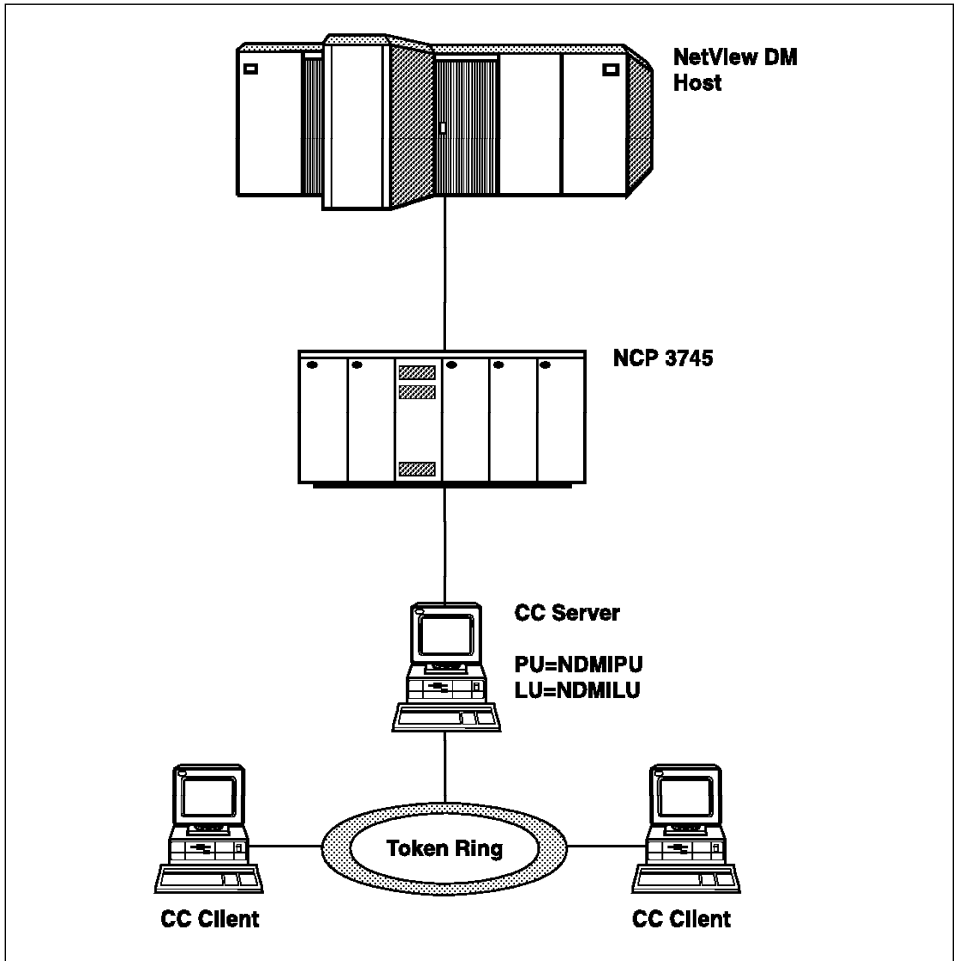

Figure 21. TME 10 Software Distribution, Version 3.1.5 for AIX and NetView DM for MVS using a direct link and a 3745

NCP Definition for Token-Ring Connection: The following NCP statements show the NCP macros used to define the connectivity to the token ring and the PU/LU pairs required to support devices on a switched line. (TME 10 Software Distribution, Version 3.1.5 for AIX servers on a LAN token ring are supported as devices on a switched line.)

```
TO27TRPG GROUP ECLTYPE=PHYSICAL
TO27TRL1 LINE  ADDRESS=(1092,FULL),LOCADD=400037271092,PORTADD=1
               RCVBUFC=4095,MAXTSL=1108,TRSPEED=16,ADAPTER=TIC2
TO27TRP1 PU    ADDR=01,PUDR=NO,ANS=CONT
LUTIC2   LU    LOCADDR=0,ISTATUS=INACTIVE
TO27TRG1 GROUP ECLTYPE=LOGICAL,AUTOGEN=24,PHYPORT=1,CALL=INOUT
```

Figure 22. NCP macros for Token-Ring connectivity

VTAM Switched Major Node Definitions: The following statements show the VTAM macros used to define the switched major node containing the PU and LU statements for the TME 10 Software Distribution, Version 3.1.5 for AIX server.

Note: LOCADDR=0 indicates an independent LU.

```
DIALNMD6 VBUILD TYPE=SWNET,MAXGRP=2,MAXNO=2

NDM6PU    PU ADDR=C1,IDBLK=071,IDNUM=00013,PUTYPE=2,ISTATUS=ACTIVE,
             DLOGMOD=NVDMNORM,USSTAB=TPOUSS,MAXOUT=7,MAXDATA=265,    *
             PACING=0,VPACING=0,ANS=CONT,MODETAB=NDMLU62P

NDM6LU    LU LOCADDR=0,MODETAB=NDMLU62P,DLOGMOD=NVDMNORM,
             ISTATUS=ACTIVE
```

Figure 23. VTAM macros for TME 10 Software Distribution, Version 3.1.5 for AIX connectivity

TME 10 Software Distribution, Version 3.1.5 for AIX Server Definitions: Use this information in your procedure to customize NetView DM for MVS, TME 10 Software Distribution, Version 3.1.5 for AIX, and SNA Server/6000 Version 2.1 or Communications Server, Version 5 for your server.

VTAM Parameter	SNA Server/6000 Version 2.1 Parameter	Communications Server, Version 5 Parameter
NETID=NETWK1	Network name	Network name
IDNUM=00013	Node ID	Node ID
DLOGMOD=NVDMNORM	Mode name	Mode name
PU=NDM6PU	PU name	PU name
LU=NDM6LU	LU name for TME 10 Software Distribution, Version 3.1.5 for AIX	LU name for TME 10 Software Distribution, Version 3.1.5 for AIX
LOCADDR=0	Indicates that this in an independent LU	Indicates that this in an independent LU
LOCADD=400037271092 (NCP parameter)	The address that is specified in the SNA Server/6000 Version 2.1 link definition.	The address that is specified in the Communications Server, Version 5 link definition.

Connection through a Token-Ring Gateway

This section gives the 3174 and NCP/VTAM definitions necessary for
TME 10 Software Distribution, Version 3.1.5 for AIX to communicate with NetView DM
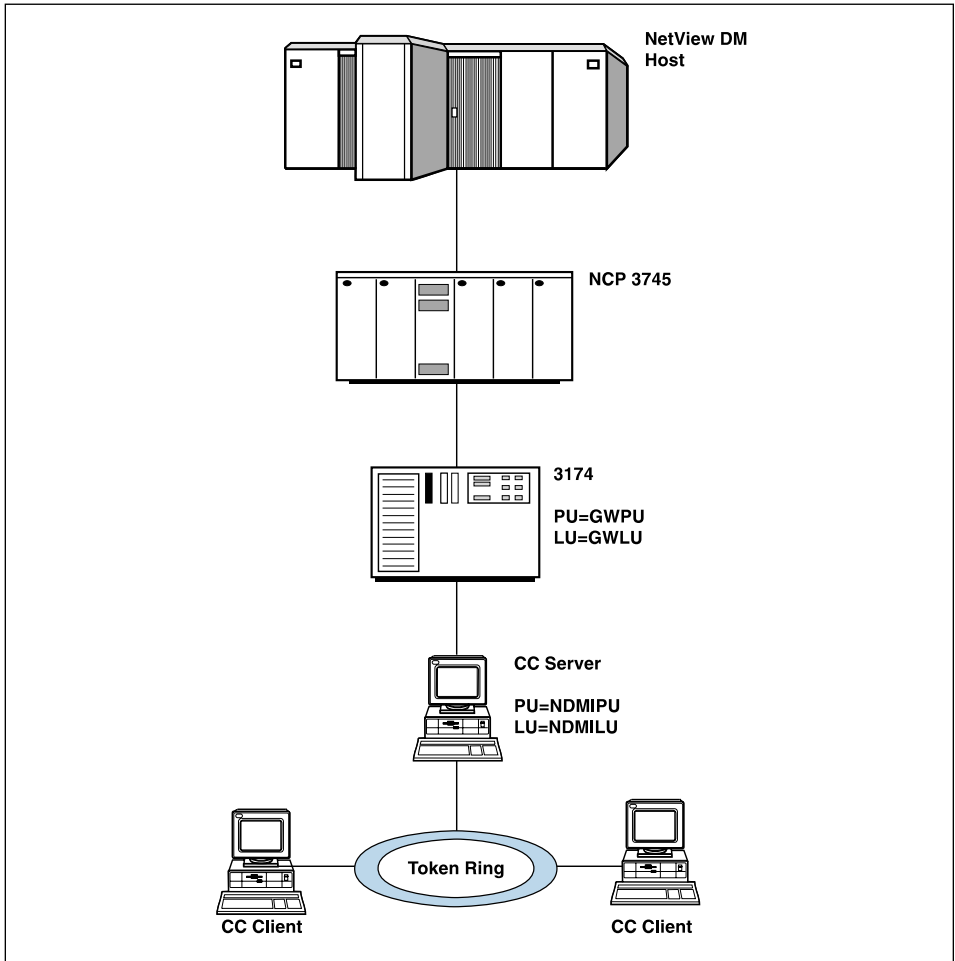for MVS through a token-ring gateway. See Figure 24.

Figure 24. TME 10 Software Distribution, Version 3.1.5 for AIX and NetView DM for MVS using a Token-Ring gateway

The gateway can be any proprietary software or it can be a 3174. Instructions for
configuring the gateway are not given here; refer to the documentation provided with
the gateway.

The example in Figure 24 is for a 3174 gateway that does not support APPN* (for
example, a 3174 Model 90R). The configuration details are considerably different if
your gateway supports APPN (for example, a 3174 with the APPN feature). The
principal difference is that, using APPN, the server resources appear to NetView DM for

MVS to be situated at the gateway. This means that only one PU is defined (that of the gateway) and the LUs for access to the server are listed as belonging to that PU.

Many gateways adopt the same approach as a 3174 with the APPN feature. That is, you only define a PU for the gateway and you define LUs for accessing the TME 10 Software Distribution, Version 3.1.5 for AIX server under the definition of that PU. The difference is that a 3174 is a controller for multiple downstream PUs, whereas a gateway simply bridges between network types and relays the SNA messages.

You must decide which type of gateway you have and configure NCP and VTAM accordingly.

NCP/VTAM Definition: Figure 25 on page 166 shows sample NCP/VTAM definitions. They refer to the network shown in Figure 24 on page 164.

Configuring VTAM and NetView DM for MVS

```
GSDLC    GROUP CLOCKNG=EXT,
               DIAL=NO,
               LNCTL=SDLC,
               REPLYTO=0.5,
               RETRIES=(19,4,5),
               TYPE=NCP

L002     LINE  ADDRESS=002,
               DUPLEX=FULL,
               ETRATIO=25,
               SPEED=9600,
               ISTATUS=ACTIVE,
               RETRIES=(7,2,2)

GWPU     PU    ADDR=C0,
               ISTATUS=ACTIVE,
               PACING=(1),
               PUDR=YES,
               PUTYPE=2,
               DISCNT=(NO)

GWLU     LU    LOCADDR=1,
               ISTATUS=ACTIVE,
               MODETAB=TPOMODE,
               DLOGMOD=SD82

NDM6PU   PU    ADDR=C1,
               MAXDATA=265,
               MAXOUT=7,
               PASSLIM=8,
               PUTYPE=2,
               SSCPFM=USSSCS,
               XID=YES

NDM6LU   LU    LOCADDR=0,
               ISTATUS=ACTIVE,
               MODETAB=NDMLU62P,
               DLOGMOD=NVDMNORM,
               RESSCB=2,
               PACING=1
```

Figure 25. VTAM and NCP macros for connectivity through a gateway

The definition in Figure 25 contains the following:

- A PU for the gateway (GWPU)

- An LU for 3270 use at the gateway (GWLU)

- A PU for the TME 10 Software Distribution, Version 3.1.5 for AIX server using the *same line* as that used to reach the gateway (NDM6PU)

- An independent LUs for the server (NDM6LU)

  Note that this LU is defined under the PU for the TME 10 Software Distribution, Version 3.1.5 for AIX server. The LU has LOCADDR=0 to indicate an independent LU.

TME 10 Software Distribution, Version 3.1.5 for AIX Server Definitions: This is
the information that you need to use in your procedure to customize NetView DM for
MVS, TME 10 Software Distribution, Version 3.1.5 for AIX, and SNA Server/6000
Version 2.1 or Communications Server, Version 5 for your server:

VTAM Parameter	SNA Server/6000 Version 2.1 Parameter	Communications Server, Version 5 Parameter
NETID=NETWK1	Network name	Network name
DLOGMOD=NVDMNORM	Mode name	Mode name
PU=NDM6PU	PU name	PU name
LU=NDM6LU	LU name for TME 10 Software Distribution, Version 3.1.5 for AIX	LU name for TME 10 Software Distribution, Version 3.1.5 for AIX
ADDR=C1	Local station address	Local station address

Direct Connection Using an SDLC Link: This section gives examples of the NCP
and VTAM definitions necessary for TME 10 Software Distribution, Version 3.1.5 for
AIX to communicate with NetView DM for MVS over a direct SDLC link.
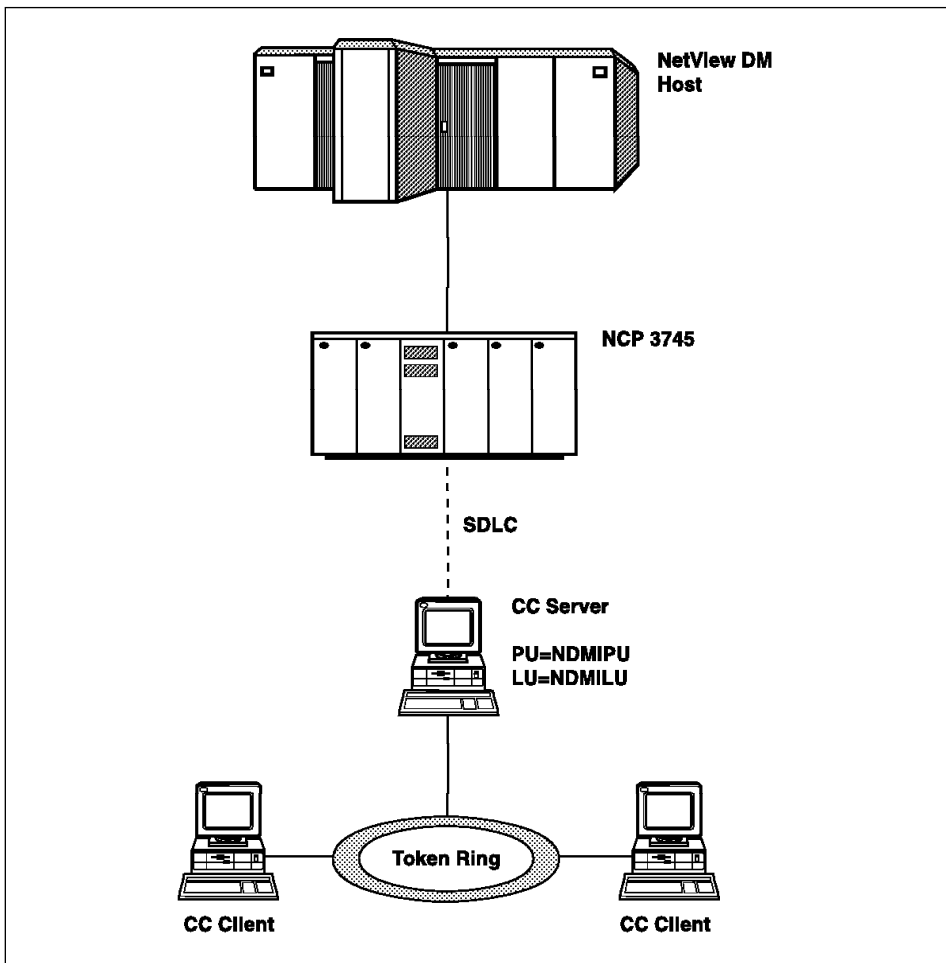
Configuring VTAM and NetView DM for MVS

Figure 26. TME 10 Software Distribution, Version 3.1.5 for AIX and NetView DM using an SDLC link

NCP/VTAM Definition: Figure 27 shows sample NCP/VTAM definitions. They refer to the network shown in Figure 26.

```
GSDLC     GROUP CLOCKNG=EXT,
                DIAL=NO,
                LNCTL=SDLC,
                REPLYTO=0.5,
                RETRIES=(19,4,5),
                TYPE=NCP

L002      LINE  ADDRESS=002,
                DUPLEX=FULL,
                ETRATIO=25,
                SPEED=9600,
                ISTATUS=ACTIVE,
                RETRIES=(7,2,2)

NDM6PU    PU    ADDR=C1,
                MAXDATA=265,
                MAXOUT=7,
                PASSLIM=8,
                PUTYPE=2,
                SSCPFM=USSSCS,
                XID=YES

NDM6LU    LU    LOCADDR=0,
                ISTATUS=ACTIVE,
                MODETAB=NDMLU62P,
                DLOGMOD=NVDMNORM,
                RESSCB=2,
                PACING=1
```

Figure 27. VTAM and NCP macros for connectivity using an SDLC link

The definition in Figure 27 contains the following:

1. A direct attachment on line L002 to the PU for the TME 10 Software Distribution, Version 3.1.5 for AIX server (NDM6PU)

2. An independent LU (LOCADDR=0) for access to the TME 10 Software Distribution, Version 3.1.5 for AIX server defined directly under the PU (NDM6LU).

TME 10 Software Distribution, Version 3.1.5 for AIX Server Definitions: This is the information that you use in your procedure to customize NetView DM for MVS, TME 10 Software Distribution, Version 3.1.5 for AIX, and SNA Server/6000 Version 2.1 or Communications Server, Version 5 for your server:

VTAM Parameter	SNA Server/6000 Version 2.1 Parameter	Communications Server, Version 5 Parameter
NETID=NETWK1	Network name	Network name
DLOGMOD=NVDMNORM	Mode name	Mode name
PU=NDM6PU	PU name	PU name
LU=NDM6LU	LU name for TME 10 Software Distribution, Version 3.1.5 for AIX	LU name for TME 10 Software Distribution, Version 3.1.5 for AIX

VTAM Parameter	SNA Server/6000 Version 2.1 Parameter	Communications Server, Version 5 Parameter
ADDR=C1	Local station address	Local station address

Logon Mode Table Definition

The logon mode table is used to define the parameters for sessions between NetView DM for MVS and TME 10 Software Distribution, Version 3.1.5 for AIX. This table defines the logon mode NDMLU62P that is referred to in all the previous examples.

```
MODETAB

NDMLU62P MODEENT LOGMODE=NVDMNORM,
              FMPROF=X'13',
              TSPROF=X'07',
              PRIPROT=X'B0',
              SECPROT=X'B0:',
              COMPROT=X'50A1',
              ENCR=B'0000',
              RUSIZES=X'8585',
              PSNDPAC=X'03',
              SRCVPAC=X'03',
              SSNDPAC='00',
              PSERVIC=X'0602000000000000000002400',
              TYPE=X'0',
              COS=COSNAME
         MODEEND
```

Figure 28. Logon mode table

The following parameters are specific for defining the LU session between NetView DM for MVS and TME 10 Software Distribution, Version 3.1.5 for AIX:

Parameter	Description
LOGMODE	Specifies the entry name that is used to point to the set of session parameters in this logon mode table.
	You can use any entry name with up to eight characters; however, this must match the mode name specified within the configuration for SNA Server/6000 Version 2.1 or Communications Server, Version 5.
FMPROF	Specifies the function management profile for this entry. Set FMPROF to X'13', which indicates that Function Management Profile 19 (X'13') rules are to be used for these LU 6.2 sessions. See *System Network Architecture Formats*, GA27-3136.

Parameter	Description
TSPROF	Specifies the transmission services profile for this entry. Set TSPROF to X'07', which indicates that Transmission Services Profile 7 (X'07') rules are to be used for these LU 6.2 sessions. See *System Network Architecture Formats*, GA27-3136.
PRIPROT	Specifies the primary LU protocols for this entry. Set PRIPROT to X'B0', which indicates the following protocols are to be used: • Multiple RU chaining • Immediate request mode • Definite or exception response.
SECPROT	Specifies the secondary LU protocols for this entry. Set SECPROT to X'B0'.
COMPROT	Specifies the common LU protocols for this entry. Set COMPROT to X'50A1', which indicates that the following protocols are to be used: • Segmenting • FMH allowed • Brackets used • CEB used • No alternate code set • BIND RSP not held • HDX-FF • Symmetric recovery • SLU=winner • HDX-FF reset is SEND for PLU and RCV for SLU
ENCR	Specifies the type of cryptography to be used with the VTAM data encryption facility. Set ENCR to B'B0000', that is, no cryptography because VTAM does not support encryption for LU 6.2 sessions.

Logon Mode Table Definition

Parameter	Description
RUSIZES	Specifies the maximum length of data (request units or RU) in bytes that can be sent by the primary LU and the secondary LU when they are in session with each other.
	RUSIZES is represented by four hexadecimal digits. The two leftmost digits apply to the secondary LU and the two rightmost digits apply to the primary LU. The format is the same for both sets of digits: in each set, the first digit is the mantissa (m) and the second digit is the exponent (n) in the formula $m \times 2^{**n}$. The mantissa must be in the range X'8'–X'F'. The exponent must be in the range X'0'–X'F'.
	This formula is then used to calculate the maximum RU sizes that can be sent by the primary or secondary LU. For example, RUSIZES=X'858C' specifies that the secondary LU can send an RU of maximum length $8 \times 2^{**5}$ (or 256) bytes and the primary LU can send an RU of maximum length $8 \times 2^{**C}$ (or 32768) bytes.
	TME 10 Software Distribution, Version 3.1.5 for AIX supports all RU sizes. However, you should note the following guidelines when setting up the logon mode table and when configuring SNA Server/6000 Version 2.1 or Communications Server, Version 5 (see "Defining Conversation Modes" on page 143).
	• Small RUs make poor use of your data transfer mechanism. Transfer times, especially for large files, are considerably increased by using small RUs. The smallest RU size that you should consider is 256 bytes.
	• Large RUs can cause saturation at the receiver if the data cannot be processed as fast as it is received.
	• Some SNA connections, especially slow ones like public telecommunication lines, are better suited to small RUs (for example 256 or 512 bytes).
	• SNA Server/6000 Version 2.1 and Communications Server, Version 5 support any legal RU size between 256 and 3840 bytes. Any BIND proposing a size smaller than 256 is rejected by SNA Server/6000 Version 2.1 or Communications Server, Version 5; any BIND proposing a size greater than that configured is negotiated downward.

Parameter	Description
RUSIZES (cont.)	Therefore, RUSIZES is best set to X'F8F8' (4KB) maximum and X'8585' (256 bytes) minimum. For a TME 10 Software Distribution, Version 3.1.5 for AIX node connected directly through a token ring to a channel-attached NCP, an RU size of 3840 is expected to be optimal. For other attachments, link speeds and connectivity must be considered when you are selecting the optimal value.
	NetView DM for MVS does not check the RUSIZES value for NetView DM for MVS-to-TME 10 Software Distribution, Version 3.1.5 for AIX sessions. In the sample logmode definition, RUSIZES is set to X'8585' (256 bytes) for NetView DM for MVS sessions. These values should be consistent with those defined to the SNA Server/6000 Version 2.1 or Communications Server, Version 5 (see "Defining Conversation Modes" on page 143).
PSNDPAC	Specifies the primary send pacing count. This value is not checked by TME 10 Software Distribution, Version 3.1.5 for AIX. If PSNDPAC is omitted, PSNDPAC=X'00' is the default.
SRCVPAC	Specifies the secondary receive pacing count. If SRCVPAC is omitted, SRCVPAC=X'00' is the default.
	SRCVPAC must be appropriate to the RU size selected for the primary LU, and is determined by the formula:
	$(2n - 1) \times$ primary send RU size <= 4KB
	where n is the SRCVPAC value.
	For example, SRCVPAC=X'01' if the two rightmost digits of RUSIZES are X'F7'. If the RUSIZES chosen is X'8585', SRCVPAC can be as high as X'08'.
	The value of SRCVPAC greatly influences throughput. Higher values of this parameter ensure better throughput. You should set SRCVPAC as high as the calculation above allows.
	Note that SRCVPAC can be negotiated downward but never upward.
SSNDPAC	Specifies the secondary send pacing count. TME 10 Software Distribution, Version 3.1.5 for AIX has no dependencies on this value. It is used as specified. If SSNDPAC is omitted, SSNDPAC=X'00' is the default.

Parameter	Description
PSERVIC	Specifies the presentation services profile for this entry. Set PSERVIC to X'060200000000000000002400' which indicates that the following will be used: • LU 6.2 • No attach security • Sync level=confirm • PLU reinitiates • Parallel sessions are not supported.
TYPE	Specifies the type of BIND command for this entry. Set TYPE to X'0', which means that the secondary LU can support a negotiable BIND.
COS	Specifies the name of an entry in a class of service table to be used for sessions established with this logon mode. Because it is a *batch* mode, you should use an entry that specifies low-priority virtual routes so as not to interfere with interactive traffic.

Configuring NetView DM for MVS

NetView DM for MVS requires two specific customization steps to manage TME 10 Software Distribution, Version 3.1.5 for AIX nodes.

1. Node types with change management entry point (CMEP) functional capabilities have to be defined for:

   • TME 10 Software Distribution, Version 3.1.5 for AIX server and single nodes
   • TME 10 Software Distribution, Version 3.1.5 for AIX clients

2. When the host product is in a condition to manage this type of node, the specific network definition has to be prepared for:

   • Directly connected nodes
   • Indirectly connected nodes

These steps are described in the following sections.

Node Type Definition

This operation belongs to the product customization phase. If a node type with CMEP functional capabilities has not been defined, then you must perform a new run of the installation macros.

Prepare the following macros for the NetView DM for MVS stage 1 installation job:

• NDMNODE macro for a single node or a server
• NDMNODE macro for a client
• NDMTCP macro

- NDMCP macro
- A transmission profile

For each node type, prepare a macro defining its characteristics.

In NetView DM for MVS, specify each node type and declare it to have change management entry point (CMEP) functional capabilities.

In the examples in Figure 29 and Figure 30 on page 176, the node types have been assigned the following names:

NDM6   TME 10 Software Distribution, Version 3.1.5 for AIX node with single node or server functions.

NDMC   A client.

Now prepare the following macros:

1. An NDMNODE macro to define a node type with CMEP functional capabilities for TME 10 Software Distribution, Version 3.1.5 for AIX nodes (see Figure 29).

```
NDMNODE TYPE=NDM6,
     LOGM=NVDMNORM,FUNC=CMEP,
     XMFUNC=(SEND, RETR, DELE),
     RESTYPE=(0060,0070,0080,0100,0120,0220,0230,0240,0250)
```

Figure 29. Sample NDMNODE macro for TME 10 Software Distribution, Version 3.1.5 for AIX Base and Server

The variables defined are as follows:

Parameter	Description
LOGM	This value should correspond to the name of a logon mode table that defines the session parameters for communication between TME 10 Software Distribution, Version 3.1.5 for AIX and NetView DM for MVS. (Refer to the example in Figure 28 on page 170.)
FUNC	The functional capability must be defined as CMEP. A CMEP node can manage changes to itself and has limited ability to manage changes to other nodes.

Configuring NetView DM for MVS

Parameter	Description
XMFUNC	This can be set to (SEND, RETR, DELE) for a TME 10 Software Distribution, Version 3.1.5 for AIX node.
	This parameter defines the transmission-function authorization parameters. That is, it defines those functions that the TME 10 Software Distribution, Version 3.1.5 for AIX nodes are allowed to initiate against NetView DM for MVS. TME 10 Software Distribution, Version 3.1.5 for AIX nodes support sending, retrieving, and deleting files but cannot initiate change control commands.
RESTYPE	This parameter gives the data object classifications that the TME 10 Software Distribution, Version 3.1.5 for AIX node is authorized to work with. TME 10 Software Distribution, Version 3.1.5 for AIX can work with all of the defined resource types, so you should list them all unless you have some reason to limit the types of files that you want a particular node to have permission to transfer.
	This parameter must be present if the XMFUNC parameter is specified. Refer to the *NetView DM for MVS Installation and Customization Guide* for possible values of the codes.
SFUNC	This parameter lists the functions that may be issued to the TME 10 Software Distribution, Version 3.1.5 for AIX node. The default is the full set (SEND, RETR, DELE, REMO, ACTI, INIT, ACCE, INST, UNIN).
	This parameter need not be specified as TME 10 Software Distribution, Version 3.1.5 for AIX supports all of the functions. To limit the functions that can be issued by NetView DM for MVS to a particular node, you must use this parameter and specify only those functions that are allowed.

2. An NDMNODE macro to define a node type for TME 10 Software Distribution, Version 3.1.5 for AIX clients (see Figure 30).

```
NDMNODE TYPE=NDMC,
     LOGM=LU62,FUNC=CMEP,
     XMFUNC=(SEND, RETR),
     RESTYPE=(0060,0070,0080,0100,0120,0220,0230,0240,0250)
```

Figure 30. Sample NDMNODE macro for a TME 10 Software Distribution, Version 3.1.5 for AIX Client

Note that this definition is identical to that for the single node-server node. TME 10 Software Distribution, Version 3.1.5 for AIX clients can send, retrieve, and delete files of all data object classifications. They can also support all distribution and change control functions that the NetView DM for MVS can issue.

To limit the permitted functions for a specific client, use the XMFUNC, RESTYPE, and SFUNC parameters as described for the single and server nodes.

3. An NDMTCP macro specifically for the TME 10 Software Distribution, Version 3.1.5 for AIX application. A sample macro highlighting the lines that are significant for TME 10 Software Distribution, Version 3.1.5 for AIX is shown in Figure 31.

```
NDMTCP APPLID=(RAKADT03,*),
       IAPPLID=(RAKADI03,*),
       IPLUNAM=NONE,    (LU NAME OF DEFAULT IOF PRINTER)
       OPCTL=NETV,      (SELECTED MESSAGES ARE ROUTED TO NV)
       ROUTCD1=2,
       ROUTCD2=2,
       DSCD1=6,
       DSCD2=6,
       RESWAIT=300,     (SECONDS NDM WAITS FOR LU62 RESPONSE)
       STALINE=1,       (DEFAULT IS NO)
       AUTOSTR=NO,      (NO IS THE DEFAULT)
       AUTOEND=NO,      (NO IS THE DEFAULT)
       RETRY=3,         (RETRY COUNT FOR INTERRUPTED SESSION)
       RETINT=30,       (TIME WAITED BY TCP BEFORE A RETRY)
       APPC=YES,        (YES IF NDM IS TO HAVE LU6.2 SESSION)
       MAXTASK=(4,1),   (CONCURRENT SESSION TOTAL,SWITCHED SESS)
       SWDLY=5,         (SECS. WAITED BEFORE VTAM SESSION RETRY)
       SWRTRY=3,        (NUMBER OF VTAM SESSION RETRIES)
       DDPREQ=YES,      (FORCED TO YES IF APPC=YES)
       RESYNCH=4,       (4098K BYTE BLOCKS BETWEEN CHECKPOINTS)
       MSGINFO=2,       (ALL MSGS GO TO SYSPRINT/IOF/CONSOLE)
       HOPCNT=5,        (NO. OF NODES AN LU6.2 MSG. CAN HOP)
       AUTEXIT=NDMEXIT, (USER-EXIT TO POINT TO RACF FOR IOF)
       NDCCAPI=NO,      (DO NOT HAVE API FEATURE INSTALLED)
       QMSURPT=NO,      (DEFAULT.  YES IF U HAVE USER APPLS)
       SUFFIX=18
```

Figure 31. Sample NDMTCP macro for TME 10 Software Distribution, Version 3.1.5 for AIX

The significant parameters are:

Parameter	Description
RESWAIT	This defines the number of seconds NetView DM for MVS waits for a response from a TME 10 Software Distribution, Version 3.1.5 for AIX node.
APPC	This must be set to YES to indicate that APPC is in use.

Parameter	Description
HOPCNT	A distribution is passed from node to node across the SNA network. Each time it is passed it performs a *hop*.
	The hop count is the maximum number of hops that can be included in a single routing chain from NetView DM for MVS. It must reach its destination within this number of hops.
	This parameter is provided to prevent messages looping for ever in complex networks with conflicting routing instructions at different nodes. In simple networks this parameter should be given a small number.

4. An NDMCP macro to define connection profiles that group LU 6.2 logical units that have the same attributes and connection capabilities.

   If your TME 10 Software Distribution, Version 3.1.5 for AIX workstation is connected by a line that allows the workstation to establish a session (for example, a leased line), you should:

   a. Define a connection profile with the polling parameter set to NO.
   b. Assign the LU name of the TME 10 Software Distribution, Version 3.1.5 for AIX workstation to that connection profile.

   In this way, TME 10 Software Distribution, Version 3.1.5 for AIX will not poll for a reply for a host-initiated request. An example is shown in Figure 32.

```
NDMCP  CPNAME=CP02,
       POLLING=NO
```

Figure 32. Example of an NDMCP macro

5. A transmission profile that groups nodes to connect them to the central site using the same type of line. Grouping nodes in this way optimizes line usage. For example, if a multipoint line is used to connect eight nodes, these nodes could be associated with a transmission profile that specifies that no more than three concurrent transmissions can take place against them, thus ensuring better load balancing across different lines.

   Each transmission profile can have its own retry specifications.

   An example of the NDMTP customization macro that is used to define a transmission profile is shown in Figure 33.

```
NDMTP  TPNAME=TPNDM2,TPTYPE=L,MINGR=27,MAXN=50
```

Figure 33. Example of an NDMTP macro

   where:

   • TPNAME is the name of the transmission profile

- TPTYPE specifies the connection type of the nodes (in this example, a leased line)

- MINGR specifies the number of transmission tasks that the TCP can grant to the transmission profile

- MAXN specifies the maximum number of transmission tasks that can be active at the same time.

Part 3. TME 10 Software Distribution Scenarios

This part presents scenarios that guide you through typical tasks using the TME 10 Software Distribution functions. The scenarios show how to:

- Prepare software using generic preparation
- Do a push installation of a change file to clients
- Authorize a client to do pull installations
- Do a pull installation of a change file from a client

Chapter 17. Software Distribution Scenarios

The following scenarios show the distribution and installation of applications.

The Software Distribution Environment

Before running the scenarios, be sure that you have, for example, the environment shown in Figure 34.
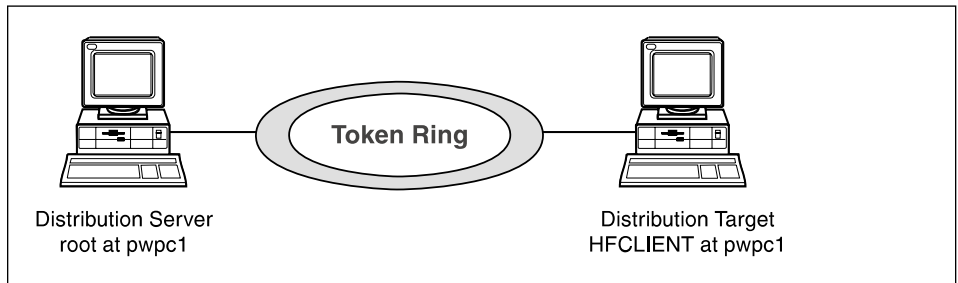
Figure 34. Software distribution environment

Be sure also that you have completed the following activities:

1 Installing the distribution server

The TME 10 Software Distribution server must be installed and configured on the LAN administrator workstation. The User ID used during configuration is, for these scenarios, **root** at **pwpc1**. See Chapter 9, "Installing TME 10 Software Distribution, Version 3.1.5 for AIX" on page 73 for a detailed explanation of installing and configuring a TME 10 Software Distribution server.

2 Installing the distribution client

The TME 10 Software Distribution client must be installed and configured on the target workstations. For these scenarios, the User ID used during configuration is **HFCLIENT** at **pwpc1**. HFCLIENT at pwpc1 uses root at pwpc1 as its distribution server. See Chapter 9, "Installing TME 10 Software Distribution, Version 3.1.5 for AIX" on page 73 for a detailed explanation of installing and configuring a TME 10 Software Distribution client.
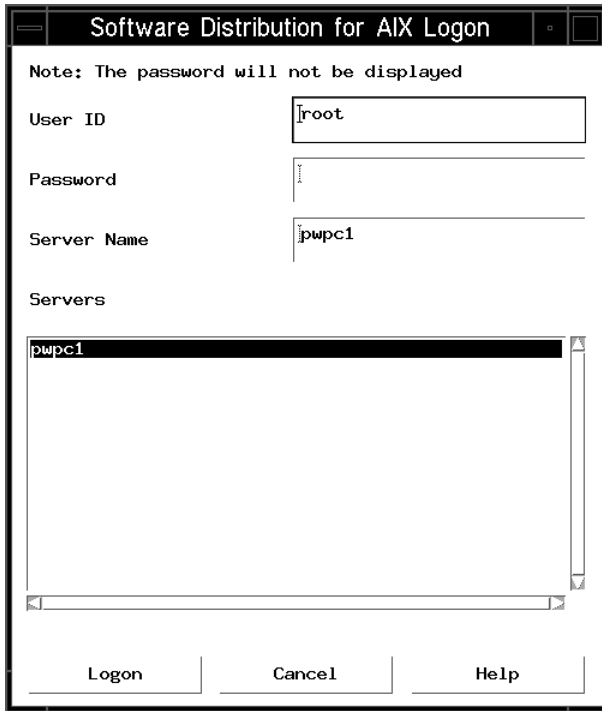
3 Starting the distribution server

The server is automatically started at system startup. If for any reason a distribution server (in this example, root at pwpc1) has been stopped, restart it by performing the following steps at the server workstation:

a At an AIX command line, from the `/usr/lpp/netviewdm/` directory, enter the **nvdmgi &** command.

The Logo window is displayed.

b Click on the **OK** push button. The Logon window is displayed, as shown in Figure 35 on page 184.

Figure 35. Logon window

c Enter your user ID and password and click on the **Logon** push button.

The Catalog window is displayed, as shown in Figure 36.

Figure 36. Catalog window

d Select **Engine** from the menu bar.

e Select **Start the system** from the pull-down menu.

The Catalog Informational message is displayed, as shown in Figure 37.

Figure 37. TME 10 Software Distribution Catalog informational message

f Click on the **OK** push button, the restarting of the distribution server is completed.

4 Defining the distribution clients

To define a group of clients as the target for distribution, refer to "Scenario 3: Defining and Authorizing a Client Workstation to Install Change Files" on page 201.

5 Starting the distribution clients

When a distribution server starts, all of its distribution clients that are started are automatically connected to the server. To view the list of the distribution clients, enter the following command from an AIX shell:

```
nvdm lstg *
```

If for any reason a distribution client (in this example, HFCLIENT at pwpc1) has been stopped, restart it by performing the following steps at the client workstation:

a Open the Catalog window, as shown on page 183.
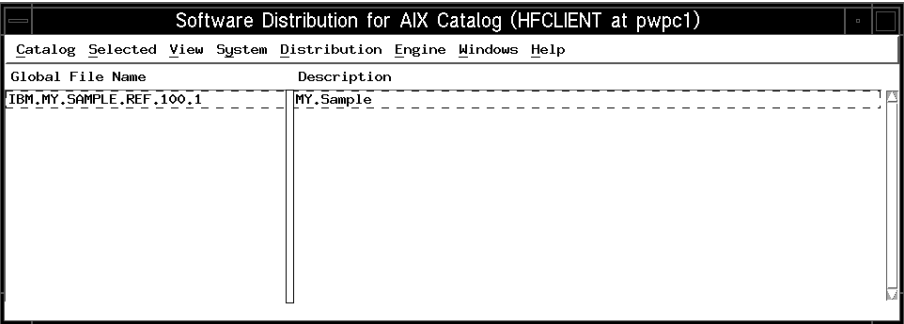
The Catalog window is displayed, as shown in Figure 38,

Figure 38. Catalog window

b Select **Engine** from the menu bar.

c Select **Start the system** from the pull-down menu.

The Catalog Informational message is displayed, as shown in Figure 39.

Figure 39. TME 10 Software Distribution Catalog informational message

d Click on the **OK** push button, the restarting of the distribution client is completed.

Scenario 1: Preparing an Application

During this scenario, you add a change file to the catalog at the distribution server.

Before beginning software preparation, perform these steps:

1 Make the sample directory in the /usr/lpp/netviewdm/ path of the preparation site workstation.

2 Create sample1 and sample2 files under the sample directory.

To add an application, such as the above sample product, to the catalog perform the following steps starting from the preparation workstation:

1 At an AIX command line, from the /usr/lpp/netviewdm/ directory, enter the **nvdmgi &** command.

The Logo window is displayed.

2 Click on the **OK** push button. The Logon window is displayed, as shown in Figure 40 on page 187.
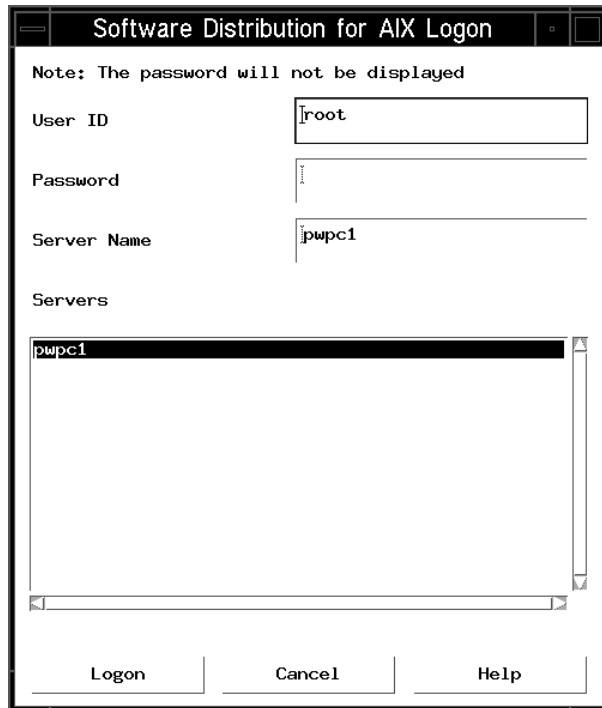
Figure 40. Logon window

3 Enter your user ID and password and click on the **Logon** push button.
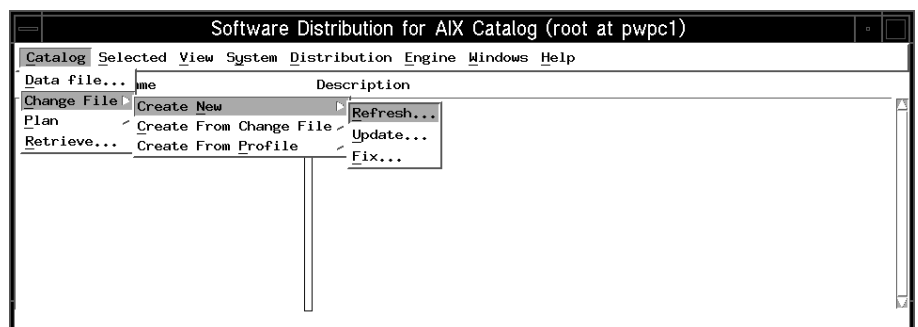
The Catalog window is displayed, as shown in Figure 41.

Figure 41. Catalog window

4 Select **Catalog** from the menu bar.

5 Select **Change File**.

6 Select **Refresh...** from the **Create New** choice in the pull-down menu.

The Change File Type window is displayed, as shown in Figure 42.

```
┌──────────────────────────────────────────┐
│ ─ │      Change File Type (root)          │
├──────────────────────────────────────────┤
│ ┌──────────────────────────────────┐ ┌─┐ │
│ │AIX installp                      │ │▲│ │
│ │Generic                           │ └─┘ │
│ │                                  │     │
│ │                                  │     │
│ │                                  │     │
│ │                                  │     │
│ │                                  │     │
│ │                                  │ ┌─┐ │
│ │                                  │ │▼│ │
│ └──────────────────────────────────┘ └─┘ │
│ ┌─┐                              ┌─┐      │
│ │◄│                              │►│      │
│ └─┘                              └─┘      │
│  ┌──────────┐  ┌──────────┐  ┌──────────┐ │
│  │    OK    │  │  Cancel  │  │   Help   │ │
│  └──────────┘  └──────────┘  └──────────┘ │
└──────────────────────────────────────────┘
```
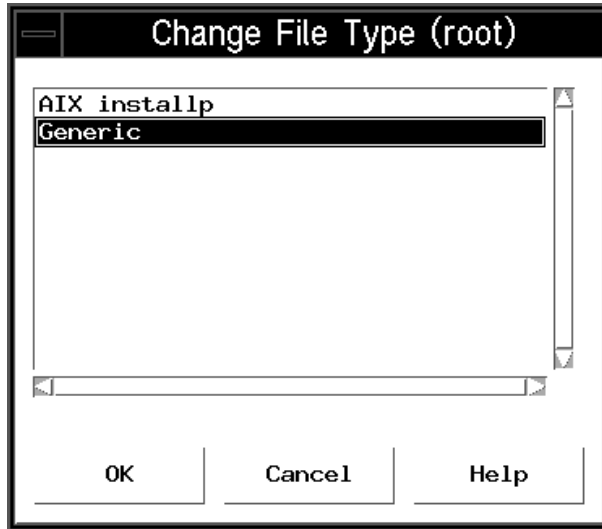
Figure 42. Change File type window

7 Select **Generic** from the list.

8 Click on the **OK** push button.

The Change File window is displayed, as shown in Figure 43 on page 189.

Figure 43. Change File window

  9 For **Component name** enter **IBM.MY.SAMPLE**.

10 For **Level** enter **100**.

11 For **Version** enter **1**.

12 For **Description** enter **MY.Sample**.

13 Click on the **Files** push button. The Files and Directories in Change File window is displayed, as shown in Figure 44 on page 190.
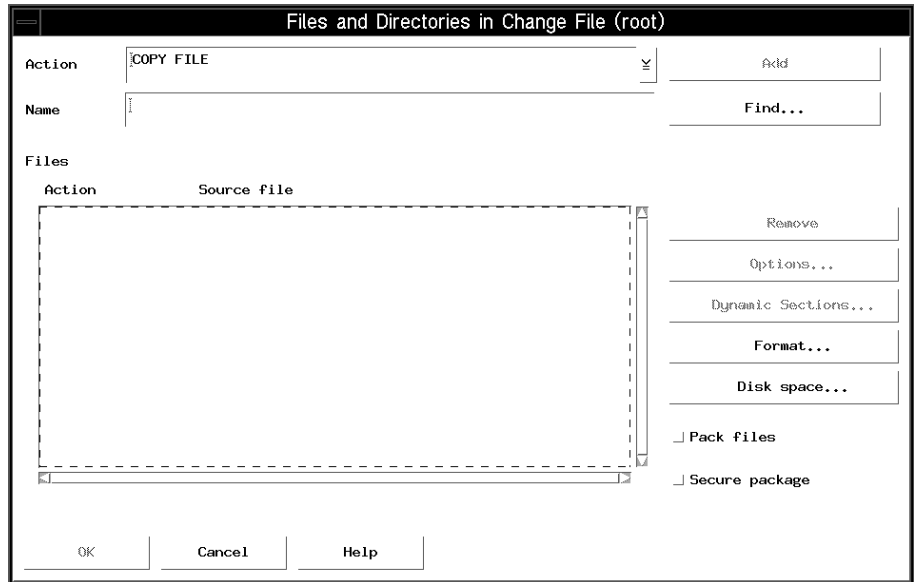
Preparing an Application

Figure 44. Files and directories in Change File window

14 To choose the files to be included in the change file, click on the **Find** push
button and the Find Files to Add in Change File window is displayed, as shown in
Figure 45 on page 191.

```
 ┌─────────────────────────────────────────────────────────────┐
 │ ▭    Find Files to Add to Change File (root)                 │
 ├─────────────────────────────────────────────────────────────┤
 │                                                             │
 │   Filter                                                    │
 │   ─────────────────────────────────────────────────────    │
 │   /usr/lpp/netviewdm/sample/*                               │
 │                                                             │
 │   Directories                               Files           │
 │  ┌──────────────────────────────────────┐▲ ┌───────────┐   │
 │  │/usr/lpp/netviewdm/sample/.            │█ │sample1    │   │
 │  │/usr/lpp/netviewdm/sample/..           │  │sample2    │   │
 │  │                                       │  │           │   │
 │  │                                       │  │           │   │
 │  │                                       │  │           │   │
 │  │                                       │  │           │   │
 │  │                                       │  │           │   │
 │  │                                       │  │           │   │
 │  │                                       │  │           │   │
 │  │                                       │▽ │           │   │
 │  │◁─────────────────────────────────▷   │  │◁──███─▷   │   │
 │  └──────────────────────────────────────┘  └───────────┘   │
 │                                                             │
 │   Selection                                                 │
 │   ─────────────────────────────────────────────────────    │
 │   /usr/lpp/netviewdm/sample/sample1                         │
 │                                                             │
 │  ┌──────┐      ┌────────┐      ┌───────┐      ┌──────┐      │
 │  │ Add  │      │ Filter │      │ Close │      │ Help │      │
 │  └──────┘      └────────┘      └───────┘      └──────┘      │
 │                                                             │
 └─────────────────────────────────────────────────────────────┘
```

Figure 45. Find files to add in Change File window

15 Select the files to be included in the change file, in this example **sample1** and
sample2, and click on the **Add** push button.

16 Click on the **Close** push button.

The selected files are included in the Files list on the Files and Directories in
Change File window, as shown in Figure 46 on page 192.

Preparing an Application
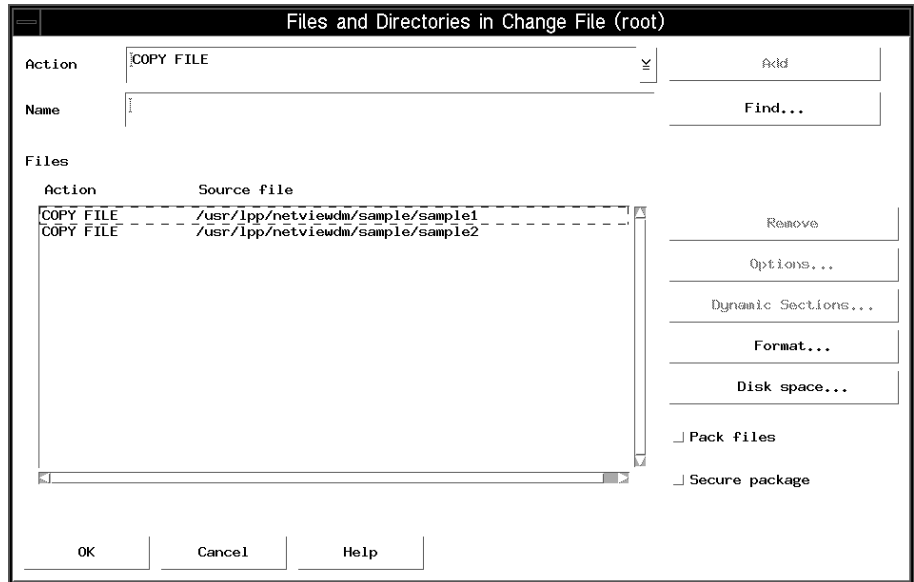
Figure 46. Files and directories in Change File window

17 Click on the **OK** push button.

The Change File window is displayed again, as shown in Figure 47 on page 193.

```
┌──────────────────────────────────────────────────────────────┐
│ ▬▬                   Change File (root)                       │
├──────────────────────────────────────────────────────────────┤
│                                                                │
│  Type              Generic, Refresh                            │
│                   ┌──────────────────────────────────────┐     │
│  Component name   │ IBM.MY.SAMPLE                        │ ⩔   │
│                   └──────────────────────────────────────┘     │
│                   ┌──────────────────────────────────────┐     │
│  Level            │ 100                                  │     │
│                   └──────────────────────────────────────┘     │
│                   ┌──────────────────────────────────────┐     │
│  Version          │ 1                                    │     │
│                   └──────────────────────────────────────┘     │
│                   ┌──────────────────────────────────────┐     │
│  Description      │ MY.Sample                            │     │
│                   └──────────────────────────────────────┘     │
│                   ┌──────────────────────────────────────┐     │
│  File name        │                                      │     │
│                   └──────────────────────────────────────┘     │
│                   ┌──────────────────────────────────────┐     │
│  Cost             │                                      │     │
│                   └──────────────────────────────────────┘     │
│                   ┌──────────────────────────────────────┐     │
│  Install. duration│ 00:00:00                             │     │
│                   └──────────────────────────────────────┘     │
│                   ┌──────────────────────────────────────┐     │
│  DAK              │ NONE                                 │ ⩔   │
│                   └──────────────────────────────────────┘     │
│                                                                │
│  Creation Date     07/11/96 15:20:59                           │
│                                                                │
│                                                                │
│   ┌───────────────┐   ┌───────────────┐   ┌───────────────┐    │
│   │   Files...    │   │ Parameters... │   │  Options...   │    │
│   └───────────────┘   └───────────────┘   └───────────────┘    │
│   ┌───────────────┐   ┌───────────────┐   ┌───────────────┐    │
│   │   Profile...  │   │ Compression...│   │ Products...   │    │
│   └───────────────┘   └───────────────┘   └───────────────┘    │
│   ┌───────────────┐   ┌───────────────────────────────────┐    │
│   │  Condition... │   │      Dynamic Sections...          │    │
│   └───────────────┘   └───────────────────────────────────┘    │
│   ☑ Build             ☑ Catalog            ☑ Import            │
│                                                                │
│   ┌───────────────┐   ┌───────────────┐   ┌───────────────┐    │
│   │     OK        │   │    Cancel     │   │    Help       │    │
│   └───────────────┘   └───────────────┘   └───────────────┘    │
└──────────────────────────────────────────────────────────────┘
```

Figure 47. Change File window

18 Leave the check boxes selected and click on the **OK** push button. The Software
Objects Catalog window is displayed containing the **IBM.MY.SAMPLE.REF.100.1**
change file just created, as shown in Figure 48 on page 194.

Preparing an Application
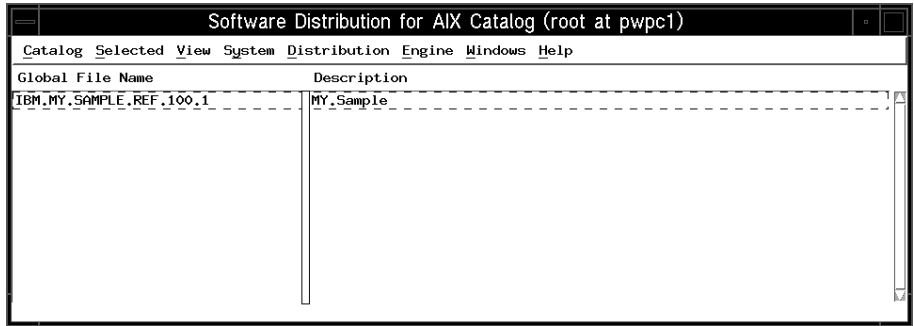
Figure 48. Change Files Catalog window

Now the **IBM.MY.SAMPLE.REF.100.1** change file is ready to be installed.

Scenario 2: Installing a Change File at a Target (Push Installation)

To install a change file on a client workstation, open the Catalog window, as shown in "Scenario 1: Preparing an Application" on page 186.

The Catalog window is displayed, as shown in Figure 49.

```
┌─────────────────────────────────────────────────────────────────────────┐
│ ═══      Software Distribution for AIX Catalog (root at pwpc1)     ▫  □  │
├─────────────────────────────────────────────────────────────────────────┤
│ Catalog  Selected  View  System  Distribution  Engine  Windows  Help     │
├─────────────────────────────────────────────────────────────────────────┤
│ Global File Name              Description                                 │
│ ┌──────────────────────────┐  ┌────────────────────────────────────────┐ │
│ │IBM.MY.SAMPLE.REF.100.1   │  │MY.Sample                               │ │
│ │                          │  │                                        │ │
│ │                          │  │                                        │ │
│ │                          │  │                                        │ │
│ │                          │  │                                        │ │
│ │                          │  │                                        │ │
│ │                          │  │                                        │ │
│ └──────────────────────────┘  └────────────────────────────────────────┘ │
└─────────────────────────────────────────────────────────────────────────┘
```
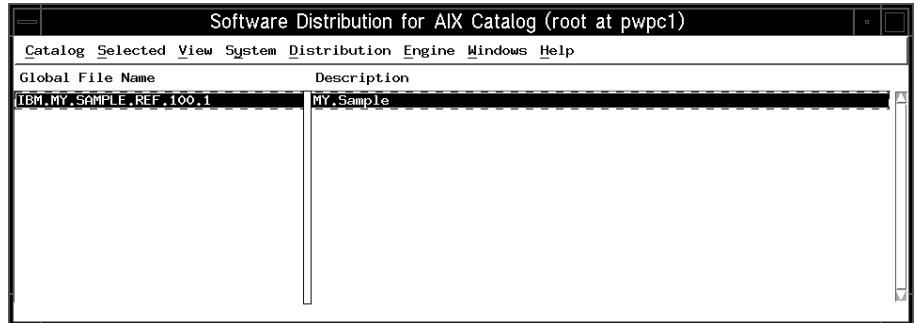
Figure 49. Catalog window

Perform the following steps:

1 Select **IBM.MY.SAMPLE.REF.100.1** change file.

2 Select **Install...** from the **Selected** menu choice. The Install Change Files window is displayed, as shown in Figure 50 on page 196.

Push Installation

```
┌─────────────────────────────────────────────────────────────────────────┐
│ ─  ─              Install Change Files (root)                             │
├───────────────────────────────────────────────────────────────────────────┤
│ Change Files                                                              │
│ ┌───────────────────────────────────────────────────────────────────┐ ▲ │
│ │IBM.MY.SAMPLE.REF.100.1                                            │ ░ │
│ │                                                                   │   │
│ │                                                                   │   │
│ │                                                                   │ ▼ │
│ └◄──┴──────────────────────────────────────────────────────────┴───┘   │
│                                                                           │
│ Targets                                                                   │
│ ┌──────────────────────────────────────────────────────────────────┐ ▲ │
│ │HFCLIENT           │Client (push)              My client          │ ░ │
│ │                   │                                              │   │  Select...
│ │                   │                                              │   │
│ │                   │                                              │   │
│ │                   │                                              │ ▼ │
│ └───────────────────┴──────────────────────────────────────────────┘   │
│                                                                           │
│ Schedule        Immediately                                              │
│ Execution                                  Scheduling...                 │
│   Not before :   When received by target                                │
│   Not after  :   Undefined                                              │
│                                                                           │
│ Procedure    [            ]          Find...    Target    [pwpc1    ]    │
│                                                                           │
│   Install        Options...        Frequency...      Close       Help    │
└─────────────────────────────────────────────────────────────────────────┘
```
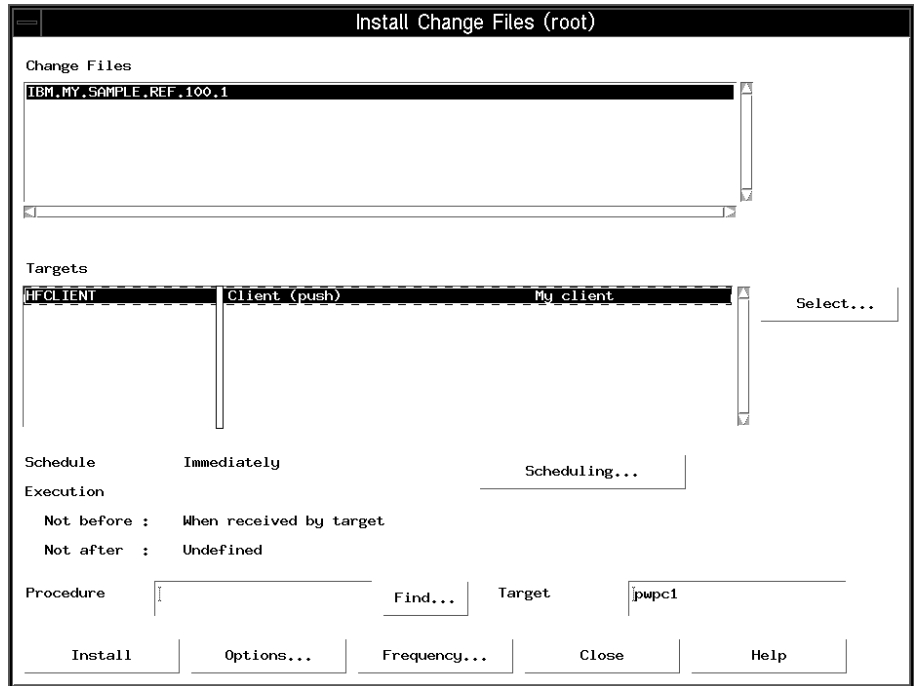
Figure 50. Install Change Files window

3 Select the target in the Targets list.

4 To schedule the installation, click on the **Scheduling...** push button. The Scheduling Information window is displayed, as shown in Figure 51 on page 197.
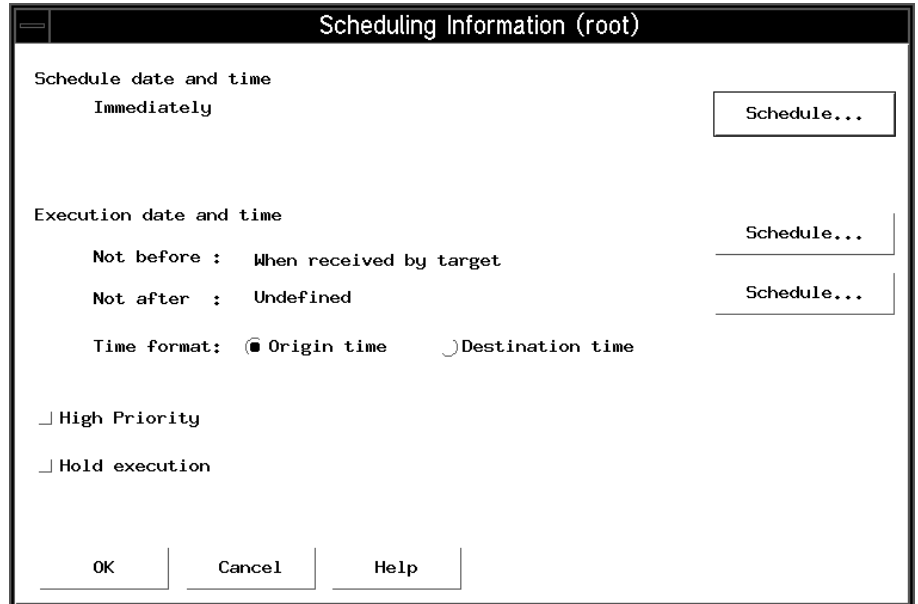
Figure 51. Scheduling information window

5 Click on the first **Schedule...** push button. The Schedule Time window is displayed, as shown in Figure 52.
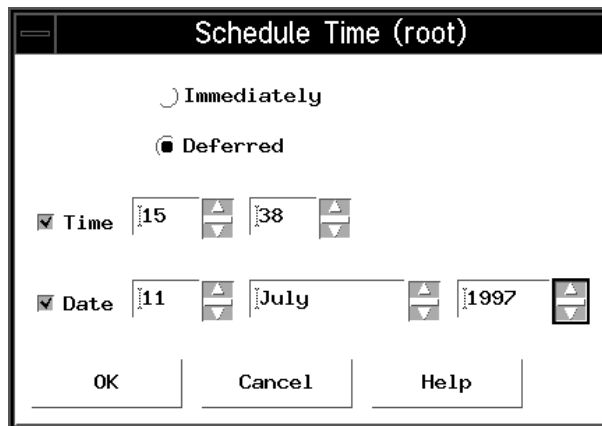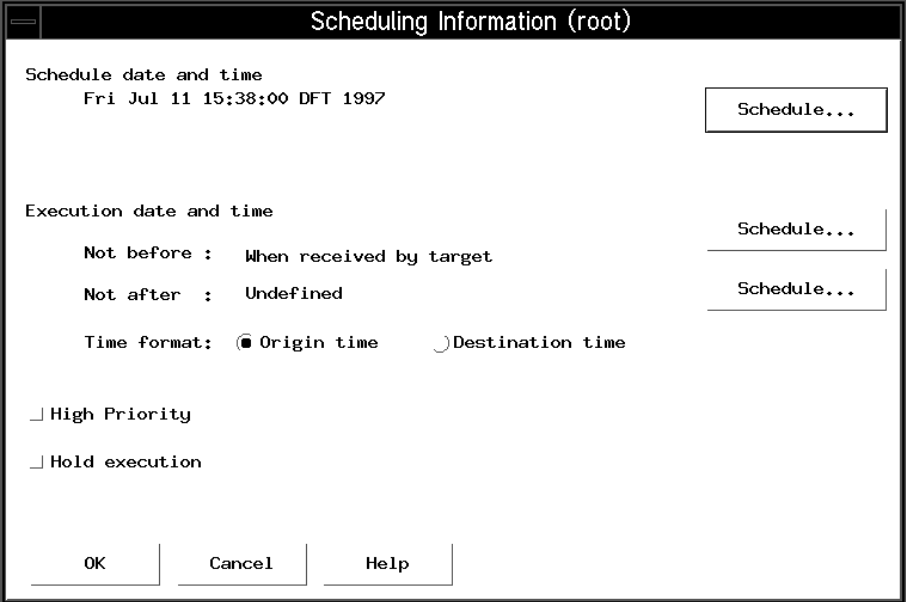
Figure 52. Schedule time window

6 Enter the new time and date and click on the **OK** push button. The Scheduling Information window is displayed again, as shown in Figure 53 on page 198.

Push Installation

Figure 53. Scheduling information window

7 Click on the **OK** push button. The Install Change Files window is displayed, as shown in Figure 54 on page 199.
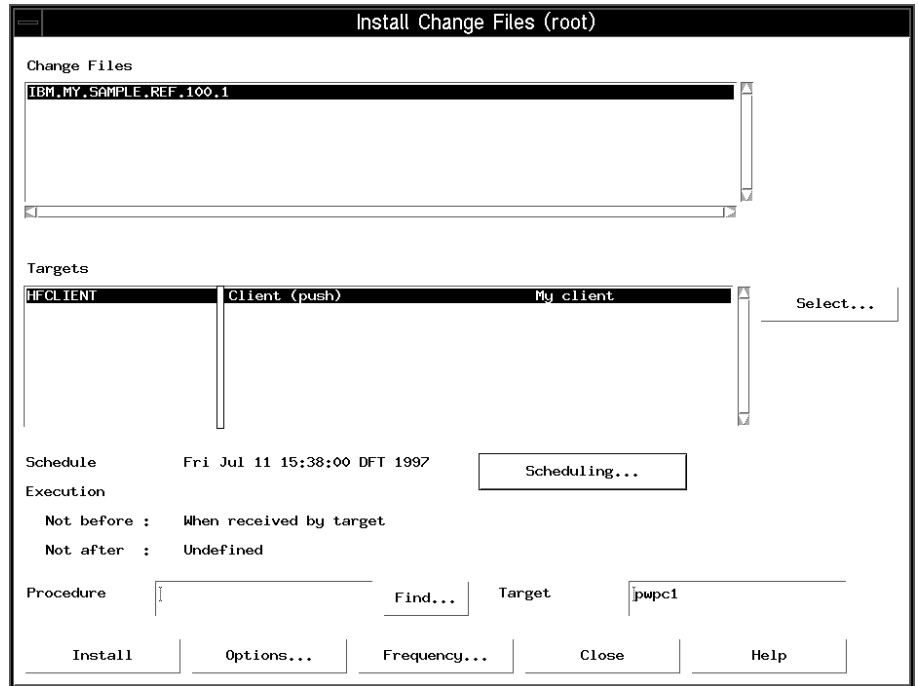
Figure 54. Install Change Files window

8 Click on the **Install** push button. The Correlators window is displayed, as shown in Figure 55.
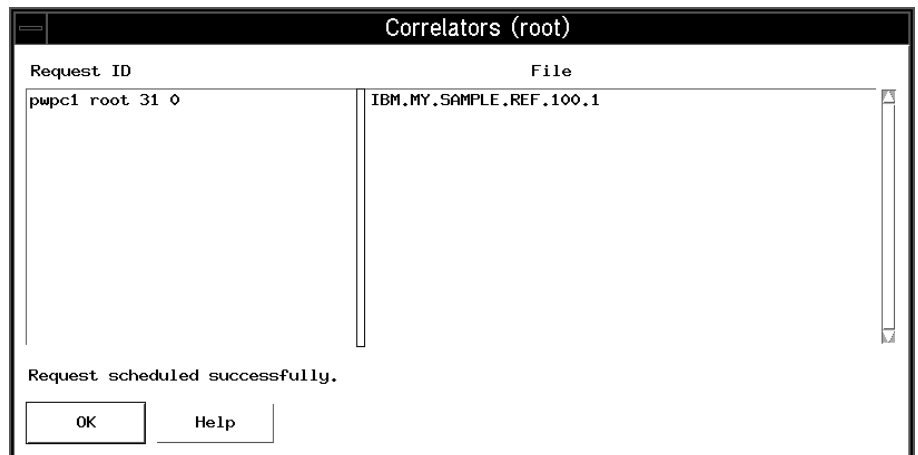
Figure 55. Correlators window

9 Click on the **OK** push button. The Install Change Files window is displayed again.

10 Click on the **Close** push button. The Catalog window is displayed.

The installation of the change file is completed.

Scenario 3: Defining and Authorizing a Client Workstation to Install Change Files

In this scenario you first define a client workstation and then you authorize it to install change files. Then from this workstation, you install a change file, using pull installation.

By default, a change file created at a server can be installed on request, without authorization, by other servers, but not by clients. A change file created at a client can be installed on request, without authorization, by that client and by servers, but not by other clients. You can change that default, and make all change files available to be installed on request from all clients.

On this scenario the default has not been changed. You define the **HFCLIENT** at **pwpc1** client workstation, as a target, from your **root** at **pwpc1** server workstation and authorize the client to install a software object from the catalog of **root** at **pwpc1**.

To define a target workstation, follow these steps:

1 Open the Catalog window, as shown in "Scenario 1: Preparing an Application" on page 186.

2 In the Catalog window, select **Targets** from the **Windows** menu bar. The Targets window is displayed, as shown in Figure 56.

```
┌──────────────────────────────────────────────────────────────────────┐
│ ─   Software Distribution for AIX Targets (root at pwpc1)        □ ▣   │
│ ┌──────────────────────────────────────────────────────────────────┐ │
│ │ Target  Selected  View  Windows  Help                            │ │
│ ├──────────────────────────────────────────────────────────────────┤ │
│ │ Name              Type              OS          Description       │ │
│ │                                                                    │ │
│ │                                                                    │ │
│ │                                                                    │ │
│ │                                                                    │ │
│ │                                                                    │ │
│ │                                                                    │ │
│ └──────────────────────────────────────────────────────────────────┘ │
└──────────────────────────────────────────────────────────────────────┘
```
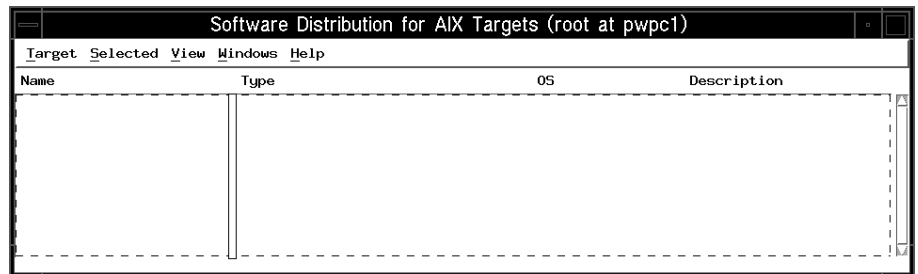
Figure 56. Targets window

3 Select **New target...** from the Target menu bar choice. The New Local Target window is displayed, as shown in Figure 57 on page 202.

Figure 57. New local target window

4 Fill in the fields as shown in Figure 57.

Note that the target address and LAN address fields must be filled in with specific data related to the client workstation that you are defining.

Name
Name of the target workstation.

Description
Brief description of the target workstation.

Target address
Address of the target workstation (it depends on the communications protocol that you choose by clicking on the **Protocol Type...** push button).

LAN address
LAN address of the target workstation.

Target type
Type of the target workstation.

Server name
Name of the distribution server for the target workstation.

Target OS
Operating system of the target workstation.

Domain Address
Address of the domain in which you define a remote target.

Access Key
Access key assigned to the target.

Password
Password to be used to access the target. You have to enter it every time you issue a command.

Verify Password
A check of the password just entered.

5 Click on the **OK** push button and the Targets window is displayed, as shown in Figure 58 on page 203.

```
┌─────────────────────────────────────────────────────────────────────────────┐
│─│          Software Distribution for AIX Targets (root at pwpc1)      │ □ │ ■ │
├─────────────────────────────────────────────────────────────────────────────┤
│ Target  Selected  View  Windows  Help                                         │
├─────────────────────────────────────────────────────────────────────────────┤
│ Name                    Type                      OS          Description      │
│ HFCLIENT                Client (push)             AIX         My client     ▲  │
│                                                                              │ │
│                                                                              │ │
│                                                                              │ │
│                                                                              │ │
│                                                                              ▼ │
└─────────────────────────────────────────────────────────────────────────────┘
```
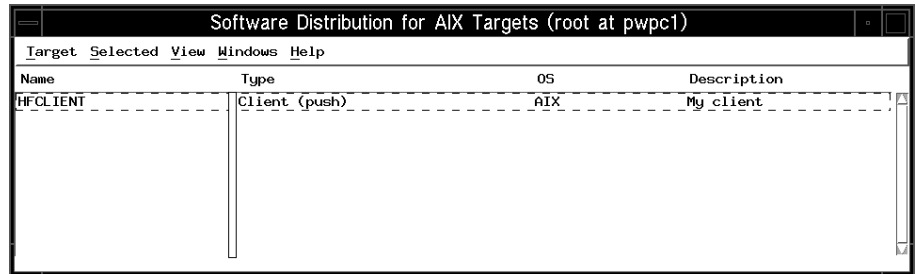
Figure 58. AIX targets window

The **HFCLIENT** at **pwpc1** client workstation is now defined.

Now you are ready to authorize this client to install software objects from the **root** at **pwpc1** server workstation.

6 Select **Catalog** from the **Windows** menu choice. The Catalog window is displayed, as shown in Figure 59.

```
┌─────────────────────────────────────────────────────────────────────────────┐
│─│          Software Distribution for AIX Catalog (root at pwpc1)      │ □ │ ■ │
├─────────────────────────────────────────────────────────────────────────────┤
│ Catalog  Selected  View  System  Distribution  Engine  Windows  Help          │
├─────────────────────────────────────────────────────────────────────────────┤
│ Global File Name                    Description                                │
│ IBM.MY.SAMPLE.REF.100.1             MY.Sample                              ▲  │
│                                                                              │ │
│                                                                              │ │
│                                                                              │ │
│                                                                              │ │
│                                                                              ▼ │
└─────────────────────────────────────────────────────────────────────────────┘
```
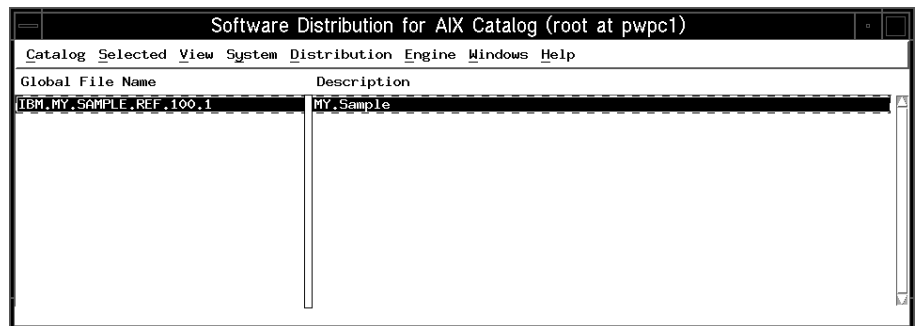
Figure 59. Catalog window

7 Select the **IBM.MY.SAMPLE.REF.100.1** change file.

8 Select **Authorize...** from the **Selected** pull-down. The Authorize Files window is displayed, as shown in Figure 60 on page 204.
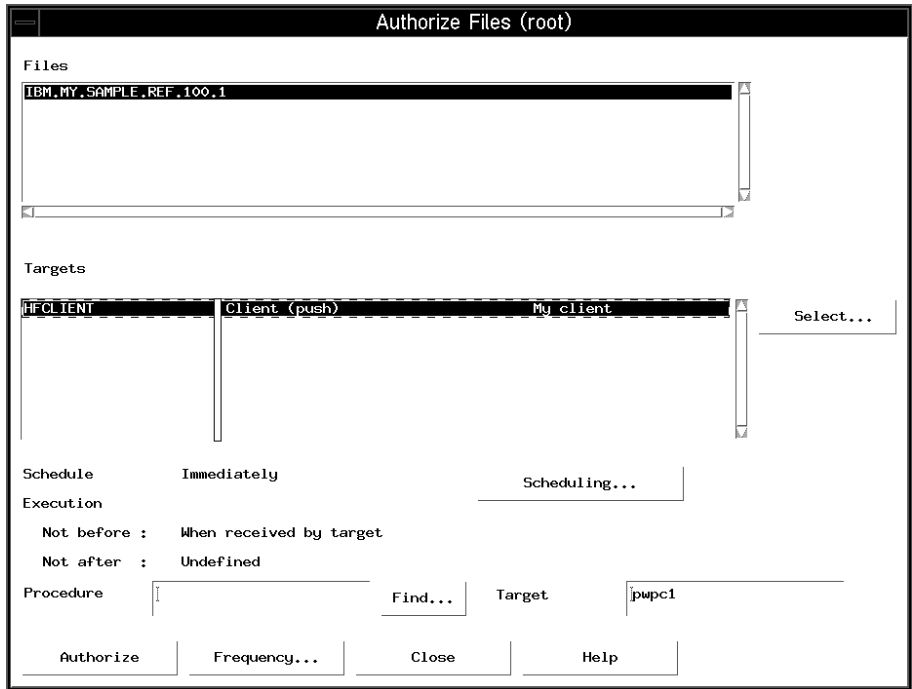
Figure 60. Authorize files window

9 Select **HFCLIENT** at **pwpc1** in the Targets list and click on the **Authorize** push
button. The Correlators window is displayed, as shown in Figure 61.
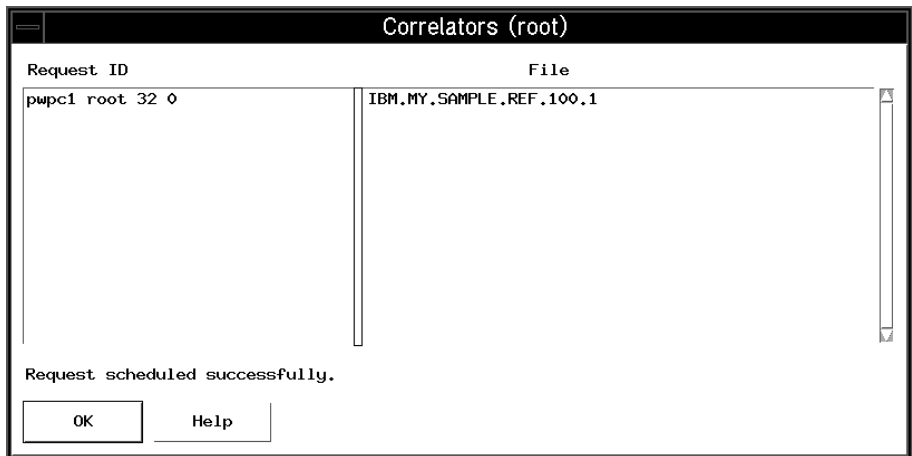
Figure 61. Correlators window

10 Click on the **OK** push button. The Authorize Files window is displayed again.

11   Click on the **Close** push button.

Now the **HFCLIENT** at **pwpc1** client workstation is authorized to install the
IBM.MY.SAMPLE.REF.100.1 change file.

Scenario 4: Installing an Application at a Client (Pull Installation)

After a client has been authorized to install an application, as shown in "Scenario 3: Defining and Authorizing a Client Workstation to Install Change Files" on page 201, a user at the client can request the installation. To request installation of an authorized application from a client, follow these steps at the client workstation:

1 Open the Catalog window, as shown in "Scenario 1: Preparing an Application" on page 186.

2 The Catalog window is displayed, as shown in Figure 62.
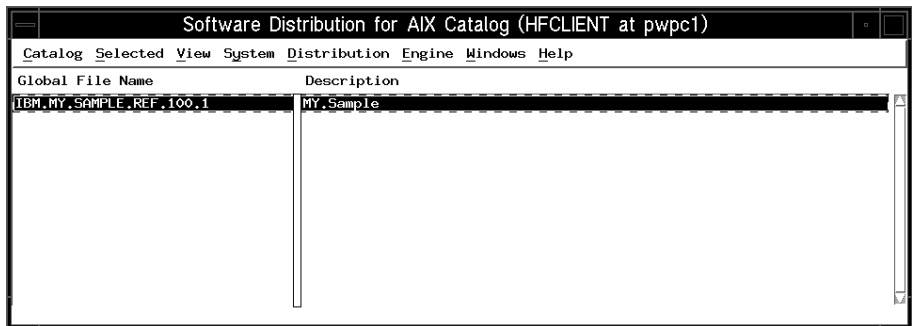
Figure 62. Catalog window

3 Select the **IBM.MY.SAMPLE.REF.100.1** change file.

4 Select **Install** from the **Selected** pull-down. The Install Change Files window is displayed, as shown in Figure 63.
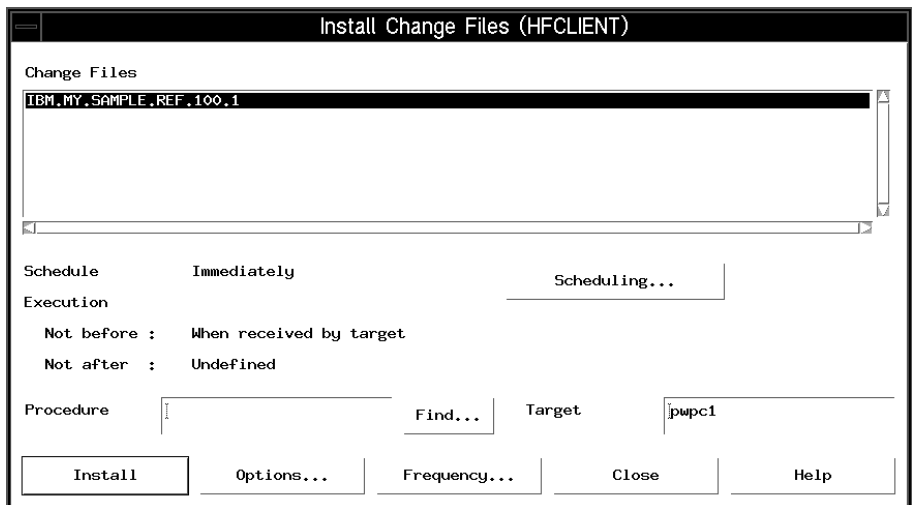
Figure 63. Install Change Files window

5 Click on the **Install** push button. The Correlators window is displayed, as shown in Figure 64 on page 207.

```
┌──────────────────────────────────────────────────────────────────────┐
│ ─ │              Correlators (HFCLIENT)                              │
├──────────────────────────────────────────────────────────────────────┤
│ Request ID                              File                          │
│ ┌─────────────────────────┐ ┌──────────────────────────────────┐ ▲   │
│ │ pwpc1 HFCLIENT 1 ◇      │ │ IBM.MY.SAMPLE.REF.100.1          │     │
│ │                         │ │                                  │     │
│ │                         │ │                                  │     │
│ │                         │ │                                  │     │
│ │                         │ │                                  │     │
│ │                         │ │                                  │     │
│ │                         │ │                                  │     │
│ │                         │ └──────────────────────────────────┘ ▼   │
│ └─────────────────────────┘                                          │
│ Request scheduled successfully.                                       │
│ ┌──────────┐   ┌──────────┐                                          │
│ │   OK     │   │   Help   │                                          │
│ └──────────┘   └──────────┘                                          │
└──────────────────────────────────────────────────────────────────────┘
```
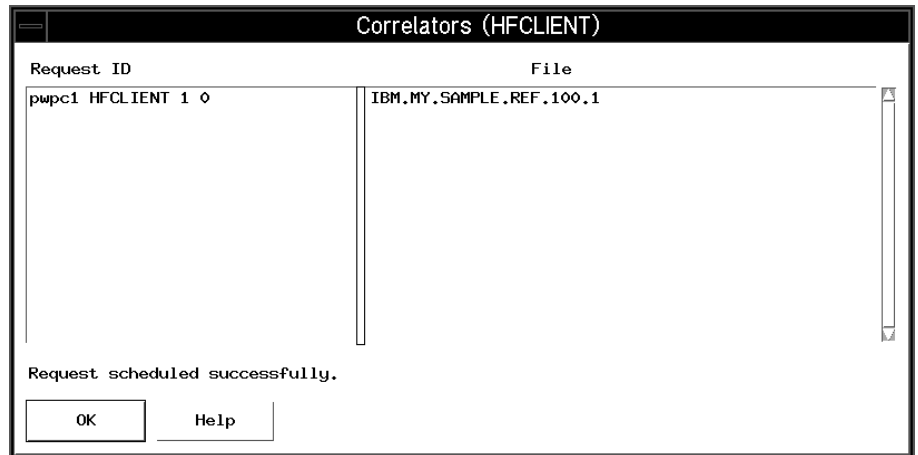
Figure 64. Correlators window

6 Click on the **OK** push button.

The pull installation of the **IBM.MY.SAMPLE.REF.100.1** change file on the **HFCLIENT** at **pwpc1** client workstation is completed.

Part 4. Appendixes

This part contains information to which you may need to refer from time to time while using TME 10 Software Distribution. The appendixes are:

Appendix A. Implementing Inventory Discovery

Inventory discovery is the process that discovers what hardware and software is present on a system. You can make use of inventory information when you formulate the prerequisites for change control and distribution requests. For instance, you can specify that a change file be installed only on workstations that have at least 5MB of disk space available. Inventory information can also be used as one of the criteria for forming dynamic groups.

Inventory data is transferred from clients to the server when you do either of the following:

- From the command line interface, submit the **nvdm inv** command.

- From the graphical interface, select **Inventory** from the **Selected** pull-down menu in the Targets window.

Inventory discovery is not a mandatory part of TME 10 Software Distribution. You can configure the hardware present on a local target using the user interfaces. However, the inventory discovery process is a way of compiling this information automatically and of keeping it up to date without the administrator of your system becoming involved.

Inventory discovery can be performed on remote targets if the targets involved in the operation are connected by the server-to-server (STS) transmission protocol.

Available Inventory Discovery Programs

Several inventory discovery programs are provided for your use. Some can be used on AIX targets, while another is available for OS/2 and Windows targets. For other operating systems, you can either purchase one from a third party or write one in house as described in "Writing an Inventory Discovery Program" on page 216.

AIX Inventory Discovery Programs

Software and hardware discovery processes for AIX are installed as part of TME 10 Software Distribution, Version 3.1.5 for AIX under /usr/lpp/netviewdm/bin (see "Software Inventory File" on page 217). Three programs are available:

fndinvhs   Discovers software and hardware
fndinvhw   Discovers hardware
fndinv   Discovers software

To run one of them on a target:

1 Specify the name of the program (with its complete path) in the INVENTORY PROGRAM keyword in the target's base configuration file (nvdm.cfg). The base configuration file is described in Chapter 10, "Editing the Base Configuration File" on page 79.

2 Enter the command:

Inventory Discovery

```
nvdm inv -w <target_name>
```

This command runs the program specified in `nvdm.cfg` at the target specified in `<target_name>`.

The discovered hardware and software data is collected and stored at the target in these files:

fndtkinv   Stores discovered installation parameters
fndswinv   Stores discovered software
fndhwinv   Stores discovered hardware

"Mapping to the TME 10 Software Distribution Inventory Files" on page 216 describes their formats. From these files the data is then transferred to the server's database.

Customizing the AIX Software Inventory Program

A file that you can use to customize the way the software inventory program `fndinv` discovers and catalogs software is provided under `/usr/lpp/netviewdm` with the name of `correlation.table`. By using that file, the inventory program stores information about discovered software assigning the same global name you defined in the `correlation.table`.

To discover and catalog software using the `correlation.table`, the software inventory program must be specified in the `INVENTORY PROGRAM` keyword with the following parameters:

```
INVENTORY PROGRAM:     fndinv -f <corr_path> [-p]
```

where:

<corr_path>

      Is the full path of the `correlation.table` file.

[-p]   Specifies that only the products described in the `correlation.table` must be cataloged, using the global names that are specified in that file. It must be specified between brackets.

      If the **[-p]** parameter is not specified, the discovery program catalogs all the software present on the target, assigning its own global names to the products that are not listed in the `correlation.table`.

The `correlation.table` provided with TME 10 Software Distribution for AIX contains entries only for the product options. Figure 65 on page 213 shows an extract of the `correlation.table` file.

```
IBM.NDM6000.BASE.REF.3130      /Software Distribution for AIX Base feature/
                                netviewdm6000.base.obj@3.1.5.0
                                :end.

IBM.NDM6000.SERVER.REF.3130   /Software Distribution for AIX Base feature/
                                netviewdm6000.server.obj@3.1.5.0
                                :end.

          .                                    .
          .                                    .
          .                                    .
          .                                    .

IBM.NDM6000.DE_DE.GITEXT.REF.3130   /Software Distribution for AIX GUI text files (de_DE)/
                                     netviewdm6000.gitext.de_DE@3.1.5.0
                                     :end.
```

Figure 65. Extract of the correlation.table

You can modify those entries and add new entries that the software inventory program uses when discovering and cataloging the software installed on the target. Use the following syntax to add a new entry:

```
<global_name>    /<description>/
                 <product_ID>
                 <product_ID>
                 <product_ID>
                 :end.
```

where:

<global_name>

    Is the global name assigned to the product. In Figure 65,
    `IBM.NDM6000.BASE.REF.3130` is a global name.

<description>

    Is the description of the product, always enclosed between / (slash). It must follow
    the global name. If you do not define any description, add only two slashes.

    In Figure 65, `/TME 10 Software Distribution for AIX Base Feature/` is a
    description.

<product_ID>

    Is the identifier of each option that composes the product, as defined in the ODM
    database. It is made up of two parts, separated by the symbol @ (at sign). The
    first part is the product tag, the second part is the product version and release. It
    must reflect the naming convention used in the ODM database. The format is
    case sensitive.

    In Figure 65, `netviewdm6000.base.obj@3.1.5.0` is a product ID.

:end.

    Is the tag that defines the end of an entry. It is mandatory.

When you add an entry to the `correlation.table`, you can use wildcard characters to indicate more than one global name, version, and product ID. In this way, you tell the

software inventory program to discover and catalog all software that meets the searching criteria.

Use the symbol % (percent) to indicate more than one global name and version, use an * (asterisk) to indicate more than one product ID.

For example, the following entry in the `correlation.table`:

```
XXX.SOM.REF.010000010001   /System Object Model /
                            som.somc@01.00.01.01
                            som.somd@01.00.01.01
                            som.somi@01.00.01.01
                            som.somk@01.00.01.01
                            :end.
```

can be written simply as:

```
XXX.SOM.REF.010000010001   /System Object Model/
                            som.*@01.00.01.01
                            :end.
```

or, if only one version of the product exists in the ODM database, as:

```
XXX.SOM.REF.010000010001   /System Object Model/
                            som.*@*
                            :end.
```

If more than one version of the product exists in the ODM database, you tell the software inventory program to discover all of those versions by writing in the `correlation.table`:

```
XXX.SOM.REF.%      /System Object Model Ver. %/
                    som.*@
                    :end.
```

The resulting information is:

```
XXX.SOM.REF.010000010001   /System Object Model Ver.010000010001/
                             som.somc@01.00.01.01
                             som.somd@01.00.01.01
                             som.somi@01.00.01.01
                             som.somk@01.00.01.01
                             :end.

XXX.SOM.REF.010000020001   /System Object Model Ver.010000020001/
                             som.somj@01.00.02.01
                             :end.
```

Attention! When you use the symbol % to indicate more than one version in the global name and description:

- The symbol @ must be specified in the product ID
- You can use an * before the symbol @ but you cannot specify any version after that symbol

OS/2 and Windows Inventory Discovery Program

A program that takes advantage of NetFinity inventory services on workstations that run OS/2 and Windows clients is provided as an optional component at client installation. It is called fndinv, and it uses NetFinity services to create the output files at each target (fndswinv, fndhwinv, and fndtkinv) which are then used by TME 10 Software Distribution.

Before running this inventory discovery program you must:

1 Make sure a NetFinity Manager workstation is installed in your network.

At the NetFinity Manager workstation:

2 Edit the inventory dictionary file named DEFAULT.SID. (You can also begin with a different dictionary file, but the final file that TME 10 Software Distribution is to use must be named DEFAULT.SID.) From the **Dictionary** pull-down, select **Edit** to edit the dictionary file. Select, in turn, each product you want to be detected by the TME 10 Software Distribution inventory discovery process. Edit the product to fill in two fields:

NVDM Change Object

Is the change file name that TME 10 Software Distribution is to use to identify this product.

NVDM Location Token

Is a symbol for which TME 10 Software Distribution will substitute the path in which the product is installed.

At the distribution server:

3 Retrieve `DEFAULT.SID` from the NetFinity Manager workstation. (If you already have a template `DEFAULT.SID` file on each target workstation, you can skip this step.)

4 Send `DEFAULT.SID` to the targets where you want to perform software and hardware inventory.

5 Identify the workstations where you want to run inventory discovery, and add the following keyword to their `nvdm.cfg` file:

```
INVENTORY PROGRAM:    fndinv
```

6 Enter:

```
nvdm inv -w target_name
```

The program gathers all hardware and software data from the target and stores it in the server's catalog.

Writing an Inventory Discovery Program

Inventory discovery processes are not supplied as part of every operating system. If a discovery process is not available, you can choose to buy one

If you are writing an inventory discovery process, you need only provide the information that TME 10 Software Distribution requires. This information is presented to TME 10 Software Distribution in text files, the format of which is described in "Mapping to the TME 10 Software Distribution Inventory Files." You do not need to provide information about all of the hardware or software present. Even a subset of the information can be used by TME 10 Software Distribution. In particular, you can choose to provide information about the hardware only or the software only.

To run the program when the inventory discovery is performed, the program name, with its complete path, must be specified in the `INVENTORY PROGRAM` keyword in the `nvdm.cfg` file. Inventory discovery is performed at each target running the program specified in the target's `nvdm.cfg` file.

Mapping to the TME 10 Software Distribution Inventory Files

Information about discovered software and hardware is presented to TME 10 Software Distribution at targets in text files. If you write inventory discovery yourself, you can send information directly to these files. If you choose to store the information differently, or if you are using a third-party inventory discovery procedure, you must produce a process to map your data to the inventory discovery files.

Installation Parameters Inventory File

You present information about discovered installation parameters in the installation parameters inventory file. This file is stored as `/usr/lpp/netviewdm/fndtkinv`. You need root privileges to write to it.

The first time you run inventory discovery at a target, the information stored in its `fndtkinv` file is registered at the server. Any additional updates to the `fndtkinv` file are registered at the server when the inventory discovery process is run again.

The file is a text file with a fixed format. Each line defines a single installation parameter. You can define up to 1000 installation parameters. Blank lines are permitted. Comment lines begin with #.

Each parameter is presented as a keyword followed by a colon (:) or an equal sign (=). For example, the installation parameters inventory file can contain the following lines:

```
dir1:   /u/mydir
dir2= /usr/lpp/mydir
```

Software Inventory File

You present information about discovered software in the software inventory file. This file is stored as `/usr/lpp/netviewdm/fndswinv.` and it is created when you run inventory discovery.

A default software inventory process, called `fndinv`, is installed as part of TME 10 Software Distribution under `/usr/lpp/netviewdm/bin`. This is the program that is executed when you run inventory discovery and creates the `fndswinv` file.

The file is a text file with a fixed format. Blank lines are permitted. Comment lines begin with #. You need root privileges to write to it.

Each product is described by keywords.

Figure 66 on page 218 shows a discovered software package. The package contains Version 1.3 of the messages required by a holiday booking application.

Mapping to the TME 10 Software Distribution Inventory Files

```
PRODUCT:            EUROTOURS.HOLS.MESSAGES.REF.0103
CHANGE FILE TYPE:   GEN
DESCRIPTION:        Message file for Eurotours Holidays Application
PRODUCT:
    TAG:            HOLMSG
    REVISION:       1.3
    ARCHITECTURE:   AIX RISC/6000
    VENDOR TAG:     IBM
    TITLE:          Message in a bottle

    FILESET:
        TAG:        Base
        REVISION:   1.1.2
        TITLE:      Base feature

    FILESET:
        TAG:        Message
        REVISION:   1.2
        TITLE:      Message feature
```

Figure 66. Discovered software package

The software inventory file that was provided with NetView DM/6000 Version 1.2 is still supported under TME 10 Software Distribution, Version 3.1.5 for AIX 3.1. It supplies this information:

```
PRODUCT:            EUROTOURS.HOLS.MESSAGES.REF.0103
DESCRIPTION:        Message file for Eurotours Holidays Application
```

Hardware Inventory File

You present information about discovered hardware in the hardware inventory file. This file is stored as `/usr/lpp/netviewdm/fndhwinv`. You need root privileges to write to it.

The file is a text file with a fixed format. Each line defines a single hardware parameter. You can define up to 128 hardware parameters. Blank lines are permitted. Comment lines begin with #.

Each parameter is presented as a keyword followed by a colon (:) followed by a number. The definition of the keyword and the meaning of the numeric value depends on the construction of the change file that references the hardware prerequisites.

For example, the hardware configuration file might contain the following line to indicate that the target has a 600 MB hard disk.

```
FixedDisk.capacityMb:    600
```

TME 10 Software Distribution, Version 3.1.5 for AIX uses the following default keywords to identify discovered hardware devices. If you use another tool, be sure to map the hardware keywords generated to these standard keywords:

Inventory.dateAcquired	Equipment.machineModel	OperatingSystem	OperatingSystem.version
PWS.memorySizeMb	PWS.parallelPorts	PWS.serialPorts	PWS.planar/
PWS.planarFRUNumber/	PWS.busType/	PWS.biosType/	PWS.biosRevisionLevel/
PWS.biosLevelDate/	PWSType.processor/	PWSType.processorSpeed/	PWS.coProcessors
PWS.coProcessor/	PWS.language/	PWS.cacheSizeKb	Display.adapters
Display.adapterType/	Display.type/	Display.memory/	Display.colors/
Display.hozResolution/	Display.verResolution/	Display.hozSize/	Display.verSize/
Keyboards	Keyboard.adapter/	Keyboard.type/	Keyboard.countryCode/
Keyboard.subCountryCode/	Keyboard.codePage/	Printers	Printer/
Printer.driver/	Plotters	Plotter/	Mouses
Mouse.type/	Mouse.buttons/	LogicalDisk.driveLetter/	LogicalDisk.volumeLabel/
LogicalDisk.fileSystem/	LogicalDisk.availCapacityMb/	DisketteDrives	DisketteDrive/
DisketteDrive.type/	DisketteDrive.adapter/	FixedDisks	FixedDisk.capacityMb/
DASD.adapterType/	DASD.adapterAttributes/	DASD.adapterIOType/	CDROMs
CDROM/	CDROM.capacityMb/	CDROM.typeModel/	Tapes
Tape/	Tape.typeModel/	EthernetAdapters	EthernetAdapter/
EthernetAdapter.address/	TRAdapters	TRAdapter/	TRAdapter.address/
NetworkInterfaces	NetworkInterface/	IP_Interface/	IP_Network.ip_address/
IP_Network.ip_subnet_mask/	IP_Network.ip_network_name/	AdapterBoards/	MagnetoOpticalDrive/
MagnetoOpticalDrive.capacityMb/	ProtocolConverter/	Repeater/	Router/
SerialDevice/	SerialPort/	Tablet.adapter/	Tablet.type/
ParallelPort/	SCSIAdapter/	OpticalAdapter/	TTY/
TTY.speed/	PTY/	HFT/	DLC/
LVM/	LFT/	RCM/	Adapter/
Driver/	Drawer/	OtherDevice/	Tablets
TTYs	PTYs	HFTs	DCLs
LVMs	LFTs	RCMs	MultiSubchannels
ParallelPorts	SerialPorts	MultiprotocolPorts	ProtocolDrivers
SCSIAdapters	OpticalAdapters	Drawers	OtherAdapters
OtherDrivers	OtherDevices		

Defining a Filter for Hardware Keywords

You do not have to store all the hardware keywords discovered for targets in the server database. You can define a subset of those you are interested in, and when the inventory data is sent to the server they will be filtered out from among those discovered. You may also specify keywords that are not found in the default list in "Hardware Inventory File" on page 218.

To record only selected hardware keywords in the catalog, specify them in the text file called hwfilter, which is stored in the db/ directory on the server. Each line in the file can contain only one keyword which contains no blanks. The file can include comment lines that are preceded by the number sign (#).

Each server in a hierarchy can have a different filter file (see "Routing Inventory Information to Upstream Servers" on page 220). Downstream servers send the complete file of discovered hardware to servers above it in the hierarchy; any filtering is performed at each server the inventory file is routed to.

Running Inventory Discovery

You do not normally need to run inventory discovery more often than every time you start your target. It is probably best to include inventory discovery as part of your startup sequence.

If you add software packages without using TME 10 Software Distribution and without restarting, run the inventory discovery command so that the new software is discovered and inserted in the database.

Routing Inventory Information to Upstream Servers

Notify TME 10 Software Distribution of the discovered hardware and software by performing the inventory discovery program on your target. The inventory command sends the data from your target to the server, where it is added to the central TME 10 Software Distribution database. You must run this command to register your inventory even if your target is the server itself.

Run the inventory discovery program whenever you change the contents of the inventory files. You can run the program as part of your startup sequence or from a script or directly from the user interfaces. You need Modify Configuration authorization to issue the inventory command.

Hardware inventory information is copied to the target hardware file. Software inventory data results in the creation of a catalog entry and status record with the status of `discovered`, which means that the software is installed but that it was not installed by TME 10 Software Distribution. Discovered software is active and not removable.

The discovered hardware inventory is completely refreshed each time this command is run. The discovered software inventory is cumulative. That is, you can only add entries using this command. To delete a discovered software package from the inventory, you must remove it from the catalog.

The inventory command is scheduled automatically when you add a new target using either the graphical interface or the **nvdm addtg** command from the command line interface.

Routing Inventory Information to Upstream Servers

The inventory data discovered for a target can be stored and updated at all servers connected in an upstream hierarchy to the target's local server. Information will be routed and stored at all servers that have the following keyword set to YES in their base configuration file (`nvdm.cfg`):

```
AUTOMATIC TARGET INFO UPDATE    YES
```

The base configuration file is described in the Chapter 10, "Editing the Base Configuration File" on page 79.

Appendix B. Replacing the Quiesce Check

This appendix describes how you can replace the quiesce check on any target.

What the Quiesce Check Does

Some change control operations check that the computer is not in use before the operations actually begin. This check is called the *quiesce check*, and, by default, checks that no users are logged on to the system. The results of the check are then returned to TME 10 Software Distribution. However, there may be occasions where the default quiesce check is not appropriate. For example, if users usually do not log off at the end of a work day, then the quiesce check always fails. A check that no processing power was being used is a more appropriate check.

Also, checking that no users are logged on does not prove that no processes are running that might interfere with the change control operation. For example, you may be attempting to uninstall a word processor on a computer while a background task (started by a user who is now logged off) is performing a batch of mail merges.

TME 10 Software Distribution enables you to address these kinds of scenarios by allowing the quiesce check to be replaced by a more appropriate check. Different targets on a network can each have different quiesce checks. This appendix describes how to replace the default TME 10 Software Distribution quiesce check with a check more suitable to the needs of your environment.

Implementing Quiesce Checks

The quiesce check on a target is performed by a script. This script is named `quiesced` and is stored in the directory with the other TME 10 Software Distribution scripts:

`/usr/lpp/netviewdm/script`

To replace the quiesce check, all you need to do is replace the script. The new script should:

- Not include any parameters
- Return 0 if the target is quiesced
- Return 2 if the target is not quiesced.

Example of a Check Script Used for Quiesce Checks

Figure 67 on page 222 shows an example of a quiesce script that performs the same function as the default script supplied with TME 10 Software Distribution, returning 0 if no users are on the system and returning 2 otherwise.

Example of a Check Script Used for Quiesce Checks

```
#
#    Checks if the Target computer is quiesced or not depending
#    on the number of logged-in users.  The Target is deemed to
#    be quiesced if no users are logged-in.
#
#
# Returns:
#
#   0 - the Target is quiesced
#   2 - the Target is not quiesced
#

numusers= who | wc -l            # Calculate the number of logged-in users
                                 # by listing the users and counting the
                                 # number of lines.

if test $numusers = 0 then       # If no users are logged in
    exit 0                       #   Return 0, meaning 'quiesced'
else                             # Otherwise
    exit 2                       #   Return 2, meaning 'not quiesced'
fi
```

Figure 67. Example of a check script used for quiesce checks

Appendix C. Writing Change Control Scripts

This appendix provides guidelines to help you write a change control script. It describes the parameters for scripts and how you can use them.

You can call change control scripts before and after most change control operations. The names of the scripts are contained within the change file. If no name is specified, no script is executed.

In addition to being named in the change file, scripts must be included as files to be installed unless they already exist on the target. The specification must refer to the complete path name where the script will be installed at the target.

Each script has a different set of parameters. These parameters, detailed in the next section of this appendix, are passed to it by the change control driver when it is executed.

A script produces a return code of 0 if it is successful. Any other value signifies that an error has occurred. If an error occurs, the entire change control request that is scheduled fails.

When scripts are being executed, any output generated to standard output is automatically redirected to a file called request.out. This file is stored in the work area. It is deleted before each new request.

The following sections describe the parameters that the driver passes to change control scripts when they are executed.

Creating Pre-Install and Post-Install Scripts

Use the following parameters for the pre-install and post-install scripts:

Install to Active Area
> Whether the change file is being installed to the active or service area. Valid entries are:
>
> **YES** The installation is to the active area.
> **NO** The installation is to the service area.

Service Subdirectory
> Name of the service subdirectory. This is NUL if the installation is to the active area.

Removable
> Whether the installation is removable. Valid entries are:
>
> **YES** The installation is removable.
>
> **NO** The installation is not removable.
>
> **DESIRED** Try to make the installation removable. If it cannot be done, do not fail the installation.

Backup Subdirectory
> Name of the backup subdirectory. This entry is NUL if the installation is not removable.

Creating Pre-Remove and Post-Remove Scripts

Use the following parameters for the pre-remove and post-remove scripts:

Request type
> The type of request that gave rise to this event. Possible values are REMOVE, or INSTALL if the current removal is taking place because an install request failed to complete.

Action
> The action to be taken by the remove may take one of these three values:

> DELETE_SERVICE   The installation should be deleted from the service subdirectory

> RESTORE_SERVICE   The removal should be performed in the service area

> RESTORE_ACTIVE   The removal should be performed in the active area.

Backup Subdirectory
> The name of the backup subdirectory.

Service Subdirectory
> The name of the service subdirectory. This is NUL if the action was RESTORE_ACTIVE.

Creating Pre-Accept and Post-Accept Scripts

Use the following parameters for the pre-accept and post-accept scripts:

Request type
> This specifies the type of request that caused this event. Possible values are ACCEPT, or INSTALL if the current accept is taking place because of an install request with automatic acceptance specified.

Backup Subdirectory
> The name of the backup subdirectory.

Service Subdirectory
> The name of the service subdirectory. This is NUL if there is no service subdirectory.

Creating Pre-Uninstall and Post-Uninstall Scripts

Use the following parameters for the pre-uninstall and post-uninstall scripts:

Action

The action to be taken by the accept may take one of four values:

DELETE_SERVICE   The installation should be deleted from the service subdirectory

RESTORE_SERVICE   The uninstall operation should be performed in the service area

RESTORE_ACTIVE   The uninstall operation should be performed in the active area

DELSERV_RESTACT   The change file should be deleted from the service area and the change file should then be uninstalled from the active area.

Backup Subdirectory

The name of the backup subdirectory. This is NUL if the change file was not removable.

Service Subdirectory

The name of the service subdirectory. This is NUL if the action is RESTORE_ACTIVE.

Creating Pre-Activate Scripts

Use the following parameters for the pre-activate script:

Request type

This specifies the type of request that is being activated. Possible values are INSTALL, REMOVE, and UNINSTALL.

Service Subdirectory

The name of the service subdirectory.

Understanding Script Exit Codes

All pre- and post-requests, pre- and post-scripts, and procedures invoked by TME 10 Software Distribution return the following exit codes:

0   The script was run successfully.

1   This exit code is reserved for use by the system. It indicates that the shell returned an error.

2-50   The script returned an error.

51-100   The script was run successfully.

All other codes

This exit code is reserved for use by the system. It indicates that an error occurred.

Appendix D. Writing User Exits

General-Use Programming Interface

Before reading this chapter you should have an understanding of C language programming, and be able to compile and link applications under the operating system TME 10 Software Distribution is running under.

Overview

In certain areas of TME 10 Software Distribution, user exits are included. User exits are places in the code where you can add your own functions.

The C language source code for these empty functional areas is provided with both the TME 10 Software Distribution server and client. You can edit the code to make it perform the function that you require, and then compile it into a shared library from which it can be called by the program.

This appendix describes each user exit function and explains how to compile and link any changes that you make. It provides:

- A description of the files provided to enable you to write and compile user exit code.

- A description of the user exits, detailing when they are called.

- A description of the compilation methods to use to compile and link user exit code.

Files Provided

The files shown in Table 34 on page 228 are provided at both the server and the client to enable you to write, compile, and link user exit code.

Writing User Exits

Table 34. Files Provided for User Exits		
Directory	**Files**	**Description**
/usr/lpp/netviewdm/src	fndcx.c	Source files containing the empty user exit functions.
	fndcxcm.c	
	fndcxmo.c	
	fndssext.c	
	fndsx.c	
/usr/lpp/netviewdm/src	fndssext.h	Header files used when compiling the user exit functions.
	fndcx.h	
	fndshrr.h	
	fndhdr.h	
/usr/lpp/netviewdm/src	fndsslib.exp	Export files used when linking user exit code.
	fndcxlib.exp	
/usr/lpp/netviewdm/src	fndex.mak	Make file used when compiling and linking the user exit code.

User Exits

This section describes each of the user exit functions that TME 10 Software Distribution supports. The next section describes the mechanics of compiling and linking any changes that you make.

Header Files

Some of the user exits receive as parameters structures that are used in the TME 10 Software Distribution program. To enable you to use these structures, the necessary header files are supplied. These files contain comments describing the meanings of the fields in the structures.

Because these files are the header files used for the actual product, they contain the definitions of structures that you do not require. Ignore these extra structures.

The structures and fields of interest to you are mentioned in the description for each of the user exits. Do not change any fields unless you are advised that you can do so. IBM will not support problems due to use of unauthorized fields in user exits.

Generating Local File Names (ss_user_loc_name)

This user exit is used only on the server. It is called when TME 10 Software Distribution needs to store a file, but does not have a local name to use. This event occurs when a distribution is received or a change file is built and cataloged without being given a local name.

TME 10 Software Distribution generates a default local name by concatenating the global name with the token $(REPOSITORY). The default local name is then passed to

this user exit, that can modify the name. Control is then returned to TME 10 Software Distribution, which uses the updated name to store the local file.

For change files and plans transmitted over server-to-server connections, however, the user exits is always called before cataloging the file to allow the user to change the local file name used. For the other types of files, the user exit is called to allow the user to change the local file name corresponding to the file system ID specified.

The name is passed as a pointer to a character array. This file contains the local file name as a null terminated string in a 256-byte buffer. If you replace it with a longer string, make sure that you do not exceed the 256-byte limit.

This user exit is located in the source file `fndssext.c`, and is called `ss_user_loc_name`.

Notifying Requests (sx_server_request)

This user exit is used only on the server. It is called after a change control or distribution request has been submitted, and is passed a data structure containing the request data.

This user exit can be used, for example, to record that a request is scheduled for a target and then correlate the request with the next user exit (sx_server_report) when the current request is completed.

This user exit is supplied with a pointer to an `RR_INFO` (defined in `fndcx.h`) and to a `CX_USER_RESPONSE` structure (defined in `fndcx.h`).

The `RR_INFO` structure contains the request to be performed in the `type_data` field. The `CX_USER_RESPONSE` structure is currently not used. In the future it may be possible for the user exit to return a value to this structure to specify that the request should be postponed or canceled.

Notifying Completion (sx_server_report)

This user exit is used only on the server. It is called when TME 10 Software Distribution receives a report from a target, immediately before the report is deleted, and after the update of the status in the database. A data structure containing the report is passed to it.

This user exit is supplied with a pointer to an `RR_INFO` (defined in `fndcx.h`). The `RR_INFO` structure contains the request to be performed in the `type_data` field.

Modifying Local File Names (cx_daca_filename)

This user exit is present on all TME 10 Software Distribution workstations. It is called immediately before:

- Deleting a local file
- Storing a local file
- Sending a local file (after a send request)
- Sending a local file (in answer to a retrieve request)

The user exit is provided with the global and local names of the file. The local name has already undergone token substitution at the time that it is called. You can change the local file name, but not the global name.

One use of user exit might be to reroute local files that should be stored in one directory to another directory, or to perform a more sophisticated form of token substitution than that provided.

The parameters to the user exit are the global name held in a structure that is defined in `fndhdr.h`, and the local name stored as a null terminated string in a 256-byte character array. If you replace the local name with a longer string, make sure that you do not exceed the 256-byte limit.

The user exit is located in the source file `fndcx.c` and is called `cx_daca_filename`.

Previewing and Modifying Reports (cx_daca_report)

This user exit is present on all TME 10 Software Distribution workstations. It is called immediately before the target sends a report about change control or distribution activity back to the server. A data structure containing the report is passed to it.

You can use this user exit to keep local copies of reports that are sent to the server, so that the user of the target can see what change control and distribution events have taken place.

The user exit is supplied with a pointer to an `RR_INFO` structure that contains the report that will be sent to the server. This structure is defined in `fndcx.h`.

The `RR_INFO` structure contains the report that will be sent to the server in the `type_data` field. The `type` field contains the type of report held in the `type_data` field.

The user exit is located in the source file `fndcx.c`, and is called `cx_daca_report`.

Previewing Requests (cx_daca_request)

This user exit is present on all TME 10 Software Distribution workstations. It is called immediately before a target carries out a change control or distribution request, and passed a data structure containing a copy of the request.

You can use this user exit to provide a warning to all users of the system that a change control or distribution operation is about to take place.

The user exit is supplied with a pointer to an `RR_INFO` structure (defined in `fndcx.h`) and a `CX_USER_RESPONSE` structure (defined in `fndcx.h`).

The `RR_INFO` structure contains the request that is about to be performed in the `type_data` field.

The `CX_USER_RESPONSE` structure is currently not used. In the future it may be possible for the user exit to return a value in this structure to specify that the request should be postponed or canceled.

The user exit is located in the source file `fndcx.c` and is called `cx_daca_request`.

Exporting Directories (sx_export_dir)

This function is called by the server to export a list of directories to a given target. It is called before enqueuing a request to the target.

The function is defined as follows:

```
DC_USHORT  sx_export_dir (REMOTE_DIR  *remote_dir,
                          TARGET_INFO *target_info,
                          DC_LONG     *errcode,
                          DC_USHORT   prot);
```

The `REMOTE_DIR` structure contains the following fields:

- Exported directory
- Mounted file system
- Mount options
- Export options

The values returned from the function are:

0   Directory successfully exported for the target. An informational message is logged.

4   Directory already exported for the target. An informational message is logged.

8   An error was encountered exporting the directory for the target. An error message is logged with the operating system error-code return in the output parameter *errcode*.

This user exit is located in the source file `fndsx.c`.

Removing Exports (sx_unexport_dir)

This function is called by the server to remove a list of directories for a given list of targets. It is called when the request is completed.

The function is defined as follows:

```
DC_USHORT sx_unexport_dir  (REMOTE_DIR  *remote_dir,
                            TARGET_INFO *target_info,
                            DC_LONG     *errcode,
                            DC_USHORT   prot);
```

The values returned from the function are:

0   Directory successfully unexported for the target. An informational message is logged.

8 An error was encountered unexporting the directory for the target. An error message is logged with the operating system error-code return in the output parameter `errcode`.

This user exit is located in the source file `fndsx.c`.

Mounting File Systems (cx_mount_fs)

This user exit is called by the agent to mount a list of file systems. It is called before executing the request.

The function is defined as follows:

```
DC_USHORT cx_mount_fs    (REMOTE_DIR  *remote_dir,
                          DC_VOID     **user_data,
                          DC_USHORT   prot);
```

The values returned from the function are:

0 File system successfully mounted. An informational message is logged.

4 File system already mounted. An error message is logged.

8 An error was encountered when mounting the file system. An error message is logged with the operating system error code return in the output parameter `errcode`.

This user exit is located in the source file `fndcxmo.c`.

Unmounting File Systems (cx_unmount_fs)

This user exit is called by the agent to unmount a list of file systems. It is called when the request is completed.

The function is defined as follows:

```
DC_USHORT cx_unmount_fs   (REMOTE_DIR  *remote_dir,
                           DC_VOID     **user_data,
                           DC_USHORT   prot);
```

The values returned from the function are:

0 File system successfully unmounted. An informational message is logged.

8 An error was encountered while unmounting the file system. An error message is logged with the operating system error-code return in the output parameter `errcode`.

This user exit is located in the source file `fndcxmo.c`.

Adding Target Parameters (sx_usrexit_trgcfg)

This user exit is called when a new target is defined. It is used to add any missing (optional) parameters to the target definition. The required fields must already be specified.

The function is defined as follows:

```
DC_SHORT sx_usrexit_trgcfg (const DC_USHORT       action
                                  const USRX_TARGET_CONFIG  *in_trg_config,
                                  USRX_TARGET_CONFIG  *out_trg_config);
/************************************************************************

 Description:
   This routine is called when a target configuration record is
   added or updated.
   A user code can be specified to set the default value of a
   target configuration record.

   The fields that the product considers are:

   description        : descriptive information
   target_os          : target operating system
   password           : target password
   lan_addr           : burned-in LAN address
   cust               : customer name
   contact            : contact name
   contact_phone      : contact phone number
   manager            : owning manager's name
   manager_addr       : owning manager's address
   num_hw_parm        : number of hardware parameters
   hw_parm            : hardware parameter array
   num_inst_parm      : number of installation parameters
   inst_parm          : installation parameter array


 Parameters:
   action
     ADD_CFG or UPD_CFG depending on command being executed

   in_trg_config
     input structure of target configuration record

   out_trg_config
     output structure of target configuration record in which
     user-exit default values can be specified

 Returns:
   0 completed successfully

 ************************************************************************/
```

This user exit is located in the source file fndsx.c.

Adding User Parameters (sx_usrexit_usrcfg)

This user-exit is called when a new user is defined in order to add any missing (optional) fields. The mandatory fields have to be already specified.

The function is defined as follows:

```
DC_SHORT sx_usrexit_usrcfg (const DC_USHORT     action
                                  const USRX_USER_DEF  *in_usr_def,
                                  USRX_USER_DEF  *out_usr_def);
/*********************************************************************

 Description:
   This routine is called when a user configuration record is
   added or updated.
   A user code can be specified to set the default value of a
   user configuration record.

   The fields that the product considers are:

   description        : descriptive information

 Parameters:
   action
     ADD_CFG or UPD_CFG depending on command being executed

   in_usr_def
     input structure of user configuration record

   out_usr_def
     output structure of user configuration record in which
     user-exit default values can be specified

 Returns:
   0 completed successfully

 *********************************************************************/
```

This user exit is located in the source file fndsx.c.

Reporting Request Completion (sx_server_report)

This user exit is called when a report is returned to the server immediately after the change control status has been updated.

The function is defined as follows:

```
DC_VOID sx_server_report (REPORT_INFO *info);
/**********************************************************************

 Description:
  This routine receives the REPORT_INFO structure immediately after
  a report has been received by the server and saved in the DB.

 Parameters:
   entry
     pointer to a REPORT_INFO structure.

 Returns:
   none


 **********************************************************************/
```

This user exit is located in the source file fndsx.c.

Reporting Fileset Statuses and Products (cx_vercm)

This function is called by a TME 10 Software Distribution agent when it has to verify the status of filesets contained in a change file. It is run at a workstation where an agent product is running, and expresses the status for filesets as a TME 10 Software Distribution change management status.

For each fileset in the change file, the agent calls the cx_vercm user exit and creates a report in file fndswcms. If the status of a fileset does not coincide with the status of the change file, the status reported by cx_vercm is In error.

For example, if an agent calls the cx_vercm user exit to check the status of OPP options in an installp change file, and the status for one of the options is Applied, then the change management status reported in the fndswcms file is Installed, removable, active.

This user exit is located in the source file fndcxcm.c. It has the following format:

```
DC_SHORT cx_vercm  (DC_CHAR        *cf_type,
                    DC_EXT_PRODUCT *product,
                    DC_EXT_FILESET *fileset,
                    DC_CHAR         *mask);
```

Product Information

Information related to products is also provided as input to this user exit. The same information is reported in the fndswcms file. It is supplied using the following format:

```
typedef struct dc_ext_product (DC_CHAR          tag[MAX_TAG_LEN];
        DC_CHAR         revision[MAX_REVISION_LEN];
        DC_CHAR         architecture[MAX_ARCHITECTURE_LEN];
        DC_CHAR         vendor_tag]MAX_VENDOR_TAG_LEN[;
        DC_CHAR         title[MAX_PF_TITLE_LEN];
                                 ) DC_EXT_PRODUCT;
```

Status Exchange Area

The area in the user exit used to exchange information about the status of a file has the following format. The agent provides the change management status of the change file to the procedure using the status field. The cx_vercm function determines the status of the file at the workstation and updates the status field.

```
typedef struct dc_ext_fileset (DC_CHAR          tag[MAX_TAG_LEN];
          DC_CHAR         revision[MAX_REVISION_LEN];
          DC_CHAR         title[MAX_PF_TITLE_LEN];
          DC_CHAR         status;
          DC_USHORT       fseterror;
                                 ) DC_EXT_FILESET;
```

If the fseterror field contains the value FS_NOT_DISCOVERED, then cx_vercm reports the Not discovered status in the fndswcms file.

TME 10 Software Distribution Statuses

A status is represented by one byte, where:

- The four least significant bytes represent a change management status.

- The four most significant bytes represent a specific TME 10 Software Distribution status.

TME 10 Software Distribution statuses can be one of the following:

- Available
- Installed, removable, active
- Installed, not removable, active
- Installed, removable, inactive
- Installed, not removable, inactive
- Scheduled
- In error
- Removed, inactive
- Uninstalled, inactive

- Backlevel
- Discovered
- Distributed
- Distribution pending
- Authorized at target
- Authorized at server

VERCM for installp Change Files

A specific implementation of cx_vercm is provided for installp change files. In these change files, the files are the OPP options and the status associated with each file is the status of the OPP option in the ODM. The cx_vercm function queries the ODM for the status of the OPP option and converts the status to a TME 10 Software Distribution change management status, according to the table represented by the following data structure:

```
struct nvdm_aix_state_stru (
                        DC_SHORT aix_state;
                        DC_CHAR nvdm_state;
                    ) nvdm_aix_state[4]=(
                                        (ST_AVAILABLE,CM_NONE      ),
                                        (ST_COMMITTED,CM_I_NR_ACTIVE),
                                        (ST_APPLIED,  CM_I_R_ACTIVE ),
                                        (ST_BROKEN,   CM_ERROR_STATE)
                                         );
```

The fndswcms Report

The fndswcms file is cataloged at the server as flat data so that it can be retrieved.

Following is an example of a report file.

```
GLOBAL NAME:          TRY.REF.1.0
CHANGE FILE TYPE:     AIXINSTP

PRODUCT;
    TAG:              product1
FILESET
    TAG:              opp_option1
    REVISION:         0.0.0.1
    STATUS:           available
FILESET
    TAG:              opp_option2
    REVISION:         0.0.0.1
    STATUS:           installed, removable, active
FILESET
    TAG:              opp_option2
    REVISION:         0.0.0.1
    STATUS:           available

OLD CM STATUS:        available
NEW CM STATUS:        in error, available


GLOBAL NAME:          TRY1.REF.1.0
CHANGE FILE TYPE:     AIXINSTP

PRODUCT;
    TAG:              product1
FILESET
    TAG:              opp_option1
    REVISION:         0.0.0.1
    STATUS:           available
FILESET
    TAG:              opp_option2
    REVISION:         0.0.0.1
    STATUS:           available
FILESET
    TAG:              opp_option2
    REVISION:         0.0.0.1
    STATUS:           available

OLD CM STATUS:        available
NEW CM STATUS:        available

GLOBAL NAME:          TRY2.REF.1.0
CHANGE FILE TYPE:     GEN

OLD CM STATUS:        available
NEW CM STATUS:        available

(TRY2.REF.1.0 is a change file without products or files.)
```

The Mask Parameter

Use the mask parameter to avoid having some change management statuses written in fndswcms. When the agent passes this parameter to cx_vercm, all the bits in it are set to 1. If cx_vercm returns the parameter with any bits set to 0, the agent ignores the related status when writing to fndswcms.

Setting Secure Keys (cx_set_secure_key)

This user exit is included in the TME 10 Software Distribution base package. It can be called when change files are built and the **secure package** attribute is set (see "Defining Data Security" on page 53).

The function calculates and associates a *secure key* with the change file. The user exit is defined as follows.

```
extern DC_VOID cx_set_secure_key( DC_ULONG  *crc_vector,
                                  DC_USHORT  number_of_crcs,
                                  DC_VOID   *secure_key,
                                  DC_USHORT  secure_key_len);
/*********************************************************************

 Description:
  This function is called by Software Distribution user interfaces
  to calculate, using a user-defined algorithm, the secure key
  to associate with a change file.


 Parameters:
     crc_vector:      Pointer to the array containing the CRCs
                      associated with all the objects in the
                      change file (the CRC is 0 for directories and
                      for objects to be deleted).

     number_of_crcs: Number of elements in the vector.

     secure_key:      Buffer containing the calculated secure key
                      to be checked at installation time
                      (to be returned). Software Distribution
                      assumes that if all the bytes are equal to
                      \0, the buffer is empty and no secure
                      key will be associated with the
                      change file. This is the default.

     secure_key_len: Length of the secure key (equal to 32 bytes)

 Return Codes: None
 *********************************************************************/
```

Checking Secure Keys (cx_check_secure_key)

This user exit is included in the TME 10 Software Distribution client package. It checks the secure key associated with the change file using the `cx_set_secure_key` user exit to make sure that the file has not been manipulated. It can be used to:

- Install the change file only if it contains a secure key.

- Install the change file only if it contains a secure key, and information in the secure key semantics identifies the target where the change file was built (for example, if the first three characters of the key are 111).

The user exit is defined as follows.

```
 extern DC_VOID cx_check_secure_key( DC_VOID   *secure_key,
                                     DC_USHORT  secure_key_len,
                                     DC_USHORT *action);
 /*********************************************************************

  Description:
   This function is called by Software Distribution
   when performing an installation to check the secure key
   set in the change file by the cx_set_secure_key.


  Parameters:
      secure_key:     Buffer containing the secure key to be check.
      secure_key_len: Length of the secure key (32 bytes).
      action:         Possible values are:
                    - CM_INSTALLABLE
                            This is the default. If this value is
                            returned, the agent proceeds with
                            the installation.

                    - CM_NO_INSTALLABLE
                            If this value is returned, the agent
                            halts the installation and reports
                            a failure.

  Returns:     None
 *********************************************************************/
```

Checking CRC Numbers (cx_check_crcs)

This user exit is included in the TME 10 Software Distribution client package. It recalculates the secure key associated with the change file using the `cx_set_secure_key` user exit to make sure that it is the same. If it is not, the change file is not installed.

The user exit is defined as follows.

```
extern DC_VOID cx_check_crcs( DC_ULONG  *crc_vector,
                              DC_USHORT  number_of_crcs,
                              DC_VOID   *secure_key,
                              DC_USHORT  secure_key_len,
                              DC_USHORT *action);
/*********************************************************************

 Description:
  This function is called by Software Distribution Client
  when performing an installation to recalculate the secure key
  set in the change file and compare it with the key set
  by the cx_set_secure_key when the change file was prepared.


 Parameters:
     crc_vector:      Pointer to the array containing the CRCs
                      associated with all the objects in the
                      change file (the CRC is 0 for directories
                      and for objects to be deleted).

     number_of_crcs: Number of elements in the vector.

     secure_key:      Buffer containing the calculated secure key
                      to be verified during the installation.
     secure_key_len: Length of the secure key (32 bytes).

     action:          Possible values are:

                  - CM_INSTALLABLE
                        This is the default. Proceed with the
                        installation.

                  - CM_NO_INSTALLABLE

                        If this value is returned, it means that
                        the comparison between the secure key
                        calculated when the change file was
                        prepared and the secure key calculated
                        during installation was not successful.
                        In this case Software Distribution Client
                        ends the installation and reports
                        a failure.

 Returns:     None
 *********************************************************************/
```

Compilation Method

A UNIX make file is provided to simplify compilation of the user exit code. This make file is called `fndex.mak`, and is located in `/usr/lpp/netviewdm/src` along with the user exit source code.

The user exits are compiled into two shared libraries:

- The `libfndss.a` library contains the user exit defined in `fndssext.c`.
- All others are contained in `libfndcd.a`.

The make file can be used to generate both of these libraries or either one of them. The syntax is:

`make -f fndex.mak sx`   To build `libfndsx.a`

`make -f fndex.mak ss`   To build `libfndss.a`

`make -f fndex.mak`   To build both of the above libraries

If you compile your exit routines on AIX 4.1 and you get the following message:

`ld: 0711-327 WARNING:Entry point not found: _nostart`

ignore it and continue your process.

If you need to include other source files in the compilation of user exit code, modify the make file accordingly.

————————— End of General-Use Programming Interface —————————

Appendix E. Using TME 10 Software Distribution from NetView/6000

If you are a NetView/6000 Version 2 user, you can access TME 10 Software Distribution to distribute the data and change files that are stored in the TME 10 Software Distribution catalog to targets in your network. The TME 10 Software Distribution functions available allow you to:

- Define the targets in your network

- Perform send, retrieve, and delete operations on data files and on change files

- Perform retrieve operations on noncataloged files

- Perform install, remove, accept, uninstall, and execute change control functions for change files

- Authorize and unauthorize data and change files for use at targets in your network

- Display the change control history records of targets in your network

- Verify the change control status of change files at targets

- Obtain inventory information for targets in a network

This chapter describes how to set up TME 10 Software Distribution from NetView/6000, how to define targets to both systems, and how to use change control and distribution.

Installing and Setting Up TME 10 Software Distribution

Install TME 10 Software Distribution by following the installation steps described in the *TME 10 Software Distribution Installation and Customization Guide*. After installation is complete, perform the following tasks from TME 10 Software Distribution to ensure that the two products are fully integrated:

1. Run the following TME 10 Software Distribution script:

   `/usr/lpp/netviewdm/script/nv6k.sh`

   To disable access to TME 10 Software Distribution from NetView/6000, run the script:

   `/usr/lpp/netviewdm/script/un_nv6k.sh`

2. Make sure that all NetView/6000 daemons are activated.

3. Enter the command `nvdm start`.

Defining Targets

Targets that you want to work with must be defined to both NetView/6000 and
TME 10 Software Distribution. The following two sections describe how to define:

- TME 10 Software Distribution targets to NetView/6000
- NetView/6000 targets to TME 10 Software Distribution

Defining TME 10 Software Distribution Targets to NetView/6000

When script `nv6k.sk` is run, the TME 10 Software Distribution target and all its local
targets are defined in the NetView/6000 catalog. You do not need to provide further
definitions of these targets to either system. Any remote targets connected to the
TME 10 Software Distribution target, however, are not automatically defined. To
perform operations with these targets, you must define them as TME 10 Software
Distribution targets to NetView/6000. Perform the following steps:

1. Display the IP Internet RootMap.
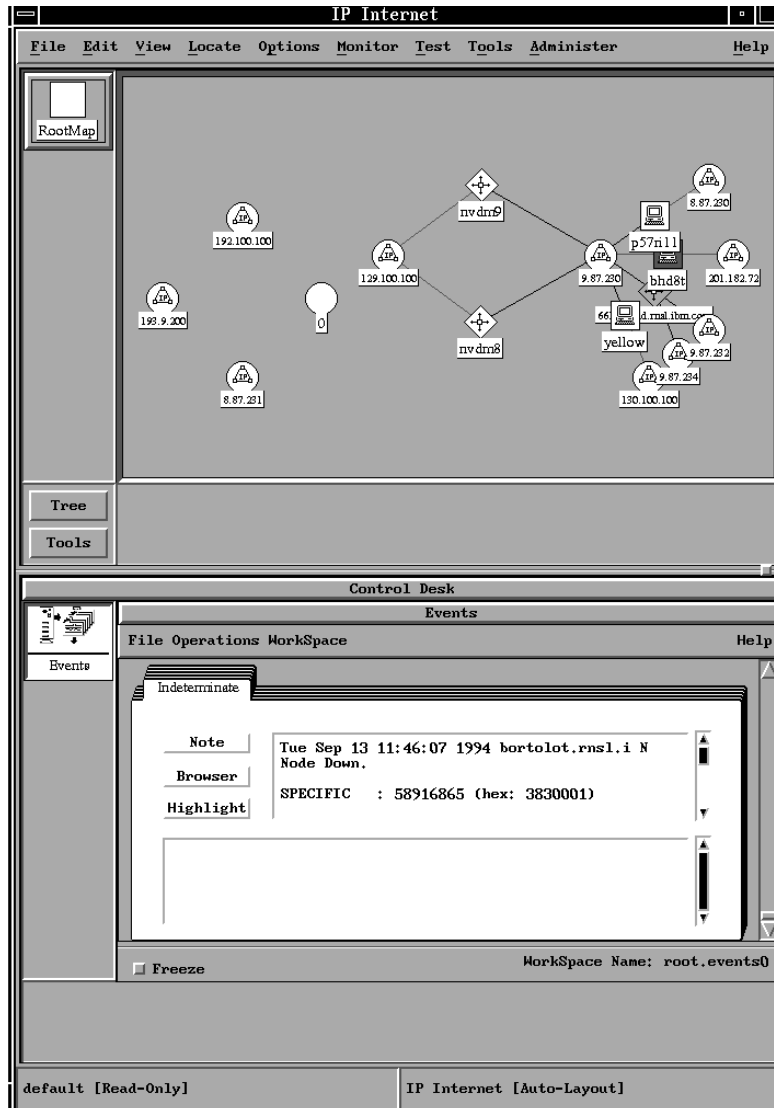
Figure 68. The IP Internet RootMap

2. Select a target from the RootMap by clicking the left mouse button.

3. Click on the right mouse button to display a menu.

4. Select **Edit**.

5. Select **Modify/Describe**.

6. Select **Object**. The Object Description window is displayed.

7. Select **General Attributes**.

8. Select **View/Modify Object Attributes**.

9. On the window displayed:

   - Enter the **TME 10 Software Distribution Target Name**.
   - Select the **True** check box in the **isNetViewDMTarget** field.

10. Select the **OK** push button twice to return to the IP Internet RootMap.

Defining NetView/6000 Targets to TME 10 Software Distribution

Define targets that exist in NetView/6000 to TME 10 Software Distribution by performing the following steps:

1. Display the IP Internet RootMap.

2. Select a target from the RootMap.

3. From the menu bar, select **Tools** and from the pull-down menu select **Software Distribution for NetView/6000**.

4. Select **Define the node as Software Distribution target** from the following menu.

5. Select **Local** or **Remote**, according to the type of node you are defining.

   The TME 10 Software Distribution Define Node window is displayed.

Performing Change Control and Distribution Operations

To perform change control and distribution operations on a target from NetView/6000, perform the following steps:

1. Display the IP Internet RootMap.

2. Select a target from the RootMap.

3. From the menu bar, select **Tools** and from the pull-down menu select **NetView DM/6000 for NetView/6000**.

4. Select one of the following operations from the further menu:

   - Install objects
   - Remove objects
   - Accept objects
   - Uninstall objects
   - Execute procedures
   - Verify CM status
   - Send objects
   - Retrieve objects
   - Retrieve noncataloged objects
   - Authorize objects
   - Unauthorize objects
   - Delete objects
   - View target history
   - Target inventory
   - Send file
   - Retrieve file

- Delete file
- Execute file

For example, select **Install objects**.

The TME 10 Software Distribution Catalog window is displayed.

```
┌─┬──────────────────────────────────────────────────────────────────┬──┬─┐
│ │          Software Distribution for AIX Catalog (root at p57ri11)  │ □│□│
│ Global File Name        Description                                        │
│ IBM.NDM6000.BASE.FIX.1 │NetView DM/6000 U436928 fix for Base feature   │▲│
│ IBM.NDM6000.BASE.P57RI │NetViewDM/6000 Base feature                    │ │
│ IBM.NDM6000.BASE.P57RI │NetViewDM/6000 Base feature                    │ │
│ IBM.NDM6000.BASE.REF.1 │NetView DM/6000 Base feature                   │ │
│ IBM.NDM6000.BASE.REF.2 │NetViewDM/6000 Base feature                    │ │
│ IBM.NDM6000.BASE.REF.3 │Software Distribution for AIX Base feature     │ │
│ IBM.NDM6000.CLGI.REF.3 │Software Distribution Client for AIX Graphical Int│
│ IBM.NDM6000.CLIENT.REF │NetViewDM/6000 Client feature                  │ │
│ IBM.NDM6000.CLIENT.REF │Software Distribution Client for AIX Client featur│
│ IBM.NDM6000.COMMS.FIX. │NetView DM/6000 U436928 fix for Communications fea│
│ IBM.NDM6000.COMMS.REF. │NetView DM/6000 Communications feature         │ │
│ IBM.NDM6000.COMMS.REF. │Software Distribution for AIX Communications featu│
│ IBM.NDM6000.EN_US.GITE │Software Distribution Client for AIX GUI text feat│
│ IBM.NDM6000.EN_US.GITE │Software Distribution for AIX GUI text feature │ │
│ IBM.NDM6000.EN_US.MAN. │Software Distribution Client for AIX Man Pages fea│
│ IBM.NDM6000.EN_US.MAN. │Software Distribution for AIX Man Pages feature│ │
│ IBM.NDM6000.EN_US.MSG. │Software Distribution Client for AIX Message Catal│
│ IBM.NDM6000.EN_US.MSG. │Software Distribution for AIX Message Catalog feat│
│ IBM.NDM6000.GI.FIX.112 │NetView DM/6000 U436928 fix for Graphical Interfac│
│ IBM.NDM6000.GI.REF.112 │NetView DM/6000 Graphical Interface            │▼│
│ ┌───────┐ ┌─────────┐ ┌─────────┐ ┌─────────┐ ┌────────┐ ┌──────┐       │
│ │Install│ │Details..│ │Include..│ │ Refresh │ │ Cancel │ │ Help │       │
│ └───────┘ └─────────┘ └─────────┘ └─────────┘ └────────┘ └──────┘       │
└────────────────────────────────────────────────────────────────────────┘
```

Figure 69. The Catalog window from NetView/6000

All the change files included in the catalog are displayed. Select the **Include** push button to limit the files displayed in the window.

5. Select the file or files you want to install on the target.

6. Select the **Install** push button. (The push button on the far left always corresponds to the action you chose on the previous NetView/6000 menu.)

Viewing Change Control History for a Target

To display a record of the change control and distribution operations that have been performed on a target, perform the following steps:

1. Display the IP Internet RootMap.

2. Select a target from the RootMap.

3. From the menu bar, select **Tools** and from the pull-down menu select **NetView DM/6000 for NetView/6000**.

4. Select **View target history** from the following menu. The Target History window is displayed.

Viewing Change Control History for a Target
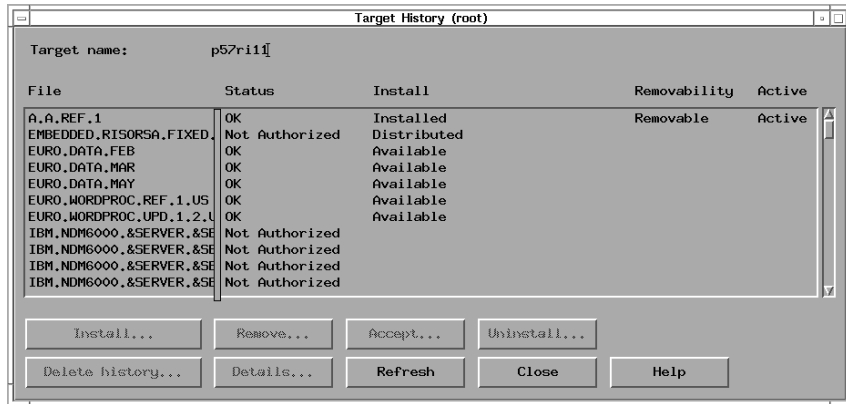
Figure 70. The target history window from NetView/6000

5. After viewing the information, select the **Close** push button to return to the IP Internet RootMap.

Appendix F. Customizing Message Alert Logging

You can specify whether TME 10 Software Distribution is to send alerts to NetView/6000 and to NetView, and the level of the alert each message is to have. Messages are assigned default alert levels, listed in Table 35 on page 251. You can change the levels using the procedure described in this chapter.

To customize the logging of alerts, use the alerts.cfg file and the two scripts, fndupdal and fndlsal, which are available at the CC server, in the /usr/lpp/netviewdm/script directory:

alerts.cfg   Is the standard output of the message catalog. It contains the TME 10 Software Distribution messages in the following format:

```
aaaa b c -d FNDxynnn <message text>
```

where:

aaaa   Is the number of the message in the catalog.

b   Is the flag that specifies whether an alert should be sent. Specify a to send alerts.

c   Is the alert number. It is currently set to 1-9. This parameter is ignored unless **b** is set to a.

-d   Identifies one of the log levels:

- -m (minimal)
- -n (normal)
- -d (diagnostic)

xy   Identifies one of the following TME 10 Software Distribution components:

- CL (Command Line Interface)
- CM (D&CC Agent)
- CO (Common Routines)
- DB (Database Access Services)
- DM (D&CC Agent Manager)
- EN (External Network Agent)
- FS (FS Server)
- GI (Graphical Interface)
- IN (Installation)
- NG (External Network Gateway)
- RB (Request Block Handler)
- RX (Request Block Transport)
- SH (Scheduler)
- TC (Transmission Controller)
- UC (User Interface Common Functions)

nnn   Identifies the message ID.

Customizing Message Alerts

fndupdal   Updates the `usr/lib/nls/msg/En_US/fndcomsg.cat` file with any changes you make to the alert settings (`alerts.cfg`). Its syntax is:

`fndupdal <input_file>`

The default input file is `alerts.cfg`.

To update only certain messages, you can create a file `<input_file>` containing those messages, which have the same format as the `alerts.cfg` messages.

fndlsal   Lists the message catalog standard output. To insert its output in a file, use the following syntax:

`fndlsal > <output_file>`

where

`<output_file>` is the name of any file.

Run this script every time you want to create a copy of the current message catalog file.

The following daemons must be active to ensure that alerts are correctly routed from TME 10 Software Distribution to NetView/6000 and to NetView:

trapgend *(for NetView/6000)*
     Must be installed and running on all the targets where the alerts are generated.

trapd *(for NetView/6000)*
     Must be running where NetView/6000 is installed.

snmpd *(for AIX)*
     Must be running on the targets where the alerts are generated.

spappld *(for NetView Service Point)*
     Must be running where NetView Service Point is installed

tralertd *(for NetView Service Point)*
     Must be running where NetView Service Point is installed

Not all TME 10 Software Distribution messages can be routed as alerts. Under the following circumstances, alerts cannot be produced because they are not logged in the `fndlog` file of the CC server:

- The product prints them only on the screen.
- Client/server communications are not working.
- The error occurs before TME 10 Software Distribution logging is active.

Message Categories and Alert Levels

Messages are classified in the following alert categories:

Category	Use
FND001	Internal faults and program logic errors

FND002 Configuration errors

FND003 Unexpected data from a TME 10 Software Distribution or related CC client task

FND004 Mismatched levels of TME 10 Software Distribution or related CC client software

FND005 Unexpected data from other software

FND006 System or hardware failures that may cause loss of data

FND007 Mismatch between the prerequisites for a change control operation and the current status of the target

FND008 Memory or disk allocation errors

FND009 Attempted security violations

The following matrix gives the default relationship between alert categories and messages.

Table 35 (Page 1 of 4). Categories of Alerts and Related Messages

Categories of Alert	Alertable Messages				
FND001	FNDCL019E	FNDCO127E	FNDFS003E	FNDTC032E	FNDTC138E
		FNDCO152E	FNDFS007E	FNDTC033E	FNDTC139E
	FNDCM023E		FNDFS008E	FNDTC034E	FNDTC153E
	FNDCM112E	FNDDB009E	FNDFS009E	FNDTC035E	FNDTC183E
		FNDDB010E		FNDTC036E	FNDTC184E
	FNDCO005E	FNDDB011E	FNDGI002E	FNDTC037E	FNDTC188E
	FNDCO014E	FNDDB012E	FNDGI017E	FNDTC038E	FNDTC189E
	FNDCO018E	FNDDB013E	FNDGI076E	FNDTC040E	FNDTC264E
	FNDCO026E	FNDDB014E	FNDGI077E	FNDTC041E	
	FNDCO030E	FNDDB028E	FNDGI078E	FNDTC042E	FNDUC007E
	FNDCO031E	FNDDB033E	FNDGI079E	FNDTC043E	FNDUC008E
	FNDCO055E	FNDDB034E		FNDTC044E	FNDUC010E
	FNDCO064E	FNDDB075E	FNDRB005E	FNDTC045E	
	FNDCO071E	FNDDB076E	FNDRB007E	FNDTC046E	
	FNDCO076E	FNDDB077E	FNDRB019E	FNDTC050E	
	FNDCO077E			FNDTC083E	
	FNDCO078E	FNDDM001E	FNDRX007E	FNDTC084E	
	FNDCO091E		FNDRX008E	FNDTC085E	
	FNDCO092E	FNDEN043E	FNDRX024I	FNDTC086E	
	FNDCO093E	FNDEN048E		FNDTC087E	
	FNDCO094E	FNDEN058E	FNDSH029E	FNDTC114E	
	FNDCO099E	FNDEN067E	FNDSH057E		
	FNDCO126E		FNDSH061E		

Table 35 (Page 2 of 4). Categories of Alerts and Related Messages

Categories of Alert	Alertable Messages				
FND002	FNDCL022E	FNDCO049E	FNDDB079E	FNDGI011E	FNDTC134E
	FNDCL141E	FNDCO050E	FNDDB080E	FNDGI020E	FNDTC135E
	FNDCL152E	FNDCO069E	FNDDB081E		FNDTC136E
		FNDCO070E	FNDDB082E		FNDTC137E
	FNDCM044E	FNDCO072E		FNDRB008E	FNDTC145E
	FNDCM065E	FNDCO074E	FNDEN032E	FNDRB011E	FNDTC151W
	FNDCM075E	FNDCO075E	FNDEN033W	FNDRB012E	FNDTC152E
	FNDCM085E	FNDCO079E	FNDEN049E	FNDRB018E	FNDTC155E
	FNDCM105E	FNDCO095E	FNDEN051E	FNDRB020E	FNDTC156E
	FNDCM106E	FNDCO096E	FNDEN068E	FNDRB023E	FNDTC162E
		FNDCO307E	FNDEN069E	FNDRB063E	FNDTC190E
	FNDCO025E		FNDEN070E		FNDTC209E
	FNDCO035E	FNDDB023E	FNDEN071E	FNDRX014E	FNDTC211E
	FNDCO036E	FNDDB024E	FNDEN072E	FNDRX016E	FNDTC245E
	FNDCO037E	FNDDB025E	FNDEN074W	FNDRX081E	FNDTC254E
	FNDCO038E	FNDDB026E			FNDTC256E
	FNDCO039E	FNDDB027E	FNDFS043E	FNDSH055E	FNDTC265E
	FNDCO040E	FNDDB032E	FNDFS044E	FNDSH080E	FNDTC266E
	FNDCO041E	FNDDB035E	FNDFS045E	FNDSH092E	FNDTC270E
	FNDCO043E	FNDDB036E	FNDFS046E	FNDSH093W	
	FNDCO044E	FNDDB037E	FNDFS047E	FNDSH098E	
	FNDCO045E	FNDDB038E	FNDFS048E	FNDSH101E	
	FNDCO046E	FNDDB039E	FNDFS068E	FNDSH200E	
	FNDCO047E	FNDDB053E		FNDSH201W	
	FNDCO048E	FNDDB058E		FNDSH220E	
FND003	FNDCM010E	FNDCM192E	FNDFS011E	FNDTC093E	FNDTC105E
	FNDCM014E	FNDCM193E		FNDTC094E	FNDTC118W
	FNDCM019E	FNDCM194E	FNDRB024W	FNDTC095E	
	FNDCM020E	FNDCM364E		FNDTC097E	
	FNDCM113E		FNDTC056E	FNDTC098E	
	FNDCM190E	FNDEN039E	FNDTC091E	FNDTC099E	
	FNDCM191E	FNDEN059E	FNDTC092E	FNDTC100E	
FND004	FNDCC002E	FNDCM016E	FNDGI010E	FNDSH003E	FNDTC096E
	FNDCC003E	FNDCM049E		FNDSH070W	FNDTC101E
			FNDRB004E		FNDTC131E
	FNDCM011E	FNDGI001E			FNDTC165E

Table 35 (Page 3 of 4). Categories of Alerts and Related Messages

Categories of Alert	Alertable Messages				
FND005	FNDCM155E	FNDFS020E	FNDRB100E	FNDTC068E	FNDTC172E
	FNDCM156E	FNDFS021E		FNDTC069E	FNDTC176E
	FNDCM157E	FNDFS024E	FNDSH044E	FNDTC070E	FNDTC195E
		FNDFS060E		FNDTC088E	FNDTC204E
	FNDCO007E	FNDFS075E	FNDTC001E	FNDTC089E	
	FNDCO010E		FNDTC002E	FNDTC103E	
	FNDCO063E	FNDNG005E	FNDTC003E	FNDTC104E	
	FNDCO101E	FNDNG007E	FNDTC004E	FNDTC106E	
	FNDCO115E	FNDNG008E	FNDTC005E	FNDTC109E	
	FNDCO116E	FNDNG009E	FNDTC006E	FNDTC110W	
	FNDCO117E	FNDNG010E	FNDTC007E	FNDTC111W	
	FNDCO121E	FNDNG011E	FNDTC008E	FNDTC112W	
	FNDCO122E	FNDNG012E	FNDTC009E	FNDTC113E	
	FNDCO123E	FNDNG013E	FNDTC010E	FNDTC117W	
	FNDCO128E	FNDNG014E	FNDTC011E	FNDTC120E	
		FNDNG015E	FNDTC012E	FNDTC124E	
	FNDDB078E	FNDNG016E	FNDTC013E	FNDTC125E	
		FNDNG017E	FNDTC014E	FNDTC126E	
	FNDEN027E	FNDNG018E	FNDTC015E	FNDTC127E	
	FNDEN040E	FNDNG019E	FNDTC017W	FNDTC132E	
	FNDEN044E	FNDNG020E	FNDTC018E	FNDTC133E	
	FNDEN045E	FNDNG021E	FNDTC030W	FNDTC159E	
	FNDEN046E	FNDNG023E	FNDTC031W	FNDTC161E	
	FNDEN054E	FNDNG024E	FNDTC053E	FNDTC164E	
		FNDNG025E	FNDTC060E	FNDTC165E	
	FNDFS012E	FNDNG026E	FNDTC062W	FNDTC166W	
	FNDFS017E	FNDNG027E	FNDTC063W	FNDTC167W	
	FNDFS018E	FNDNG028E	FNDTC066E	FNDTC170E	
	FNDFS019E	FNDNG029E	FNDTC067E	FNDTC171E	
FND006	FNDCC008E	FNDCM182E	FNDCO068E	FNDDB040E	FNDTC054E
	FNDCC009E	FNDCM183E	FNDCO080E		FNDTC055E
		FNDCM184E	FNDCO081E	FNDEN031W	FNDTC061W
	FNDCL041E	FNDCM185E	FNDCO082E	FNDEN035E	FNDTC178W
		FNDCM186E	FNDCO083E	FNDEN036E	FNDTC179W
	FNDCM006W	FNDCM187E	FNDCO084E	FNDEN055E	FNDTC210E
	FNDCM008E	FNDCM400W	FNDCO085E	FNDEN056E	FNDTC213E
	FNDCM009E	FNDCM401E	FNDCO086E	FNDEN061E	FNDTC219E
	FNDCM108E	FNDCM402E	FNDCO087E	FNDEN062E	FNDTC220E
	FNDCM114E	FNDCM403E	FNDCO088E		FNDTC222E
	FNDCM115E	FNDCM406E	FNDCO089E	FNDNG006E	FNDTC258E
	FNDCM154E		FNDCO107E		
	FNDCM170E	FNDCO002E	FNDCO108E	FNDRX009E	
	FNDCM171E	FNDCO003E	FNDCO124E	FNDRX011W	
	FNDCM174E	FNDCO004E	FNDCO125E	FNDRX013E	
	FNDCM175E	FNDCO006E	FNDCO129E	FNDRX015E	
	FNDCM176E	FNDCO013E	FNDCO130E	FNDRX075E	
	FNDCM177E	FNDCO029E	FNDCO131E	FNDRX077E	
	FNDCM178E	FNDCO053E	FNDCO314E	FNDRX084E	
	FNDCM179E	FNDCO054E	FNDCO315E		
	FNDCM180E	FNDCO056E	FNDCO317E	FNDTC051W	
	FNDCM181E	FNDCO057E	FNDCO318E	FNDTC052E	

Customizing Message Alerts

Table 35 (Page 4 of 4). Categories of Alerts and Related Messages

Categories of Alert	Alertable Messages				
FND007	FNDSH004E	FNDSH031E	FNDSH065E	FNDSH085E	FNDSH120E
	FNDSH005E	FNDSH032E	FNDSH066E	FNDSH086E	FNDSH130E
	FNDSH006E	FNDSH033E	FNDSH067E	FNDSH087E	FNDSH131E
	FNDSH007E	FNDSH034W	FNDSH068E	FNDSH096E	FNDSH132E
	FNDSH027W	FNDSH041E	FNDSH082E	FNDSH097E	FNDSH133E
	FNDSH028E	FNDSH042E	FNDSH083E	FNDSH099E	FNDSH134E
	FNDSH030E	FNDSH056E	FNDSH084E	FNDSH118E	FNDSH135E
FND008	FNDCL149E	FNDCO136E	FNDFS027E	FNDTC140E	
		FNDCO303E		FNDTC154E	
	FNDCM005E	FNDCO304E	FNDNG022E	FNDTC180E	
		FNDCO306E	FNDNG075E	FNDTC185E	
	FNDCO027E				
	FNDCO132E	FNDDB003E	FNDRB002E	FNDUC001E	
	FNDCO133E		FNDRB003E		
	FNDCO134E	FNDDM002E	FNDRX026E		
	FNDCO135E	FNDDM003E	FNDRX076E		
FND009	FNDCL017E	FNDRX017E	FNDTC243E		
			FNDTC244E		
	FNDRB009E	FNDSH039E	FNDTC261E		
	FNDRB010E	FNDSH137E			
	FNDRB017E				

Appendix G. Tivoli Enterprise Console and the Event Adapter

Tivoli Enterprise Console (T/EC) is a Tivoli Systems product that handles the alerts in the network. Event Server collects all the events, and send them to the administrator consoles. The administrator receives the events belonging to the classes for which he has been authorized. He can read the events, acknowledge them, take the appropriate actions, and close the events.

As you can specify whether TME 10 Software Distribution is to send alerts to NetView/6000 and to NetView, and the level of the alert each message is to have, so you can specify whether TME 10 Software Distribution is to send alerts to T/EC Event Server using TME 10 Software Distribution Event Adapter. Event Adapter establish an alert mechanism for the error messages issued by TME 10 Software Distribution

The Event Adapter interfaces the Event Server with the TME 10 Software Distribution server feature. The TME 10 Software Distribution Event Adapter is completely integrated with the TME 10 Software Distribution functions, and it is released with the server option. When you install the TME 10 Software Distribution server, you install the Event Adapter too.

Appendix H. Configuring the TME 10 Software Distribution Event Adapter

To configure the TME 10 Software Distribution Event Adapter so that TME 10 Software Distribution sends alerts to the Tivoli Event Server, perform the following tasks. You can find detailed information on how to configure the message alerts, in the Appendix F, "Customizing Message Alert Logging" on page 249 section.

1. To activate the alert mechanism insert in the /usr/lpp/netviewdm/db/nvdm.cfg server configuration file the following tag:

       TIVOLI ALERT: YES

2. To have the connection with Event Server configure the /usr/lpp/netviewdm/db/tecad_nvdm6k.conf and the /usr/lpp/netviewdm/db/alert_nvdm6k.conf files as follows:

   • Insert the hostname of the machine on which Event Server resides in the Server Location field of the /usr/lpp/netviewdm/db/tecad_nvdm6k.conf file.

   • In the /usr/lpp/netviewdm/db/alert_nvdm6k.conf file associate, to each alert category the severity. For detailed information on the message categories and the alert levels, see "Message Categories and Alert Levels" on page 250.

   The severities that Event Server provides by default are:

           FATAL
           CRITICAL
           MINOR
           WARNING
           HARMLESS
           UNKNOWN

   It is possible to associate, for example, the severity MINOR to the FND007 category. Assign different severities according to the importance you give to certain problems respect to others. For example, a bank could assign a severity of FATAL type for FND009 alert category. These category regards the security problems.

   T/EC gives the opportunity to insert new severity levels, modifying the root.baroc file under the following directory:

       $Inst_Directory/tec/rb_dir/TEC_CLASSES

   These new severities can also be inserted in the alert configuration file. If the event adapter fails in finding the alert configuration file, or in finding the tag related to the alert category that it should issue, it assigns the CRITICAL severity by default.

   The tecad_nvdm6k.conf and the alert_nvdm6k.conf files can be modified at run time, without stopping TME 10 Software Distribution. On the contrary, in order to make active a different value of the *TIVOLI ALERT* tag in the nvdm.cfg file, you

must stop and start the product, or issue the nvdm rld command. The default value of the *TIVOLI ALERT* tag is NO.

After you performed the above steps at the TME 10 Software Distribution server, you can send alerts related to all the machines in its domain automatically.

Appendix I. Customizing the Tivoli Event Server

To customize the Tivoli Event Server to receive, recognize, and classify an event coming from TME 10 Software Distribution follow the steps below.

You can change all the names in this sequence, except for the NVDM_EVENT class name and the NVDM6K source name, because these two labels are hard-coded in the Event Adapter. You must be familiar with T/EC to execute the following steps:

1. To create a new base rule, enter:

   ```
   wcrtrb -d <tec> NVDM_Rule
   ```

   where <tec> is the directory on which T/EC is installed. If you choose the default installation options the directory on which you install T/EC is:

   ```
   /var/spool/Tivoli/<server name>.db/tec
   ```

2. To copy the Default rule in the new base rule enter:

   ```
   wcprb -cr Default NVDM_Rule
   ```

3. To Compile the new rule base enter:

   ```
   wcomprules NVDM_Rule
   ```

4. Copy the tecad_nvdm6k.baroc file from the /usr/lpp/netviewdm/db directory of the TME 10 Software Distribution server to the <tec>/TEC_CLASSES. Do this step only if the file does not exist in the directory, that means that you performed the step for another server of the your domain.

5. To check for congruency of the Baroc file with the rule base file enter:

   ```
   wchkclass <tec>/TEC_CLASSES/tecad_nvdm6k.baroc NVDM_Rule
   ```

   The output of this command must be *Validation of class succeeded*.

6. To import the baroc file in the rule base file enter:

   ```
   wimprbclass <tec>/TEC_CLASSES/tecad_nvdm6k.baroc NVDM_Rule
   ```

7. To load the new rule base file enter:

   ```
   wloadrb NVDM_Rule
   ```

8. To stop and restart the Event Server enter:

   ```
   wstopesvr

   wstartesvr
   ```

9. To create a new event source enter:

   ```
   wcrtsrc -l "NVDM Source" NVDM6K
   ```

10. To create a new event group enter:

    ```
    wcrteg "NVDM Group"
    ```

    To define a new event filter enter:

    ```
    waddegflt -s NVDM6K -c NVDM_EVENT "NVDM Group"
    ```

Customizing the Tivoli Event Server

To check if the event filter has been successfully created enter:

```
wlseg -f "NVDM Group"
```

11. To assign the event group to an administrator enter:

```
wassigneg @<ADMIN_NAME> "NVDM Group" senior admin user
```

This step allows the administrator, to open his Tivoli Enterprise Console, to manage the alerts related to TME 10 Software Distribution.

Appendix J. The TME 10 Software Distribution Event Adapter Alert Fields

The TME 10 Software Distribution Event Adapter fills the following fields for each event:

```
SOURCE               = NVDM6K
ORIGIN               = <IP ADDRESS of the originator host>
HOSTNAME             = <hostname of the request originator host>
ADAPTER_HOST         = <hostname of the TME 10 Software Distribution server>
SEVERITY             = <severity of the alert>
DATE                 = <date and time of the alert>
MSG                  = <error message from TME 10 Software Distribution>
ALERT_LEVEL       = <FND00x> with x from 1 to 9
R1_REQUEST_ORIGINATOR_USER
R2_REQUEST_ORIGINATOR_HOST
R3_REQUEST_NUMBER
R4_REQUEST_RECURSION_NUMBER
R5_REQUEST_SUB_ID
R6_REQUEST_DESTINATION_HOST
```

The fields from R1_xxx to R6_xxx are filled only if a request that causes an alert is originated.

R5_REQUEST_SUB_ID is filled only when the request is related to a transmission plan. In this case each of the subrequests has an associated ID.

Appendix K. Setting Environment Variables

Before you can accomplish the tasks listed below, you must set the environment variables.

To set an environment variable permanently, put it in the etc/environment file and reboot the workstation. For example if you want to set COMM_TMO permanently to 70, code the following:

```
CONN_TMO=70
```

To set the environment variable through the command line enter:

```
export <variable_name>=<variable_value>
```

Setting Compression or Decompression at the Local Server or at the Destination Target

Set the **ACT_ON_TARGET** environment variable to choose whether the transmitted file is to be compressed or decompressed at your local server or at the destination target.

Valid values are **0** and **1**, with the following meaning:

0 The decompression is performed at your local server.

1 The decompression is performed at the destination target.

For targets that have TME 10 Software Distribution versions prior to 3.1 installed, this variable can only be set to 0. The default is **0**.

Setting a Connection Timeout

Set the **CONN_TMO** environment variable to establish a connection timeout. The connection timeout represents the number of seconds that the TME 10 Software Distribution server waits trying to connect to the TME 10 Software Distribution client before failing and issuing a warning message. You can specify a value from 10–75 seconds. The default is 10 seconds.

Setting a Graphical Interface Retry Time

Set the **GI_RETRY_TIME** environment variable to establish the timeout to display GUI. You must specify a value in seconds. The default is 40 seconds.

Increasing the Performance of the Database Synchronization Mechanism

Set the **cindex_sync_freq** environment variable to increase the performance of the database up to 60%. This variable effects the synchronization mechanism.

You must set the variable in the etc/environment file as follows:

Setting Environment Variables

```
cindex_sync_freq=deferred
```

In case of a power off, enter the **sync** command at least five times to preserve the database integrity.

Improving the Performance of the Target History Dialog

Set the **HIDE_HIST_CM_PB** environment variable to improve the performance of the Target History Dialog window. To cause the Target History Dialog window to be displayed faster, set the **HIDE_HIST_CM_PB** environment variable in the etc/environment file as follows:

```
HIDE_HIST_CM_PB=yes
```

If you do this, the CM functions push buttons are not displayed. To execute the CM functions go to the Catalog window.

Setting a Server to Server Permanent Connection

Set the **sts_perm_conn** environment variable to establish a permanent server to server (STS) connection and to start a task for each connection.

The variable can assume an alphabetic or a numeric value in milliseconds. For example, in the etc/environment file code:

```
sts_server_connection=yes
```

In this case if the process is inactive, the STS connection is up to the value specified in the sts_idle_time variable. The process checks each 3 seconds whether there are resources to process.

If you set the sts_server_connection variable to a numeric value, for example,

```
sts_server_connection=5000
```

the process checks if there are resources to process every 5000 milliseconds elapsed.

To save memory usage if you do not have a constant traffic rate, do not set the variable in the etc/environment.

Index

A

B

Communicating Your Comments to IBM

TME® 10 Software Distribution
for AIX®
Quick Beginnings
Version 3.1.5

Publication No. SH19-4333-02

If you especially like or dislike anything about this book, please use one of the methods listed below to send your comments to IBM. Whichever method you choose, make sure you send your name, address, and telephone number if you would like a reply.

Feel free to comment on specific errors or omissions, accuracy, organization, subject matter, or completeness of this book. However, the comments you send should pertain to only the information in this manual and the way in which the information is presented. To request additional publications, or to ask questions or make comments about the functions of IBM products or systems, you should talk to your IBM representative or to your IBM authorized remarketer.

When you send comments to IBM, you grant IBM a nonexclusive right to use or distribute your comments in any way it believes appropriate without incurring any obligation to you.

If you prefer to send comments by mail, use the reader's comment form (RCF) at the back of this book. If you wish, you can give the RCF to the local branch office or IBM representative for postage-paid mailing.

If you prefer to send comments by fax, use this number, which is in Italy:

39+06+596+62077

If you prefer to send comments electronically, use this network ID:

ROMERCF at VNET.IBM.COM

Make sure to include the following in your note:

- Title and publication number of this book
- Page number or topic to which your comment applies

Help us help you!

TME® 10 Software Distribution
for AIX®
Quick Beginnings
Version 3.1.5

Publication No. SH19-4333-02

We hope you find this publication useful, readable, and technically accurate, but only you can tell us!
Your comments and suggestions will help us improve our technical publications. Please take a few
minutes to let us know what you think by completing this form.

Overall, how satisfied are you with the information in this book?	Satisfied	Dissatisfied
	☐	☐

How satisfied are you that the information in this book is:	Satisfied	Dissatisfied
Accurate	☐	☐
Complete	☐	☐
Easy to find	☐	☐
Easy to understand	☐	☐
Well organized	☐	☐
Applicable to your task	☐	☐

Specific comments, problems, or suggestions:

_____

_____

_____

_____

_____

_____

Thank you for your response. When you send information to IBM, you grant IBM the right to use or
distribute the information without incurring any obligation to you. You of course retain the right to use
the information in any way you choose.

_____   _____
Name   Address

_____
Company or Organization

_____
Phone No.

Fold and Tape　　　　　　　　**Please do not staple**　　　　　　　Fold and Tape

Fold and Tape　　　　　　　　**Please do not staple**　　　　　　　Fold and Tape

IBM ®

Program Number: 5765-476

Distributed Electronically for Customer Printing