NORTEL NETWORKS

Preside Multiservice Data Manager

Configuration Management for DPN Administration

241-6001-304

Preside Multiservice Data Manager

Configuration Management for DPN Administration

Publication: 241-6001-304
Document status: Standard
Document version: 15.1RSUP
Document date: August 2004

Printed in Canada

Publication history

August 2004

15.1RSUP Standard
Commercial availability except for MPE support which will be available in a future release.

Contents

Chapter 6
Time Stream Management using keyed MCFs 79

Chapter 7
Service data conversion 85

About this document

This document describes how to administer and configure DPN Devices configuration software.

The following topics are discussed in this section:

- "Who should read this document and why" (page 13)
- "What you need to know" (page 13)
- "How this document is organized" (page 14)
- "What's new in this document" (page 15)
- "Text conventions" (page 15)
- "Related documents" (page 16)

Who should read this document and why

This document is intended for personnel configuring the DPN Devices configuration software, and for those who are responsible for network administration and operations. To install DPN Devices configuration software, see 241-6001-100 *Preside MDM Installation*.

What you need to know

To configure DPN Devices configuration software, you must be able to log on as the *root* user. Since this is a powerful user id, we assume that you are familiar with Sun workstations, the UNIX operating system, and X.25 network communications. You should be familiar with a UNIX editing facility (such as vi), so that you are able to modify files.

How this document is organized

241-6001-304 *Preside MDM Configuration Management for DPN Administration* contains the following sections:

- "DPN Devices configuration overview" (page 19) describes the components in DPN Devices configuration and how they interact.
- "DPN Devices configuration dependencies" (page 33) explains the dependencies DPN Devices configuration has with the other Preside Multiservice Data Manager (MDM) tools and the required software levels.
- "Installation and Configuration" (page 41) provides installation and configuration details for the provisioning related process and other dependent DPN network functions.
- "Service data backup" (page 69) describes the two modes of service data backup.
- "Software images" (page 71) describes the software migration and distribution processes and how they are used to upgrade the images on DPN-100 modules.
- "Time Stream Management using keyed MCFs" (page 79) describes Time Stream Management (TSM) as related to DPN Devices configuration.
- "Service data conversion" (page 85) describes when a service data conversion is performed.
- "Operational recommendations" (page 119) provides operational recommendations concerning the Component Provisioning tool and suggests some tips to help analyze and solve particular problems.
- "Uploading and downloading MCFs" (page 123) provides important information on uploading and downloading service data.
- "ICONS" (page 131) describes the Data Spooling Dump, PAGENT, and RDS ICONs.

What’s new in this document

There are no changes in this document for this release.

Text conventions

This document uses the following text conventions:

- `nonproportional spaced plain type`

  Nonproportional spaced plain type represents system generated text or text that appears on your screen.

- **`nonproportional spaced bold type`**

  Nonproportional spaced bold type represents words that you should type or that you should select on the screen.

- *italics*

  Statements that appear in italics in a procedure explain the results of a particular step and appear immediately following the step.

  Words that appear in italics in text are for naming.

- `[optional_parameter]`

  Words in square brackets represent optional parameters. The command can be entered with or without the words in the square brackets.

- `<general_term>`

  Words in angle brackets represent variables which are to be replaced with specific values.

- UPPERCASE,lowercase

  In MDM, uppercase and lowercase letters that appear in UNIX commands and parameters must be matched exactly. The system matches upper and lowercase characters differently.

- |

  This symbol separates items from which you may select one; for example, ON|OFF indicates that you may specify ON or OFF. If you do not make a choice, a default ON is assumed.

- ...

  Three dots in a command indicate that the parameter may be repeated more than once in succession.

The term absolute path name refers to the full specification of a path starting from the root directory. Absolute path names always begin with the slash (/) symbol. A relative path name takes the current directory as its starting point, and starts with any alphanumeric character (other than /).

Related documents

See the following documents for related information:

- 241-1001-303 *DPN-100 Operator Commands and Responses*
- 241-2001-102 *DPN-100 Network Control System User Guide*
- 241-2001-340 *DPN-100 Envelope Definitions*
- 241-6001-000 *Preside MDM Documentation Guide*
- 241-6001-012 *Preside MDM Configuration Management for DPN User Guide*
- 241-6001-022 *Preside MDM Network Reporting System User Guide*
- 241-6001-100 *Preside MDM Installation*
- 241-6001-101 *Preside MDM Engineering Guide*
- 241-6001-204 *Preside MDM DPN Provisioning API Reference Guide*
- 241-6001-209 *Preside MDM Provisioning Command Filter API for DPN*
- 241-6001-301 *Preside MDM Customization Administrator Guide*
- 241-6001-303 *Preside MDM Administrator Guide*
- 241-6001-310 *Preside MDM Server Reference Guide*

- 241-6001-308 *Preside MDM Network Configuration Database for DPN Administrator Guide*

Chapter 1
DPN Devices configuration overview

This chapter gives a brief overview of DPN Devices configuration. In this chapter, you can find the following information:

- “About Configuration for DPN Devices application” (page 19)
- “Provisioning capabilities” (page 20)
- “System overview” (page 22)
- “Configuration base” (page 22)
- “Configuration toolset” (page 26)

About Configuration for DPN Devices application

Configuration for DPN Devices lets you provision most DPN-100 services. It is recommended that the DPN Devices configuration application be used to provision all supported access services and DPN-100 modules. For more information on DPN-100 services, see 241-6001-012 *Preside MDM Configuration Management for DPN User Guide*.

The provisioning data is stored on DPN–100 module disks in the master configuration file (MCF) format. The format is the same as the service data. DPN Devices configuration uploads service data from the appropriate module, decompiles the service data from the MCF format into a user presentable format and displays it for update. The updated service data is compiled back into the MCF format and downloaded to the module, ready for activation.

Security is provided by controlling access to DPN–100 modules by means of NCS. Before the provisioning session can begin, the provisioning operator is authenticated by logging on to an NCS OA. The system then issues a command (through NCS) to the module which is to be provisioned. The command is accepted by the module if the operator has the appropriate capability. The module, in response to the command, originates a call to the provisioning system. All subsequent communication is carried out on the direct connection (not through NCS). The module stores the operator ID and capability set for the duration of the call. The module validates all commands received on the direct connection for appropriate operator capability.

The service data stored on the module's disk is backed up on the optional shadow disk and can also be backed up on the Backup disk. Module software can be delivered by means of the Remote Download Site (RDS).

The above architecture provides an online distributed provisioning system for DPN-100. The distributed nature of the system provides flexibility to partition the network for provisioning and allows unlimited growth in the network size.

Provisioning capabilities

Configuration for DPN Devices provides many provisioning functions integrated on a single platform. For more information on Component Provisioning, see 241-6001-012 *Preside MDM Configuration Management for DPN User Guide*.

Provisioning service data

The provisioning system allows you to query, change, delete, and add service data for the DPN-100 packet modules (PM). Access to this functionality is through the Component Provisioning and Envelope Editor tools.

Service data reporting

Service data reporting in Component Provisioning allows users to retrieve a file or a printed copy of the service data parameters that are present for any desired services. A user can produce a report for a single component or for a group of components, for example, all ports on a particular PI. The filtering capability allows for selection of components, data values or both for the report. The report output can be saved on an workstation disk for subsequent printing, post processing or for transferring to another host.

Provisioning activity audit trails

The provisioning activity performed by an operator is logged by the on–switch logging facility. All activities are logged with the user identification who performed the activity and the detail of the activity. The events reported are the creation and deletion of MCF files on the module. These events are logged in the standard DPN–100 log stream.

Time stream management

Time stream management (TSM) allows users to manage multiple sets of service data (views) on every module. TSM is not an application but a series of recommendations that provide a mechanism to select the committed and activated views as the basis for new changes. This allows flexibility for the user to name service data views using mnemonics and dates. The bundle id in the MCF file naming convention is used to contain the user selected mnemonic date allowing a user to coordinate views across multiple modules.

Delete, cut, copy, and paste facilities

Delete, cut, copy and paste allows a user to remove (delete), duplicate (copy and paste), and move (cut and paste) service data for a component and all its subcomponents. The components can be copied or moved with the same key values (component names) or with new user-supplied key values. The facility is supported across multiple provisioning sessions on the same workstation which allows service data to be moved between modules.

Templating

Templating in Component Provisioning provides the capability to take snapshots of service data which can then be used to quickly generate a number of similar service data components. For details on Templating and Template Management, see 241-6001-012 *Preside MDM Configuration Management for DPN User Guide*.

Custom Forms

Custom Forms in Component Provisioning allows you to customize the Edit/View form for a subcomponent. The Custom Form saves both the service data and how that service data is displayed in the Edit/View form. For details on Custom Forms, see 241-6001-012 *Preside MDM Configuration Management for DPN User Guide*.

Network service integrity

Configuration for DPN Devices has extensive semantic checking to detect configuration problems prior to service data activation. These checks reduce the possibility of service disruption which is key in preventing network problems especially for large networks. This ensures the integrity of the networks, improves service quality and reduces operational costs.

System overview

Configuration for DPN Devices can be split into two categories of functionality.

- The Base provides the infrastructure for implementing the core of the overall application architecture.
- The Toolset provides the functionality which allows for definition and validation of service data for DPN-100 modules.

For a view of the DPN Devices configuration components, see the figure “Configuration for DPN Devices components” (page 24). This document is not intended to explain all the functionality in detail. See 241-6001-012 *Preside MDM Configuration Management for DPN User Guide* for more information on each of the applications.

Configuration base

The configuration base provides an object oriented application environment for a data driven provisioning system, server processes for accessing and manipulating the service data and the utilities for interprocess communications.

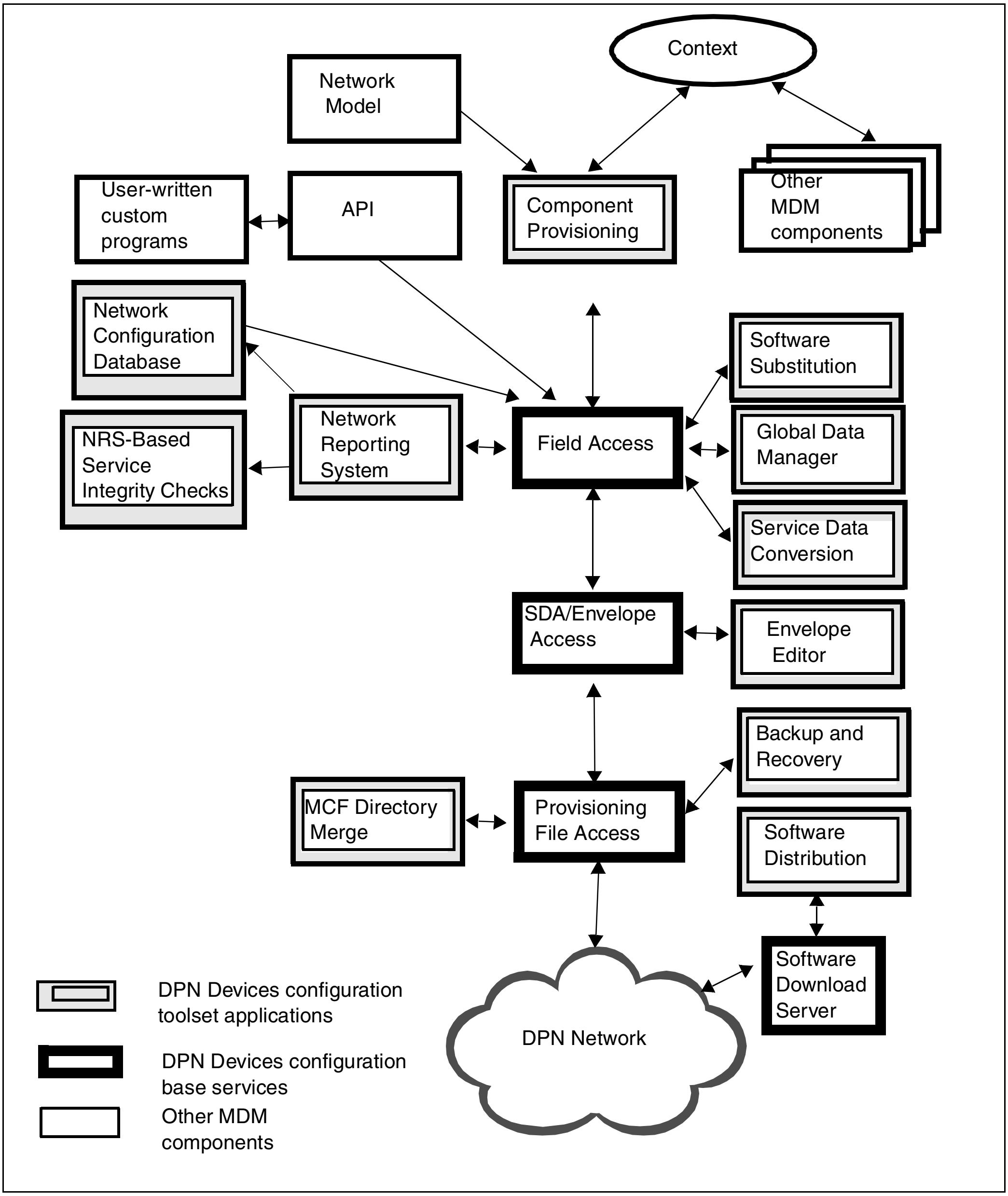
Object oriented environment

The provisioning system is designed to be a data driven system with the service data description defined as data. This approach was chosen to speed the delivery of the provisioning capability for new access services and the modification of existing services. The object oriented base provides a generalized mechanism to define the service data description.

Server processes

Configuration for DPN devices is built around the following core server processes: Provisioning File Access (PFA), Software Download (SWDLD), Service Data Area (SDA) and Envelope Access (SEA), and Field Access (FA). These processes allow access to DPN-100 module service data at different levels of granularity.

Figure 1
Configuration for DPN Devices components



Provisioning File Access

Provisioning File Access (PFA) is a server that provides access to DPN-100 packet modules for the purpose of retrieving and modifying service data files from DPN-100 modules or NMS disk.

In general, all service data resides in MCFs on the DPN-100 module. The action of retrieving an MCF from the module is called *uploading* and the action of sending an MCF to the module is called *downloading*.

In order to establish a virtual circuit or communication link between an Configuration application and the DPN-100 module, PFA interacts with the NCS interface to initiate a communication link to a DPN-100 network module. This link allows the retrieving and sending operations to take place. This process also implements the concurrency control by interacting with the PAGENT icon on the module.

Software Download

The Software Download server is a separate PFA server which is run with a different service name. This server is used to distribute DPN software images to the DPN-100 modules.

Service Data Area and Envelope Access

Service Data Area (SDA) and Envelope Access (SEA) is a server that organizes service data into SDAs and envelopes. This server uses PFA to retrieve and send MCFs. Other provisioning applications such as the Envelope Editor and MCF Directory Merge use data supplied by this process.

Field Access

Field Access (FA) is a server that provides client processes with service data at field level. FA uses the SEA server to retrieve the envelope information it requires. FA also checks the service data to ensure that all field elements have the proper data formats and are consistent with other related service data fields. This process is called *semantic validation* of service data.

Inter-process communication

A well defined application protocol has been implemented to permit the various server processes to communicate with their clients. This protocol includes a component naming strategy allowing each and every service data element to be uniquely named. All provisioning processes communicate with each other using this naming strategy. It also provides a transaction oriented messaging model and support facilities to allow the Provisioning File Access server, SDA and Envelope Access server, and Field Access server to communicate with each other:

Configuration toolset

The Configuration toolset provides a comprehensive set of applications for the manipulation of service data, from the field level to the file level. The following tools are available from the Preside MDM window:

- Component Provisioning
- Global Data Manager
- Administration
  - Service Data Backup
  - Service Data Restore
  - Software Distribution
  - Software Substitution
  - Service Data Conversion
  - Envelope Editor
  - Network Activation
- Network Reporting System
  - Service Integrity Audit
  - Configuration Reports
  - Configuration Differences
- Inventory Reports

The following tools are available through a command line:

- MCF Directory Merge
- MCF Management
- NRS-based Service Integrity Checks
-

Depending on the provisioning function being used, some or all of the above components are necessary. See the figure “Configuration for DPN Devices components” (page 24).

Component Provisioning

The Component Provisioning tool is a graphical user interface that is used to access and manipulate service data that resides on the DPN-100 modules.

Component Provisioning allows you to perform the following:

- access and navigate through the service data hierarchy
- edit or view service data
- verify the sanity of edited service data
- download edited service data
- manage several service data views, including the committed and activated MCFs
- generate service data reports
- create propagation log files
- access templates
- create custom component provisioning forms

Global Data Manager

The Global Data Manager (GDM) tool provides a mechanism whereby certain global service data components can be duplicated from one MCF to one or more MCFs in the network. It allows you to provision a specific *master* MCF as a source of service data envelopes to be copied to one or more target MCF(s). The Global Data Manager tool is used to distribute network data across the modules in your network.

Service Data Backup

The Service Data Backup tool is a graphical user interface application that provides facilities to create backups for MCF sets either on a Backup disk or on the DPN-100 module. It can be configured to backup automatically when MCFs are downloaded, or manually. When the Backup disk is selected as the backup target, the tool triggers the dumping process that transfers backed up MCFs from the DPN-100 module to the Backup disk. The *Dump Start* command overrides the dump schedule at the DPN-100 module.

Service Data Restore

The Service Data Restore tool is a graphical user interface application that provides facilities to retrieve backup MCF from the Backup disk and restore the MCFs to a specific DPN module. It also allows you to delete obsolete MCF files from the backup source.

Software Distribution

The Software Distribution tool provides the capability to download software images from a designated NMS disk, which is referred to as a Software Distribution Site (SDS), to a DPN-100 module. It also provides an easy interface to issue commands to a DPN-100 module to copy images from the Remote Download Site (RDS).

This tool can be used either by means of the graphical user interface or by the command line in a UNIX shell.

Software Substitution

Software Substitution is an application used to upgrade images from an older release to a newer release on DPN-100 modules. This tool can be used either by means of the graphical user interface or by a command line in a UNIX shell.

Service Data Conversion

The Service Data Conversion tool allows you to convert service data from one generic to another, one MCF at a time. The new service data can then be used with the new switch software. This tool allows a service data conversion for each new main release.The service data level is represented by the SD_version. This tool can be used either by means of the graphical user interface or by a command line in a UNIX shell.

Envelope Editor

The Envelope Editor tool is a graphical user interface application that is used to alter and create service data envelopes or SDA headers. This tool displays service data in ASCII hex or binary format. In the Envelope Editor you can perform a number of operations on service data. These operations are:

- listing service data envelopes in an MCF
- retrieving service data for viewing or editing purposes
- adding a new service data envelope or SDA header
- deleting service data envelopes or SDAs
- replacing service data

Network Activation

The Network Activation tool simplifies and automates the activation process for multiple nodes in a network. The activate and commit operations can be performed interactively or in batch mode. See 241-6001-012 *Preside MDM Configuration Management for DPN User Guide* for additional information.

Network Reporting System

The Network Reporting System (NRS) provides the capability to extract service data from DPN-100 modules on the network and to store all the data in one central repository. The extracted service data is in ASCII format and may be manipulated to produce customized comprehensive reports. See 241-6001-022 *Preside MDM Network Reporting System User Guide* for more information.

Inventory Reports

The Inventory Reports tool lets you report on the hardware and software configuration of selected devices in your network. See 241-6001-808 *Preside MDM Device Inventory Tools User Guide* for more information

MCF Directory Merge

MCF Directory Merge is a UNIX command line application used to merge a selected number of MCF directory files into a new MCF directory file. By merging the files into a single directory, more files than the three allowed by the on switch tidy command can be maintained. For more information on the *tidy* command, see 241-1001-303 *DPN-100 Operator Commands and Responses*.

MCF management

MCF management is a set of UNIX utilities used to help manage MCFs on PMs and NMS disks.

It provides commands to check, list, and delete MCFs on the Preside Multiservice Data Manager (MDM) workstation.

NRS-Based Service Integrity Check

NRS-Based Service Integrity Check (NSIC) is a UNIX command line application used to detect errors in the service data which could cause problems during network operation. These checks are not done by the Component Provisioning tool because they require access to different PMs. Typically these checks are run after the data has been added using Component Provisioning but before activating the service data. For more information on NSIC, see 241-6001-022 *Preside MDM Network Reporting System User Guide*.

Network Configuration Database

The Network Configuration Database (NCD) facilitates DPN module provisioning capability which provides notification of non-unique data values for components which require network wide uniqueness. In addition notification of references to non-existent DNAs is provided.

Validation of these components is performed using read-only access to an NCD database which is populated from existing MCFs and/or Passport viewfiles by the NCD Populator (*ncdpop*). Access to the NCD database is provided by the NCD Server (*ncdsvr*). See 241-6001-308 *Preside MDM Network Configuration Database for DPN Administrator Guide* for more information.

DPN Provisioning API

The DPN Provisioning Application Programming Interface (API) is an ASCII interface that provides access to Architect's Component Provisioning information. Customer-written applications access Component Provisioning through the DPN Provisioning API. It allows you to create, view, and modify service data. The API responds with one or more response messages, error messages, or both.

See 241-6001-204 *Preside MDM DPN Provisioning API Reference Guide* for more information.

Chapter 2 DPN Devices configuration dependencies

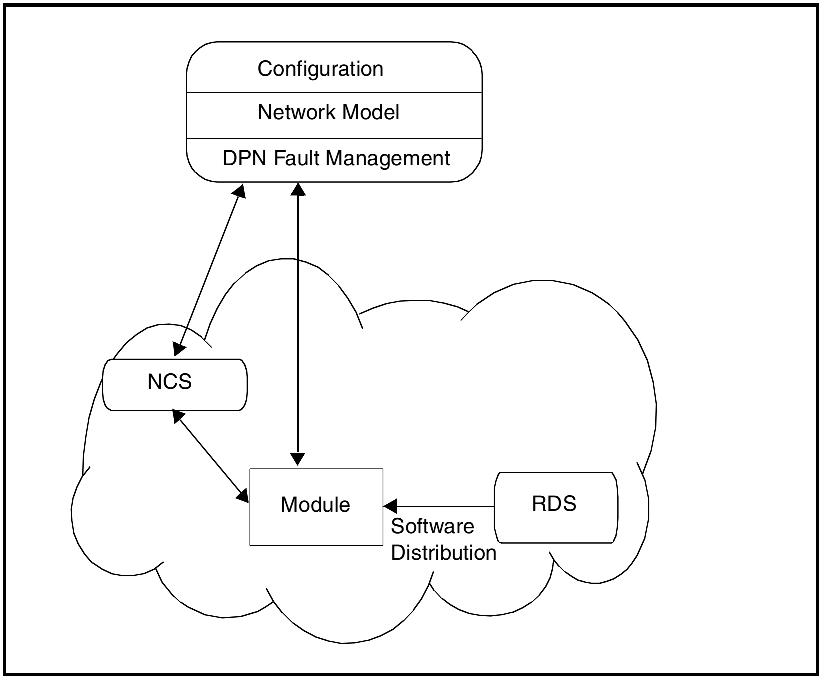
This chapter describes the interactions of Configuration for DPN Devices with other Preside Multiservice Data Manager (MDM) workstation functions. In this chapter, you can find the following information:

- “About Configuration for DPN dependencies” (page 33)
- “Network model interaction” (page 34)
- “Network Control System” (page 35)
- “On-switch network management functions” (page 37)

About Configuration for DPN dependencies

Configuration for DPN Devices is integrated into the Preside Multiservice Data Manager (MDM) workstation. It interacts and uses several other MDM workstation functions. See the figure “Configuration dependency overview” (page 34) for an overview.

Figure 2
Configuration dependency overview



Network model interaction

Configuration for DPN Devices can be run with a valid network model, a manually updated network model or without a network model.

If Configuration for DPN Devices is running with a network model which does not contain a packet module, a message is displayed in the *Messages* area specifying that the PM <mnemonic> does not exist in the network model.

If Configuration for DPN Devices is running without a network model, that is the Network Model Update server is not running, then a window message is displayed after authentication. The message specifies that the network model cannot be accessed and *Continue* must be selected to proceed without the network model.

Configuration semantics have been added to Component Provisioning when running Configuration for DPN Devices without a network model. These additional checks will ensure that the service data is consistent with the hardware configuration requirements.

It is recommended that all component changes (add/delete modules, PEs, PIs, and Ports) be reported into the network model. This can be done as follows:

- start the Command Console tool and use the MCDF command, or
- use the Network Viewer tool.

Network Control System

Configuration for DPN Devices uses the Network Control System (NCS) for provisioning operator authentication, domain control, and to route the call command to the PAGENT icon process on DPN-100 modules.

There are no specific configuration requirements for NCS. Configuration for DPN Devices interacts with NCS through the NCS application called Management Data Interface (MDI) and supports multiple virtual circuits, (VCs). Provisioning operators can share the same MDI defined for DPN Fault functions. When operating with DPN-100 NCS at generics prior to G29, the IWSIF supports only one VC and therefore, additional IWSIFs must be defined for the provisioning applications to call NCS.

Security is controlled by NCS capability sets that are configured in each NCS operations agent (OA). Each capability set has an ID and a password, and also contains the specification of capability or privileges allowed for the operator using the ID. Modules to be provisioned must be in an operator’s domain. Depending upon a customer’s organization, a separate NCS OA hierarchy for provisioning purposes may be created to keep the provisioning and Advisor operators separate. See the table “NCS minimum capability sets” (page 36) for a list of NCS capability sets.

Operators using only NAMS, OA, Config capability, do not have the capability to execute any NCS command console commands.

Table 1
NCS minimum capability sets

	NAMS			Switching		
Operation	**Network**	**OA/Device**	**Appl/Line**	**Network**	**Device**	**Line**
Service Data Backup tool	None	Privileged	None	None	None	None
To populate service data to NRS	None	Passive	None	None	Passive	None
To view service data	None	Passive	None	None	Passive	None
To view service data	None	Passive	None	None	None	None
To update and download service data	None	Config	None	None	None	None

When a backup OA is created it is usually given the same name as the primary OA, except *-B* is added to the end. For example, *OANAME -B* is the backup for *OANAME*. DPN-100 NCS supports path routing to the module when the backup OA becomes the primary OA. Configuration for DPN Devices allows the use of this implicit path routing to access a module. However, Configuration for DPN Devices does not support explicit naming of a path route that includes a backup OA name with the form *<OANAME> -B*.

NCS Communication Manager

Configuration for DPN Devices also requires a server process called NCS Communication Manager (NCSMGR). Provisioning File Access server (PFAS) interacts with NCS through this process. The NCSMGR process must be configured on the same MDM workstation as PFAS. The NCSMGR process must be set up with the Server Manager Administration tool such that the process is automatically started. PFAS shares the NCSMGR process defined for the DPN fault management. For a description of the NCSMGR server, see 241-6001-310 *Preside MDM Server Reference Guide*.

NCS OA destination definition

The Host Group Directory server configuration file, /opt/MagellanNMS/cfg/HGDS.cfg, is used to specify the NCS operations agent available to the NCS communication server manager. Configuration for DPN Devices t can use the destination mnemonics defined by the OA group for NCS access. However, if a separate hierarchy of NCS OAs is created for provisioning, the configuration file must be updated with the Preside Multiservice Data Manager (MDM) workstation interfaces defined on these NCS OAs.

On-switch network management functions

DPN-100 modules must be running G32 or later and must support the following:

- Provisioning Agent system (PA) for support of DPN Devices configuration
- Remote Download Site (RDS) for support of Software Distribution
- DPN-100 Data Spooling and Dumping for the support of Service Data Backup and Service Data Restore
- module disk directory size must be adequate

Provisioning Agent system

DPN Devices configuration interfaces with the Provisioning Agent system (PA) through a switched virtual circuit (VC) to the PAGENT icon to manage service data files on DPN–100 module’s disk. It also supports other functions like concurrency, capability validation, generation of audit logs for the provisioning activity, and processing the *Call* command to originate a call to a provisioning MDM workstation. PA supports multiple VCs to allow multiple users concurrent access to a module for provisioning. PA is configured on the module’s office PE; therefore, no additional hardware is required. The PAGENT icon is provisioned by DPN Devices configuration the same as any other icon.

The additional load on the PE caused by a provisioning session is dependent on the amount of service data in the MCF. Typically, the additional load is very small (less than 5% of the load on the PE) and can be ignored.

There are two versions of the PA: one for the PE386 office images and one for the PE286 images. The PE386 version supports a maximum of seven simultaneous VCs to the provisioning system and provides concurrency support. The concurrency control allows only one user to be in update mode at a time and up to six additional users can only view the service data on the module.

The PE286 version supports only one VC and does not support concurrency. The differences are transparent to the user. The DPN-100/1 image, even though it is PE386 based, contains the PA which supports only one VC. The SAM PE286 images do not contain the upload facility in the PA and therefore DPN Devices configuration cannot provision PE286 DPN-100/10 or PE286 DPN–100/5s.

The table “Office image functionality for DPN Devices configuration” (page 38) provides the supported functionality for each office image.

Table 2
Office image functionality for DPN Devices configuration

Image	PEs supported	Provisioned from Configuration for DPN Devices	Provisioning sessions supported
AMOFCMGR	286	Yes	One
AMOFCNLL	286	Yes	One
AMOFCNLU	286	Yes	One
AMOFF386	386	Yes	Multiple
RMOFFICE (G29)	286	Yes	One
RMOFF386	386	Yes	Multiple
SAMXSIA	386	Yes	Multiple
SAMXSIP	386	Yes	Multiple
SAMXISIT (G30)	386	Yes	Multiple
SAMXI (G29)	286	No	None
(Sheet 1 of 2)			

Table 2 (Continued)
Office image functionality for DPN Devices configuration

Image	PEs supported	Provisioned from Configuration for DPN Devices	Provisioning sessions supported
SAMX (G29)	286	No	None
SAMI	286	No	None
SAMSI (G29)	286	No	None
SAMSOA	386	Yes	Multiple
SAMCRDU	386	Yes	Multiple
DPN-100/1	386	Yes	One
(Sheet 2 of 2)			

DPN–100 data spooling and dumping

The DPN–100 data spooling and dumping system is used for backing up the service data files to the Backup disk using the *mcfcol* program.

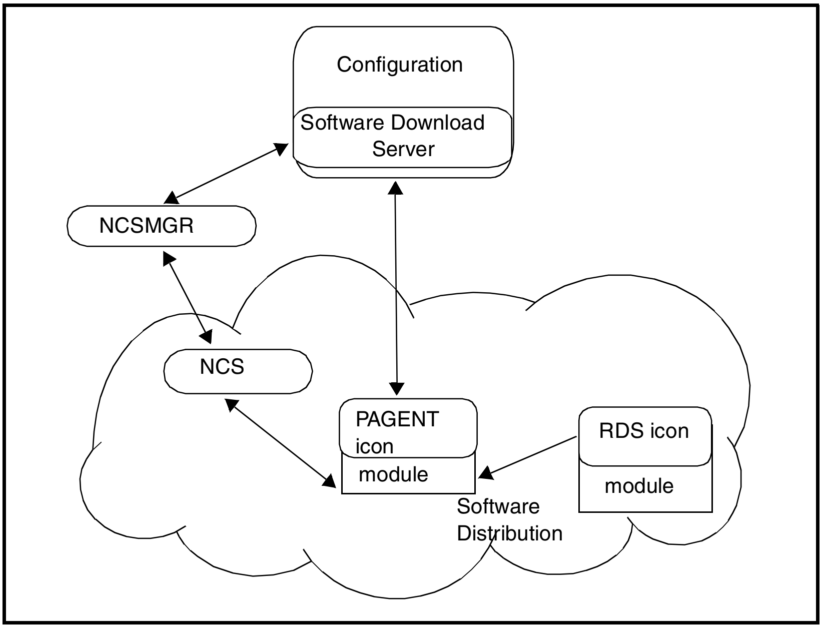
Module disk directory size

The module disk directory size defaults to 1024. This allows for larger numbers of service data files. DPN Devices configuration does not increase the number of files. However, because of the online nature of Configuration for DPN Devices, it is possible that users will make more frequent small changes as opposed to making a large number of changes and then download. In this case, the number of files on a module disk may be more than before. Ensure that the disk directory size is appropriately set if making frequent service data changes. Alternatively, more frequent *tidy* operations on the module's disk may be necessary.

Remote Download Site

The Remote Download Site (RDS) is a module designated to provide software to other modules. Software can be loaded onto the RDS with the Software Distribution tool. You can use the Software Distribution tool to download images from the SDS to the RDS. A module that requires access to the RDS using the Software Distribution tool must have its PAGENT icon provisioned with a direct call to the RDS. See “ICONS” (page 131) for PAGENT icon setup. See the figure “Remote Download Site (RDS)” (page 40) for the RDS setup.

Figure 3
Remote Download Site (RDS)



Chapter 3
Installation and Configuration

This chapter provides installation and configuration details for provisioning related processes and the other dependent DPN network management functions. Configuration for DPN Devices software is delivered on the same tape with other Preside Multiservice Data Manager (MDM) workstation applications and is installed using the same procedures and tools as other Preside MDM software.

In this chapter, you can find the following information:

Hardware requirements

Configuration for DPN Devices runs on a standard Preside Multiservice Data Manager (MDM) workstation hardware platform. See 241-6001-100 *Preside MDM Installation*, for MDM hardware requirements and 241-6001-101 *Preside MDM Engineering Guide*, for memory and swap space requirements.

Software requirements

Configuration for DPN Devices runs with standard Preside Multiservice Data Manager (MDM) workstation software. See 241-6001-100 *Preside MDM Installation*, for MDM software requirements.

If you want to distribute the DPN software from the NMS disk to the module, you must order the DPN software for the MDM workstation. See the *Software Ordering Catalogue* for more information.

Installation configuration checklist

The following steps are required to deploy Configuration for DPN Devices to be able to provision DPN AM/RM modules.

1 Install the Preside Multiservice Data Manager (MDM) software. See 241-6001-100 *Preside MDM Installation* for more information.

2 Configure Provisioning File Access server (PFAS) on each of the standalone provisioning workstations and on a server workstation for a LAN configuration. See “Provisioning File Access server customization” (page 43).

3 Use the Server Administration tool to start up the necessary processes for the MDM workstation. See NCS Communications Manager in 241-6001-310 *Preside MDM Server Reference Guide* for more information.

4 Define provisioning users, user groups and assign appropriate tool sets.

5 If necessary, add or remove any components in the network-wide data files. See “Defining network-wide data” (page 47) for more information.

6 If necessary, add NCS OAs for provisioning NCS hierarchy. Define Management Data Interface (MDI) and capability IDs for DPN-100 NCS at G32 or later.

7 Upgrade DPN-100 modules to be provisioned from the MDM workstation to G29 and later.

8 Define PAGENT icon and DPN-100 Data Spooling on these modules.

9 Distribute DPN software images (copy/download). Use the Server Administration tool to start up the software download server. See “Defining network-wide data” (page 47) for more information.

Configuring after installing

Once you have installed the Configuration for DPN Devices software, follow the post-installation configuration procedures below. You will not be able to access the toolset unless the following procedures are performed. PFAS customization requires, as a minimum, that the DNA of the X.25 port on the Preside Multiservice Data Manager (MDM) workstation being added to the HOST_ADDRESS in the configuration. You cannot access Configuration for DPN Devices until this is done.

> ***Note:*** Before modifying any .cfg file, the file must be copied from the /opt/MagellanNMS/lib/cfg directory into the /opt/MagellanNMS/cfg directory. Files can only be modified in the /opt/MagellanNMS/cfg directory.

Toolset customization

This step is optional. The DPN Devices configuration toolset is automatically added to the default Preside Multiservice Data Manager (MDM) workstation toolset definition file upon installation of the MDM software. See 241-6001-301 *Preside MDM Customization Administrator Guide*, for details on how to customize the default toolset.

If desired, the toolset may be added to any other MDM toolset definition file other than the default comprehensive toolset, which is defined in /opt/MagellanNMS/lib/tsets/Full.tsets. This is done by appending or merging the contents of /opt/MagellanNMS/lib/tsets/dpnarch.tools file with any other toolset definition file using an editor such as vi.

Provisioning File Access server customization

The DPN Devices Configuration system requires a Provisioning File Access server (PFAS) process on a workstation with an X.25 link to the network. In a standalone workstation configuration, the workstation requires an X.25 link and PFAS. In the LAN configuration the X.25 link and a PFAS process is required on at least one Preside Multiservice Data Manager (MDM) workstation in the LAN. MDM provisioning client processes on all MDM workstations in the LAN can share the same PFAS. The PFAS process must be defined in the Server Manager Administration tool (SVM), such that the process is automatically started at workstation start up. See 241-6001-303 *Preside MDM Administrator Guide* for the details on SVM.

The following procedure contains steps for setting PFAS configuration parameters by editing the file /opt/MagellanNMS/cfg/PFA.cfg.

> ***Note 1:*** This procedure must be followed on the workstation that hosts the X.25 line to the DPN network.
>
> ***Note 2:*** When configuring X.25 parameters for the workstation, the *incoming reverse charging* facility must be set. This is required since calls from the provisioning agent system have this facility set.

How to configure PFAS

1 Log on as the *root* user by using the *su* command in a UNIX window.

A UNIX root prompt appears (#).

2 Edit the file /opt/MagellanNMS/cfg/PFA.cfg to change any default values for the server configuration parameters. See the table “PFAS configuration parameters” (page 45) for format and arguments.

3 Enter each new parameter on a new line. Arguments must be preceded by *DA_SERVER*.

For example:

```
DA_SERVER RECV_TIMEOUT 60
```

The above example sets the waiting time for incoming data to 60 seconds. If this parameter is left unchanged, the waiting time defaults to 30 seconds. This command is case sensitive.

4 Save the file.

Table 3
PFAS configuration parameters

Line format for DA_SERVER		
<parameter>	**<value range>**	**<default>**
OMDATA_FILE	<pathname of OM data file>	OM file not generated
RECV_TIMEOUT	30-600	30
CALL_WAITTIME	1-10	10
MAN_WAITTIME	1-60	30
NCS_HOST	<UNIX host mnemonic where NCSMGR is running>	
HOST_ADDRESS	<X.121 host address>	
AUTO_BACKUP	<ON or OFF>	OFF
AUTO_DUMPSTART	<ON or OFF>	OFF
MODULE_BACKUP_TO_DISK	<ON or OFF>	OFF
UNIX_ROOTDIR	<pathname of service data directory>	PFAS start-up directory
UNIX_BACKUP_DIR	<full directory path, e.g. /opt/MagellanNMS/data/B_MCF>	NMS DISK
SEMAPHORE_LOCK	<ON or OFF>	ON

Parameter descriptions

OMDATA_FILE   creates a file that will contain file transfer measurement information. Measurement data is appended to the file and the operator is responsible for cleaning up disk space. If this parameter is not defined, the OM file is not generated.

RECV_TIMEOUT indicates the time in seconds to wait for incoming data. It is recommend that this parameter be set to a high value when data spooling occurs on the module.

CALL_WAITTIME defines the time in seconds to wait for the incoming calls from the PA in auto mode - access NCS by means of the NCSMGR process.

MAN_WAITTIME defines the time in seconds to wait for the incoming calls from the PA in manual mode. In this mode there is no NCS access and the user must manually establish connection to PFAS by issuing the CALL command within the specified timeout.

NCS_HOST defines the UNIX host where the NCSCOM server is running. If not defined, the host which is located by MNS daemon will be used.

HOST_ADDRESS defines the address of the port that is to be used by the PFAS server for communicating with the PAGENT icon. This parameter must be defined for the provisioning system to operate, since it is used by the PA ICON when establishing connection to the provisioning system.

AUTO_BACKUP specifies whether automatic backups are made for each service data file downloaded by Configuration for DPN Devices.

AUTO_DUMPSTART specifies whether automatic dump start of backup MCF files after download is enabled/disabled.

MODULE_BACKUP_TO_DISK specifies whether the backup MCF to disk enabled or disabled. It is only used when AUTO_BACKUP is set to ON.

UNIX_ROOTDIR defines the directory that will be used to upload and download service data files when *NMS DISK* is the specified location. Ensure that the PFAS server is allowed both read and write access to this directory.

UNIX_BACKUP_DIR defines the directory on the UNIX disk where the backup MCF is stored.

SEMAPHORE_LOCK determines whether or not PFAS will request a lock from the provisioning agent. When this parameter is set to OFF, locking requests to the provisioning agent will be ignored.

Defining network-wide data

Some components must have attribute/key values that are available or unique throughout the entire network (i.e., network-wide data). The Component Provisioning tool places a pixmap beside components containing network-wide data in order to enhance their visibility. You can specify a pixmap other than the default through a resource. See the table “Resources that can be customized in Component Provisioning” (page 67) for the resource name and description.

The components with network-wide data are listed in two files, one for DPN components and one for Passport components. You can define components other than the defaults as network-wide data if you wish. Unrecognized components in these files are ignored. You can also specify a file name other than the default values through two resources. See the table “Resources that can be customized in Component Provisioning” (page 67) for the resource names and descriptions.

> ***Note:*** Because they are configuration files, the network-wide data files are located in /opt/MagellanNMS/lib/cfg.

Since the network-wide data files are loaded at initialization, you must exit Component Provisioning in order to activate any changes. Also, since these files are stored on the workstation disk, any changes are local to that workstation.

Each component in the network-wide data files is specified by its UI name. There are both keyed components, where the key is integrated into the UI name, and non-keyed components. This gives you two choices when you specify components as network-wide data: you can specify the component or you can specify the component only when its key is a particular value.

Example

The following is an example of entry formats in a network-wide data file. Notice that the file supports comments, specified by a “#” at the beginning of the line, and blank lines.

```
# DPN_Net_Wide_Data.dat
# This file lists all the components to be flagged as
# network-wide data.

# Non-keyed component (always flagged).
AccessControl

# Keyed component that is always flagged.
Userid

# Keyed component that is flagged depending on the key.
# These entries are not required because the entry
# Userid causes all the components of this type to be
# flagged.
Userid/NOFTP
Userid/THUONG

# Keyed component that is flagged depending on the key.
# These entries flag the components
# Collector/ACCOUNTING and Collector/SCN.
# The components Collector/ALARM, Collector/LOG,
# Collector/DEBUG, Collector/STATS, and Collector/TRAP
# are not flagged.
Collector/ACCOUNTING
Collector/SCN
```

Software download server

Designate a Preside Multiservice Data Manager (MDM) workstation for a software download server to run on. This MDM must have an X.25 link connected to it and should have an NCSMGR process running as well. The directory where the DPN software release is located, /opt/MagellanNMS/cfg/dpn_img, becomes the software download site when the software download server is running.

Using the Server Administration tool, add the software download server to the list of processes managed on the MDM. By default, the file /opt/MagellanNMS/cfg/PFA.cfg is used by both PFAS and the software download server. Optionally, an alternate configuration file can be specified using the -*c* option. The start-up command for the server is:

```
/opt/MagellanNMS/bin/pfas -n swdld [-c <configuration
  file>]
```

An optional configuration file distinguishes the configuration parameters used by PFAS from those used by the software download server. For example, AUTO_BACKUP could be set to OFF for the software downloaded server and ON for PFAS. This would avoid the automatic backup of software images during download.

> ***Note:*** An optional configuration file can be obtained by editing a renamed copy of the file PFA.cfg.

The following command is used to manually start-up the server on the Software Distribution Site if SVM is not used:

```
/opt/MagellanNMS/bin/pfas -n swdld [-c <configuration
  file>] &
```

Using a multi-nodal domain server

The PFAS process provides a link from the workstations running the Configuration for DPN Devices application to the DPN network. See “DPN Devices configuration overview” (page 19) for a description of the components. By default each workstation that has Configuration for DPN Devices software installed on it will also have PFAS installed. By default, then, each workstation has a direct link to the DPN network via PFAS and an X.25 link.

The workstation can be configured using the multi-nodal name server so that only one workstation provides a link to the DPN network. Therefore, all other workstations running Configuration for DPN Devices connect directly to the workstation running PFAS. This permits multiple workstations to make use of a single X.25 link. If this is desired, see “Using a multi-nodal name server” (page 49) for more information.

Using a multi-nodal name server

To configure PFAS to run on only one workstation, use the Server Administration tool to stop the provisioning file access process on the local workstation.

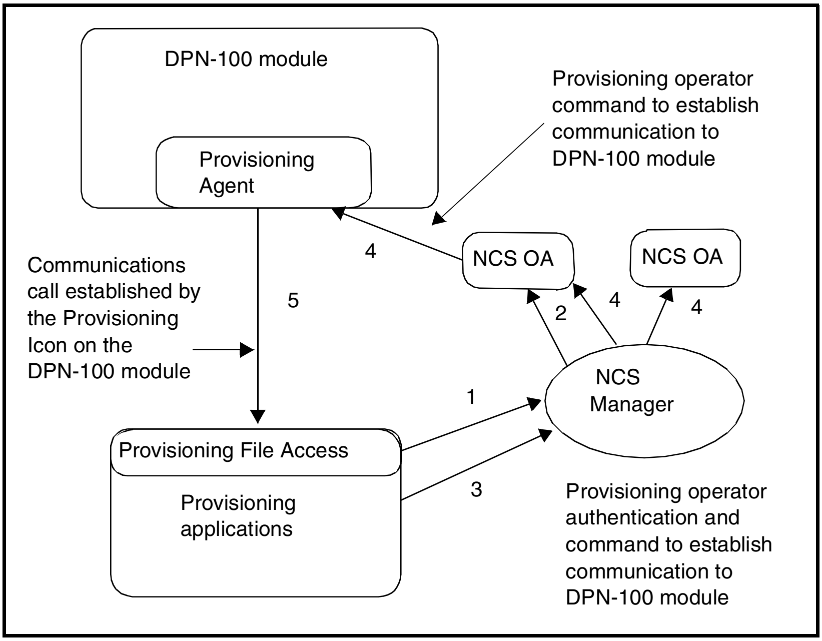
Make sure that there is one Level 2 MNS running on the LAN. This Level 2 MNS will then search through the LAN to find another PFAS to connect to the one which the Preside Multiservice Data Manager (MDM) workstation has the X.25 link to the module. See 241-6001-303 *Preside MDM Administrator Guide* for a complete description of Multi-nodal Name Server capabilities.

Configuring communication links to DPN-100 modules

To be able to establish calls to DPN-100 modules, the capability id must be configured in the NCS OA with a capability set that includes NAMS functions, device scope, and a minimum of *config* capability to modify the service data and *passive* capability to query service data. See 241-2001-102 *DPN-100 Network Control System User Guide* for complete details on defining and configuring NCS operators.

A PAGENT icon must be defined for all DPN-100 modules that will be provisioned from Configuration for DPN Devices. A direct call must be defined on the PAGENT icon to the RDS site for each module that supports Software Distribution. For more information on provisioning these icons, see “ICONS” (page 131).

Figure 4
Configuration for communicating with DPN-100 modules


DPN-100 module
Provisioning Agent
Provisioning operator command to establish communication to DPN-100 module
4
NCS OA
NCS OA
Communications call established by the Provisioning Icon on the DPN-100 module
5
2
4
4
NCS Manager
1
Provisioning File Access
Provisioning applications
3
Provisioning operator authentication and command to establish communication to DPN-100 module

The figure “Configuration for communicating with DPN-100 modules” (page 51) illustrates the sequence of calls to the Provisioning Agent System of the module:

1 authentication is performed through the NCS manager.

2 authentication is performed between NCS manager and NCS OA.

3 command to establish connection with PA is sent to NCS manager.

4 command is sent to NCS and is routed to the PA on the target DPN-100 module.

5 call is established from the PA to PFAS.

Configuring SunLink X.25 parameters

This section describes the X.25 port parameters that must be properly configured to ensure that the X.25 port properly connects the Preside Multiservice Data Manager (MDM) workstation to the DPN switch. The

X.25 parameter file is used by SunLink X.25 to define the port characteristics. See *SunLink X.25 System Administration, SunLink Multiprotocol Communication Processor Software Installation and Configuration Guide* and *SunLinkX.25 Reference Guide*. See the table “Interface and X.25 parameters” (page 53) for the recommended SunLink X.25 parameters.

X.25 port

The CUGs on the X.25 port must be defined to match the CUG on the Provisioning Agent. If the CUG on the X.25 port is not properly defined, the call setup from a provisioning agent will be rejected. The port can also be defined to allow non-CUG calls. Non-CUG calls occur if a CUG is not defined on the provisioning agent.

The packet size on the X.25 port must be defined to be at least 256 if the port is to be used for accessing NCS.

X.25 parameter

The parameters in the X.25 link must be defined to match the configuration of the port on the DPN switch that the Preside Multiservice Data Manager (MDM) workstation is connected to.

It is strongly recommended that you use the SunLink X.25 Administration Tool (x25tool) to modify the X.25 parameters. This tool performs the necessary consistency checks on each entry in the X.25 configuration file. If you are not familiar with this tool, see the *SunLink X.25 Reference Guide* for more information. To invoke the tool, enter the following command as root:

```
/opt/SUNWconn/bin/x25tool &
```

When configuring your X.25 parameters, verify the configuration of your X.25 port on the DPN switch and change the parameters of the X.25 link on the MDM accordingly. For example, the Logical Channel Ranges, Window Size, and Packet Size should be the same on the X.25 port and DPN switch.

The table “Interface and X.25 parameters” (page 53) includes specific parameters that should be defined for the interface and X.25 parameters. Other parameters should be set according to the configuration of your X.25 port on the DPN switch.

Table 4
Interface and X.25 parameters

Screen	Parameters	Description/Suggested value
Interface Configuration	Local Address Full Address Interface	X.25 address of your X.25 port X.121 address of your X.25 port DTE
LAPB Parameters	Tx Window Size Max LAPB-I Frame	Suggested value = 7 Suggested value = 519 for packet size up to 512
X.25 Parameters	Network Profile	
	PLP Mode	DTE
	Sequence Numbering	Suggested value = 8
	Logical Channel ranges	The defined values for LCN ranges must match those defined in the X.25 port configuration.
	Link Modes	
	National DNIC	The DNIC of your DPN network.
	Facilities	
	CUG, with outgoing access	ON, if the NCS has CUG
	CUG, with incoming access	ON, if the Provisioning Agent on the DPN switch has CUG
	Incoming Reverse Charging	ON, this parameter MUST BE set as required by the Provisioning Agent.
	Timers & Counters	no changes required
	Throughput	
	Throughput Class (default)	Suggested value = 10
	Packet Size (default)	Suggested value = 512
	Window Sizes (default)	Suggested value =2
	Special	
	Call Accept In (D-Bit Control)	Clear Call
	Call Accept Out (D-Bit Control)	Clear Call
	Data In (D-Bit Control)	Reset Call
	Data Out (D-Bit Control)	Reset Call

Concurrent access to a packet module

Configuration for DPN Devices allows concurrent access to a packet module.

Allowing multiple provisioning sessions

Multiple provisioning sessions may be started for a packet module. In order to have multiple sessions, you must have the following set-up:

- The packet module must be running an office image on a PE386 at G32 or higher.
- The provisioning agent icon (PAGENT) must have the maximum number of LCNs set. For example, if three concurrent sessions are desired, then the maximum number of LCNs must be set to four.

  Note: If the module is running a PE286, then only one provisioning session is permitted.

Types of provisioning sessions

There are two types of provisioning sessions: read-only and update. Applications with read-only sessions may upload service data from the packet module but are not permitted to download to the module. Applications with an update session may upload and download. Only one update session per module is permitted and all other sessions associated with that module are read-only.

Configuration tools use either or both types of sessions as follows:

- Global Data Manager, Software Substitution, and Service Data Conversion always open update sessions to a module.
- MCF Directory Merge, Service Data Backup, Service Data Restore, NRS Populator, and Software Distribution always open read-only sessions to a module.
- Component Provisioning and Envelope Editor open an update session if their applications mode on the main window is set to Edit. If View is selected, a read-only session is opened.

How module locking occurs

Module locking is accomplished through the use of a semaphore lock. The Provisioning Agent (PA) provides the necessary services to allow applications to obtain and release the lock.

An update session locks the module and a read-only session does not lock the module. Downloads to the module can only be done by the application that has the module locked. When an application attempts to open an update session, a semaphore lock is requested. If the module is not already locked, the request is granted, otherwise the request fails and the application provides notification of the failure to the user.

When a lock succeeds, PA tracks the outstanding lock through the use of a special identifier. This identifier is considered the lock owner. The lock identifier provided by the provisioning is supplied by the SEA process, which has the following format:

```
SEA<pid><hostname>
```

where:

`<pid>` is the process ID of the SEA process.

`<hostname>` is the name of the Preside Multiservice Data Manager (MDM) workstation where the SEA process is run.

For example, if the SEA's process ID is *1234* and is running on the MDM workstation *myhost*, the lock owner would be *SEA1234MYHOST*.

Unlocking a module

Removing a module lock is accomplished by issuing the following series of commands on an NCS command console:

1. Disable the Provisioning Agent.

   `<PM_Mnemonic> PA DISABLE`

2. List the locks and look for the user who owns the lock.

   `<PM_Mnemonic> PA LIST LOCKS 0`

3. Remove the lock.

   `<PM_Mnemonic> PA UNLOCK 0 SEA_LOCK`

4. Enable the Provisioning Agent.

   `<PM_Mnemonic> PA ENABLE`

Locks that belong to an active tool should not be removed. Taking a lock away from an active tool results in the tool being unable to download service data.

The lock ID is uniquely defined to prevent potential interference between provisioning operators. This unique lock ID scheme also allows for ease of identifying which provisioning stack owns the lock.

Concurrent access to MCFs on the NMS disk

Configuration for DPN Devices allows for concurrent access to MCFs on the NMS disk.

Access to MCFs on the NMS disk

Configuration for DPN Devices allows provisioning to be performed on MCFs which are stored on the NMS disk. This enables operators to pre-configure a packet module without having to actually connect to the packet module. The directory where MCFs are stored can be defined in the configuration file /opt/MagellanNMS/cfg/PFA.cfg.

Concurrent access to MCFs on the NMS disk

Similar to accessing a packet module, multiple read-only and update provisioning sessions are allowed on the NMS disk. However, unlike accessing a packet module where the module is locked in an update session, multiple update provisioning sessions are allowed when accessing the NMS disk.

A locking mechanism is provided to allow the control of NMS disk concurrency access. The locking mechanism is similar to the semaphore lock service provided by the Provisioning Agent (PA) on the packet module, in which all the locks are kept in a file on the PM disk.

For NMS disk access, the file fpa.lock is created and maintained by the PFAS server. This lock file is located in the MCF directory on the NMS disk, and each entry in the file contains the information of the lock in the following format:

```
<type> <code> <name> <id>
```

where:

`<type>` is the application-defined integer value (0-255).

`<code>` is the application-defined integer value (0-255).

`<name>` is the application-defined string containing the semaphore information.

`<id>` is the lock identifier supplied by the application.

To enable NMS disk concurrency control, the semaphore name provided by the SEA process is the name of the MCF bundle to be updated or downloaded. The lock identifier is also supplied by the SEA process, which has the following format:

```
SEA<pid><hostname>
```

where:

`<pid>` is the Process ID of the SEA process.

`<hostname>` is the host name of the MDM workstation where the SEA process is running.

Using the above locking scheme, an update session or download MCF activity to NMS disk would only lock a particular MCF bundle. When an application attempts to open an update session or download MCFs to the NMS disk, a semaphore lock for the MCF is requested. If the MCF is not already locked by an other application, the request is granted. If the MCF is already locked, the request fails and the application provides notification of the failure to the user.

The PFAS server tracks and cleans up outstanding locks when the provisioning session is completed.

Unlocking MCFs on the NMS disk

PFAS server tracks and cleans up outstanding locks. However, if an unusual event happens, for example, restarting PFAS server, which may cause the locks to remain in the lock file, the locks can be removed as follows:

1. Determine the directory where the MCFs are stored. The *PFAS.lock* lock file is placed in this directory.
2. Edit the lock file.
3. Locate and delete the line that contains the outstanding locks.
4. Save the lock file.

Note: All locks can be reset by removing the lock file. The PFAS server will automatically re-create the lock file. This should be done carefully if there are multiple PFAS servers sharing the same MCF directory or multiple update provisioning sessions are running.

User preferences

User preferences define the attributes of a provisioning session and are maintained over different provisioning sessions. User preferences are stored in profiles and are associated with Component Provisioning and Field Access. A profile may contain user preferences for more than one application, for example, both Component Provisioning and Field Access. All user preferences are optional, if a user preference is not set via a profile the system default is used. There are three types of profiles:

- administrator (/opt/MagellanNMS/cfg/ProvisioningAdmin.cfg)
- user administration (/opt/MagellanNMS/cfg/ ProvisioningUserAdmin.<userid>.cfg)
- user (/<home directory>/MagellanNMS/ProvisioningUser.cfg)

There is one administrator profile, /opt/MagellanNMS/cfg/ ProvisioningAdmin.cfg, for every workstation. It allows the administrator to initialize user preferences differently than the system does and/or restrict the domain of possible values for user preferences. This profile is optional and applies only to users that do not have a user administration profile.

Each user can also have a user administration profile, /opt/MagellanNMS/cfg/ ProvisioningUserAdmin.<userid>.cfg. It allows the administrator to customize initialization of user preferences. This profile is optional and if it does not exist, the administrator profile will set the restrictions.

Having the above two profiles allows the administrator to create a restrictive ProvisioningAdmin.cfg file and give each userid specific privileges.

Each user can also have a user profile file called ProvisioningUser.cfg in the MagellanNMS directory under their home directory. This file is used by both DPN and Passport. You create this file in the Component Provisioning tool for either DPN or Passport by selecting Options -> Save Preferences.

The user profile file lists a set of DPN Component Provisioning options and their assigned values. These options are identified by COMPONENT_PROVISIONING at the beginning of the line. Similarly, exactly the same set of options are listed for Passport and are identified by PASSPORT_COMPONENT_PROVISIONING at the beginning of the line.

DPN and Passport do not use all of the options that are assigned to them. Instead, DPN ignores the options that are invalid for it and Passport ignores the options that are invalid for it. For instance, the option DOWNLOAD_LOCATION is listed for both DPN and Passport, but it is only used by DPN because this option is invalid for Passport . You can find out the valid options for DPN and Passport by looking at the Preferences dialogs, which you access from the Options menu of the Component Provisioning tool. For more information on DPN preferences, see 241-6001-012 *Preside MDM Configuration Management for DPN User Guide*:

In addition, the table “Component Provisioning user preferences” (page 61) and “Field Access user preferences” (page 63) provide a list of the system defaults for Component Provisioning and Field Access.

When Component Provisioning and Field Access are started, the administrator or user administration profile is loaded first then the user profile is loaded.

In the administrator and user administration profiles, the domain can only be restricted. That is, an open domain preference can become closed or a closed domain preference can be restricted further. Initially, most preferences have a closed domain or finite set of possible values. For example, DOWNLOAD_MODE can only be set to KEYED, DATED or USER_SPECIFIED. However, some have an open domain or infinite set of possible values. For example, DOWNLOAD_KEY has many possible valid settings.

Creating profiles

The user profile can be automatically created using the Component Provisioning tool by selecting Options -> Save Preferences. The administrator and user administration profiles cannot be created automatically. However, a simple way to create these profiles is to copy and modify a user profile as follows:

1 Log on as root.

2 To create an administrator profile copy the user profile, /<home directory>/MagellanNMS/ProvisioningUser.cfg file to /opt/MagellanNMS/cfg/ProvisioningAdmin.cfg.

3 To create a user administration profile copy the user profile, /<home directory>/MagellanNMS/ProvisioningUser.cfg file to /opt/MagellanNMS/cfg/ProvisioningUserAdmin.<userid>.cfg

4 Edit the new configuration files to include the appropriate parameters.

See the table “Component Provisioning user preferences” (page 61) for a list of applicable parameters.

Example

The following is an example of a DPN administrator profile. This file could also be used as a user administration profile, but you may want to change some of the field values for individual users. For example, some users may require an application_type of EDIT.

```
COMPONENT_PROVISIONING APPLICATION_TYPE LOOK
COMPONENT_PROVISIONING DOWNLOAD_ACTIVATABLE TRUE
COMPONENT_PROVISIONING DOWNLOAD_CONFIRM FALSE
COMPONENT_PROVISIONING DOWNLOAD_KEY MYKEY
COMPONENT_PROVISIONING DOWNLOAD_LOCATION
DEFAULT_DESTINATION
COMPONENT_PROVISIONING DOWNLOAD_MODE KEYED
COMPONENT_PROVISIONING DOWNLOAD_TYPE INCREMENTAL
COMPONENT_PROVISIONING SDT_CONFIRM_DELETION TRUE
COMPONENT_PROVISIONING SDT_TEMPLATE_DIRECTORY /
localdisk/user1/MagellanNMS/provisioningTemplates
COMPONENT_PROVISIONING UPLOAD_KEY MYKEY
COMPONENT_PROVISIONING UPLOAD_LOCATION DEFAULT_PM
COMPONENT_PROVISIONING UPLOAD_MODE KEYED
COMPONENT_PROVISIONING DOWNLOAD_DATE 920616
COMPONENT_PROVISIONING UPLOAD_DATE 920616
```

Table 5
Component Provisioning user preferences

Key	System default	Domain	Range
APPLICATION_TYPE	EDIT	Closed	LOOK, EDIT
UPLOAD_MODE	USER_SPECIFIED	Closed	USER_SPECIFIED, KEYED, COMMITTED, DATED, ACTIVE (DPN only), EDIT (Passport only), CURRENT (Passport only)
UPLOAD_DATE	000101	Open	valid date or *today*
UPLOAD_KEY	MDM	Open	
UPLOAD_LOCATION (DPN only)	DEFAULT_PM	Closed	DEFAULT_PM, USER_SPECIFIED, NMS_DISK
ENABLE_PRELOAD) (Passport only)	TRUE	Closed	TRUE, FALSE
PROPAGATION_LOGGING	FALSE	Closed	TRUE, FALSE
PROPAGATION_LOG_FILE		Open	
DOWNLOAD_MODE	KEYED	Closed	USER_SPECIFIED, KEYED, DATED
DOWNLOAD_DATE	000101	Open	valid date or *today*
DOWNLOAD_KEY	MDM	Open	
DOWNLOAD_LOCATION (DPN only)	DEFAULT_DESTINATION	Closed	DEFAULT_DESTINATION,USER_SPECIFIED, NMS_DISK
DOWNLOAD_TYPE (DPN only)	INCREMENTAL	Closed	INCREMENTAL, COMPLETE
DOWNLOAD_CHECK (DPN only)	INCREMENTAL_MCF	Closed	INCREMENTAL_MCF, COMPLETE_MCF, MINIMAL_MCF
DOWNLOAD_ACTIVATABLE (DPN only)	TRUE	Closed	TRUE, FALSE

(Sheet 1 of 2)

Table 5 (Continued)
Component Provisioning user preferences

Key	System default	Domain	Range
SAVE_ASCII_OPTION (Passport only)	FALSE	Closed	TRUE, FALSE
SAVE_FULL_OPTION (Passport only)	FALSE	Closed	TRUE, FALSE
CHECK_FULL (Passport only)	TRUE	Closed	TRUE, FALSE
CHECK_STOP_ON_ERR (Passport only)	TRUE	Closed	TRUE, FALSE
DOWNLOAD_CONFIRM	FALSE	Closed	TRUE, FALSE
CHANGE_ACTIVATION_DATE (DPN only)	FALSE	Closed	TRUE, FALSE
ACTIVATION DATE (DPN only)	000101	Open	valid date or *today*
SDT_TEMPLATE_DIRECTORY	<home directory>/ MagellanNMS/ provisioningTemplates	Open	
SDT_CONFIRM_DELETIONS	TRUE	Closed	TRUE, FALSE
ENABLE_USER_ CUSTOM_FORMS	TRUE	Closed	TRUE, FALSE
ENABLE_WORKSTATION_ CUSTOM_FORMS	TRUE	Closed	TRUE, FALSE
SAVE_OPTION	This option is no longer used.		
(Sheet 2 of 2)			

Note 1: All fields loaded from the profiles are treated as uppercase except for the value for SDT_TEMPLATE_DIRECTORY.

Note 2: See “Dated algorithm” (page 129) for information on the date format.

Note 3: Any records not starting with an application name (for example, COMPONENT_PROVISIONING) are ignored.

Table 6
Field Access user preferences

Key	System default	Domain	Range
NCD_SERVER	empty string	Open	

Note 1: All fields loaded from the profile are case-sensitive.

Note 2: Any records not starting with an application name (for example, FIELD_ACCESS) are ignored.

Administrator and user profile syntax

The following syntax is supported in a user or administrator profile:

- The following line initializes the preference. When specified in the administrator profile, this can be successfully overridden by either the user profile or the user. Open domain preferences are treated the same as closed domain preferences. This syntax is allowed in either the user or administrator profile.

  ```
  <application_name> <preference> <setting>
  ```

  Example
  COMPONENT_PROVISIONING DOWNLOAD_LOCATION NMS_DISK

- Open domain preferences cause a data selector to be created. This syntax is only allowed in an administrator profile.

  ```
  <application_name> <preference> <setting> (SET
  ```

  Example
  COMPONENT_PROVISIONING DOWNLOAD_LOCATION NMS_DISK (SET

  COMPONENT_PROVISIONING UPLOAD_KEY 92WK01 92WK02 92WK03 (SET

  SET in the above example causes a data selector containing 92WK01, 92WK02 and 92WK03 to be created on the Upload Key data entry field. The user can still enter any valid key.

- Closed domain preferences cause other choices to be disabled. FORCE closes the domain. The preference may only be set to one of the values in the set. This syntax is only allowed in administrator profiles.

  ```
  <application_name> <preference> <setting> (FORCE
  ```

  Example
  COMPONENT_PROVISIONING UPLOAD_MODE COMMITTED ACTIVE (FORCE

  Open domain preferences result in a data selector to be created. In the following example force causes a data selector containing 92WK01, 92WK02 and 92WK03 to be created on the Upload Key data entry field. The user must enter one of these keys.

  COMPONENT_PROVISIONING UPLOAD_KEY 92WK01 92WK02 92WK03 (FORCE

  Closed domain preferences cause other choices to be disabled. In the following example FORCE causes the buttons for the other choices for DOWNLOAD_LOCATION (USER_SPECIFIED and DEFAULT_DESTINATION) to be disabled.

  COMPONENT_PROVISIONING DOWNLOAD_LOCATION NMS_DISK (FORCE

Profile loading errors

All three user profiles are loaded during initialization. When using Field Access, if any errors are encountered while processing the profiles. An error message is displayed and the system defaults will be used. If an invalid SET or FORCE is encountered during a Component Provisioning session, it causes an error dialog to be displayed and subsequent abort. The user cannot continue. Therefore, restart Component Provisioning after any changes have been made to the administrator profile to prevent invalid changes from affecting users.

It is an error if the user profile attempts to set a preference to an invalid value, that is, a value outside its domain. When initializing Component Provisioning, an error dialog is displayed and the user cannot perform any operation which requires that preference until the error has been cleared.

For example, if the administrator profile contains:

```
COMPONENT_PROVISIONING DOWNLOAD_LOCATION NMS_DISK (force
```

and the user profile contains:

```
COMPONENT_PROVISIONING DOWNLOAD_LOCATION
DEFAULT_DESTINATION
```

The user is presented with the error dialog and cannot download until *OK* is clicked on the User Preferences - Download dialog. The dialog will contain NMS_DISK as the Download Location with any other choices greyed out.

Users and groups

Preside Multiservice Data Manager (MDM) allows the definition of users and groups of users. A separate group of users is recommended for provisioning. Define the Preside MDM provisioning toolset accessible to this group. If a group of users needs access to provisioning and other tool sets, then a separate group for these users is recommended. Multiple tool sets can be made accessible to this group through the X Window Work Space file.

It is recommended that all provisioning users have access to DPN Fault tools in addition to the Configuration tools. Each provisioning user must be defined separately. The user ID along with the NCS operator ID is used for concurrency control. Both user IDs are provided in the provisioning audit logs (logs are generated by the Provisioning Agent). See 241-6001-303 *Preside MDM Administrator Guide* for the procedure on defining users and user groups and how to customize their toolsets.

Component Provisioning customizable resources

The Component Provisioning tool uses resources to describe some of its functional and appearance aspects. You can customize the resource file /opt/MagellanNMS/lib/app-defaults/C/PUI to change some of these aspects for all users of the tool. You can customize the file .Xdefaults to change some of these aspects for a single user of the tool.

Selecting options for a single user of Component Provisioning

You can select display options for a single user of Component Provisioning by modifying the resource values in the .Xdefaults file of a user’s home UNIX account. The changes made will only apply to the user account, and not to every workstation.

Modifying a user's .Xdefaults file

1 Using a UNIX editor, create or edit the .Xdefaults file in the user's home account.

For example, user oper1 edits the /home/oper1/.Xdefaults file.

2 For every resource specification you want to customize, add a line to the file. Ensure that this line is prefixed with *PUI*. For example, to set the network model option, add the following line to the .Xdefaults file:

```
PUI.*noNetworkModel: True
```

See the table "Resources that can be customized in Component Provisioning" (page 67) for a list of resources that can be customized for Component Provisioning.

3 After saving the changes, make the changes effective by issuing the following command from the UNIX access window:

```
xrdb /home/oper1/.Xdefaults
```

Selecting options for all users of Component Provisioning

You can select display options for all users of the Component Provisioning tool by changing the values of the resource variables in the PUI resource file. The changes made will apply to every user on the workstation.

Modifying the Component Provisioning resource file

1 Log on as root.

2 Make a copy of the file, enter:

```
cp /opt/MagellanNMS/lib/X11/app-defaults/C/PUI /opt/
MagellanNMSlib/X11/app-defaults/C/PUI.ori
```

3 Using a UNIX editor, modify the resource variables.

See the table "Resources that can be customized in Component Provisioning" (page 67) for a list of resources that can be customized for Component Provisioning.

4 Restart the Component Provisioning tool and verify that the changes made have been successful.

Table 7
Resources that can be customized in Component Provisioning

Resource	Description	Legal values
noNetworkModel	Ignores the network model if set to True. Default value is False.	True/False
subCompDblClickAction	Double click accelerator for the Subcomponents area. Default value is expand.	expand, contract, edit, report, view, fully expand, put context
subCompShiftDblClickAction	Shift double click accelerator the Subcomponents area. Default value is contract.	expand, contract, edit, report, view, fully expand, put context
subCompCtrlDblClickAction	Control double click accelerator for the Subcomponent area. Default value is edit.	expand, contract, edit, report, view, fully expand, put context
subCompMetaDblClickAction	Meta double click accelerator for the Subcomponent area. Default value is report.	expand, contract, edit, report, view, fully expand, put context
PUI*NetWideDataPixmap	Specifies the pixmap used in Component Provisioning to identify network-wide data.	Any valid file name
PUI*DPNNetWideDataFile	Name of file that lists the components containing network-wide data. Default name is DPN_Net_Wide_Data.dat.	Any valid file name
PUI*PassportNetWideDataFile	Name of file that lists the components containing network-wide data. Default name is Passport_Net_Wide_Data.dat.	Any valid file name

Customizing Configuration for DPN Devices for command filtering

You can specify command filtering options in the PUIDpnCmd.cfg file for the DPN Provisioning UI or the PUIPPCmd.cfg file for the Passport Provisioning UI. These files are in the /opt/MagellanNMS/cfg directory or in the toolset files that contain Component Provisioning UI invocations (/opt/MagellanNMS/bin/pui). One or more of the following options invokes command filtering:

```
-filter_appl '<UNIX command>'
```

This option specifies the UNIX Bourne shell command that is to be a customer application for command filtering. The Provisioning Command Filter API Provider sets up stdin and stdout for the customer application.

```
-filter_serv <node/IP address> <port>
```

This option specifies the node and port of the server that is to be the customer server for command filtering. The Provisioning Command Filter API Provider makes a stream socket call to the customer server.

```
-filter_trace
```

This option turns on Provisioning Command Filter API tracing. Trace output goes to stderr. This option is not recommended for Component Provisioning invocations in toolset files. The amount of trace output may be large and stderr may not be directed to a suitable location. The -filter_trace option is best used in a manual invocation of Component Provisioning from a UNIX window. If the -filter_trace option is used in a toolset file invocation, ensure that stderr is directed to a suitable location.

```
-filter_width
```

This option is used to specify the maximum length of lines output by the Provisioning Command Filter API to customer applications and servers. Values from 72 to 100000 are valid. The default value is 72.

See 241-6001-209 *Preside MDM Provisioning Command Filter API for DPN* for more information on the Provisioning Command Filter API.

Activating changes to toolset files

To activate any changes that are made to toolset file definitions, log out of the workstation and log back in again.

Chapter 4
Service data backup

This chapter gives an overview of how service data is backed up using the Configuration for DPN Devices application. In this chapter, you can find the following information:

- “About service data backup” (page 69)
- “Manual backup” (page 69)
- “Automatic backup” (page 70)

About service data backup

DPN-100 modules support shadowed disks. When used, the service data is automatically duplicated on the shadowed disk. Service data can be backed up on either an Preside Multiservice Data Manager (MDM) or other backup disk. Administratively, backups can be managed manually or automatically.

Manual backup

Manual backup mode, allows a provisioning user to back up selected MCFs using the Service Data Backup tool. All files associated with the selected MCF are backed up. The same application allows a user to back up to a selected disk. See 241-6001-012 *Preside MDM Configuration Management for DPN User Guide* for more information.

> ***Note:*** The Service Data Backup tool does not back up the DPN software images.

You are required to determine which MCFs need to be backed up. The following facilities will help with this task.

- The on-switch *File directory* command displays the date and time when the MCF was created. MCFs created after a selected date are easily identified.
- The Service Data Backup application allows users to view (List MCFs) the MCFs on the switch.
- The Service Data Restore application allows users to view the backed up service data on the backup site. This provides an easy way to identify the backups already made.

Automatic backup

Automatic backup mode allows all new files generated by the Configuration for DPN Devices application to be automatically backed up. This mode is selected by setting the AUTO_BACKUP parameter to ON in the file /opt/MagellanNMS/cfg/PFA.cfg.

Automatic backup can back up MCF files to either the NMS Backup disk or the PM module disk. To back up MCFs to the NMS Backup disk, set the MODULE_BACKUP_TO_DISK parameter to ON in the file /opt/MagellanNMS/cfg/PFA.cfg. Otherwise, the backup MCFs are created on the PM disk.

Chapter 5
Software images

This chapter describes how to download and migrate software images. In this chapter, you can find the following information:

- “Software distribution procedure” (page 71)
- “Software migration” (page 75)

Software distribution procedure

The DPN software distribution system involves the following steps. See the figure “Download images” (page 73).

1 Ensure that the software download server is properly configured on the workstation. See “Installation and Configuration” (page 41) for more information.

2 Receive the DPN software release.

For steps on how to receive DPN software, follow the steps in “Receiving DPN software images” (page 73). The software must be written to the directory /opt/MagellanNMS/cfg/dpn_img. This directory is designated as a software distribution site (SDS).

3 Download images from the NMS disk to the RDS.

The images must reside on the SDS and are downloaded to the RDS using the software download server. The software download server is a separate PFA server for the exclusive use of the Software Distribution tool. It must have access to the directory /opt/MagellanNMS./cfg/

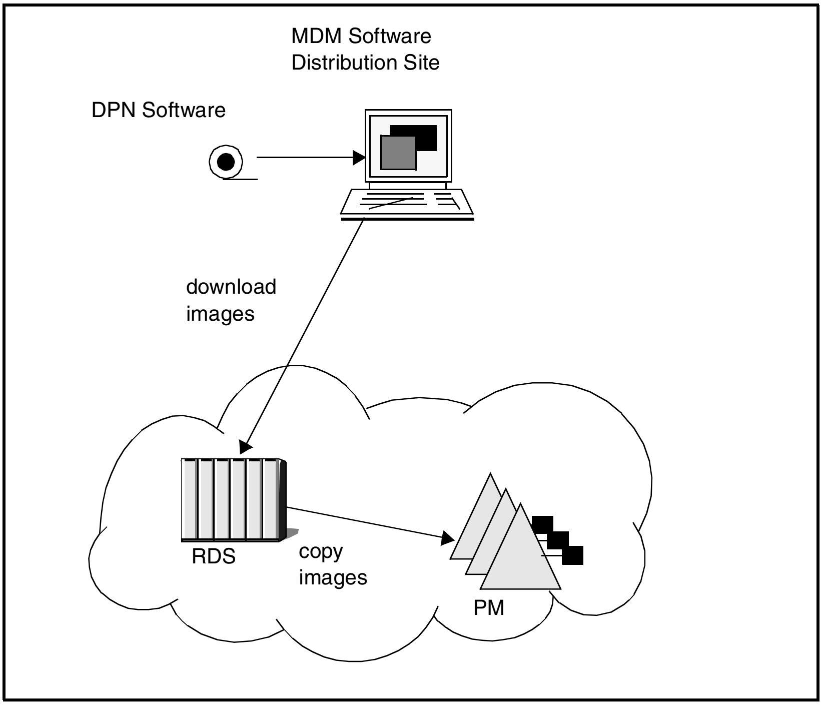
dpn_img in order to download the DPN software images. See 241-6001-012 *Preside MDM Configuration Management for DPN User Guide* for more information on the Software Distribution tool.

4 Copy images from RDS to the DPN-100 modules.

In order to instruct a DPN-100 module to copy images from an RDS, the Software Distribution application sends LEVEL-4 copy RDS commands to the target DPN-100. Based on the configuration specified in the active service data, the DPN-100 sets up a call to the designated RDS and sends requests to get the files. A direct call to the RDS must be provisioned in the PAGENT icon of the target PM. See "ICONS" (page 131) for more information on the icon service data.

The image file size and link speed will also impact the time required to download the images. It will take approximately 14 minutes to download a typical 2M image from the software distribution site to the RDS using a 19.2 Kbit/s link.

Figure 5
Download images



DPN software images

The software download server must have access to the directory /opt/ MagellanNMS/cfg/dpn_img on the Preside Multiservice Data Manager (MDM) workstation to enable the images to be downloaded from the SDS to the RDS. A tape drive or CD-ROM drive with the most recent copy of the images must be connected to the MDM workstation.

Receiving DPN software images

1 Log on to the Preside Multiservice Data Manager (MDM) workstation as root.

2 Insert the CD-ROM. Solaris automatically mounts the CD.

3 To download images from the CD-ROM drive, enter:

```
cd /MagellanNMS/cfg/dpn_img
```

```
cp /cdrom/cdrom0/* .
```

The copy command may take several minutes to complete. Once the images have been copied, the UNIX prompt will appear.

4 When the installation is complete, at the UNIX prompt, type:

```
eject /cdrom
```

5 Log off the root account.

Deleting software images

You can delete images specified in the file *images.<generic>* and macros specified in *npm.macros.rm.<generic>* and *npm.macros.nm.<generic>* with the command *tidyimg*.

> ***Note:*** The images or macros will not be deleted if they are specified in another release.

1 Log on to the Preside Multiservice Data Manager (MDM) workstation as root.

2 From the Preside MDM window, select System -> Utilities -> Unix Access.

A UNIX window appears on the screen.

3 List the release names by typing

```
cd /opt/MagellanNMS/cfg/dpn_img
ls images.*
```

The list of images, for example, images.g3101 and images.g3209, are listed. The filename extension is the release name of the files to be deleted.

4 Delete the images by typing:

```
/opt/MagellanNMS/bin/tidyimg <release>
```

where *release* is the name of the release you want to tidy. Once the tidy is complete, Tidy complete is displayed on the screen and the UNIX prompt is displayed.

Download missing images

Before activating a new MCF, use the on-switch *check mcf* command to verify that all the required images exist on the PM disk. If there are any missing images, do the following:

- Copy the required images from the RDS using the *getimg* command with the MCF (*-tm*) option. The following example copies the required images for an MCF from the RDS:

```
/opt/MagellanNMS/bin/getimg -ncs <dest_mnemonic>
<capability> <password> -tm R34 MC.2588.4034.0
```

or

- Download the required images from the SDS to the target PM directly by using the *getimg* command with the source *sds* option. The following example downloads the required images for an MCF from the software distribution site.

```
/opt/MagellanNMS/bin/getimg -ncs <dest_mnemonic>
<capability> <password> -src sds -tm R34
MC.2588.4034.0
```

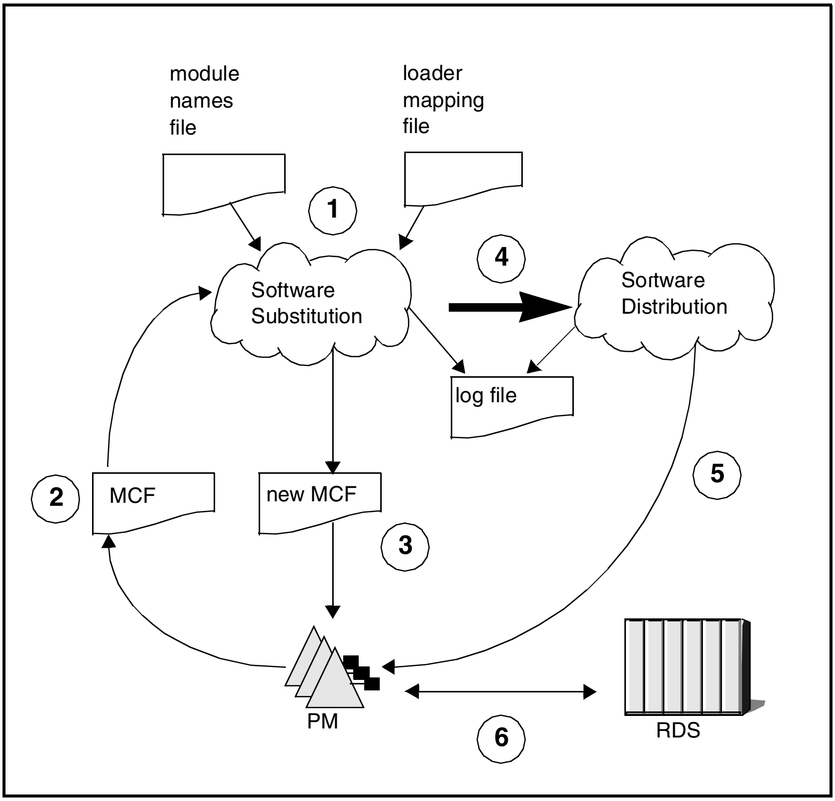
See 241-1001-303 *DPN-100 Operator Commands and Responses* for more information on the *check mcf* command.

Software migration

The Software Substitution tool has been designed to support image upgrading without forcing the use of any one method of migration. In general, one of two methods is likely to be used: migration in a single functional step - MCF modification and distribution of new images together; or migration in two functional steps - distribution of new images to modules followed later by MCF modification.

With either of the above methods, the initial image download is common to both. See “Software distribution procedure” (page 71) for more information on downloading.

Figure 6
Upgrading images



The figure “Upgrading images” (page 76) illustrates the one-step approach and shows the tasks involved in upgrading software images.

1 Software Substitution loads the module names file to determine the modules and MCFs to work with. It also loads the loader mapping file containing the mapping of old images to new images.

2 Each of the MCFs listed in module names file is uploaded from a target module, the image names in the PE_Loader envelopes are updated.

3 The updated MCF is downloaded to the target module.

4 Once the MCFs have been updated, Software Substitution invokes the Software Distribution tool to distribute the new images to the target module(s).

5 Software Distribution has each target module issue the commands it needs to retrieve the necessary images from the RDS.

6 The necessary images are copied from the RDS to the modules.

If the *-nodist* option is specified in the one-step approach, steps 4, 5 and 6 are not performed.

The first step of the two-step approach distributes the images to the target modules and includes tasks 1, 2, 4, 5, and 6 listed above. The second step in the two-step approach updates an MCF to use the new images and includes tasks 1, 2, and 3, and optionally 4, 5, and 6.

Note that it is also possible to first download the changed MCFs to the NMS disk, and then at a later time download those MCFs to the target modules and distribute the images.

Software migration procedure

The one-step approach to migration is useful for small networks or possibly for a portion of a larger network:

1 Run the Software Substitution tool as described in 241-6001-012 *Preside MDM Configuration Management for DPN User Guide*, on the required modules. Do not specify the *-nosub* and *-nodist* options. The tool will download a new MCF containing the new image names to use, and will retrieve the necessary images from the RDS and put them on the target module disk.

2 Check the generated log file to ensure the substitution and distribution of images was successful.

3 As a precaution, run *check mcf* on the new MCF to ensure all images are present, If any are missing, download the missing images. See “Download missing images” (page 75) for more information. Also, see 241-1001-303 *DPN-100 Operator Commands and Responses* for more information on the *check mcf* command.

4 Activate the updated MCF.

5 Old images no longer required by the module may now be removed from the module disk.

The two-step approach to migration is useful for large networks.

1. Run Software Substitution on the required modules, specifying the *-nosub* option on the command line as described in 241-6001-012 *Preside MDM Configuration Management for DPN User Guide*. This will result in the Software Substitution tool looking through the listed MCFs, determining which images will be required, and then distributing the images to the modules. No new MCFs are downloaded at this time.
2. Check the generated log file to ensure the distribution of images was successful.
3. At some later time, the Software Substitution tool is run again without the *-nosub* option. This time, updated MCFs are downloaded. Software Substitution will once more retrieve any needed images from the RDS. If this second retrieval is not necessary, specify the *-nodist* option on the command line. The second retrieval of images is a precaution in case a new image not on the module disk is named in the MCF uploaded by the second run of Software Substitution. This situation could happen if the MCF uploaded by the first run of Software Substitution (as described in step 1) is different than the MCF uploaded by the second run of Software Substitution.
4. Check the generated log file to ensure the substitution and distribution of images was successful.
5. As a precaution, run *check mcf* on the new MCF to ensure all images are present, If any are missing, download the missing images. See “Download missing images” (page 75) for more information. Also, see 241-1001-303 *DPN-100 Operator Commands and Responses* for more information on the *check mcf* command.
6. Activate the updated MCF.
7. Old images no longer needed by the module may now be removed from the module disk.

Chapter 6
Time Stream Management using keyed MCFs

This chapter describes Time Stream Management (TSM) as it relates to Configuration for DPN Devices. In this chapter, you can find the following information:

About Time Stream Management

Configuration for DPN Devices provisions service data on a module basis. The service data resides in a set of configuration files which describe a single module. The configuration files are downloaded to the intended module for activation. Configuration files are identified by the name of their master configuration file (MCF).

Time Stream Management (TSM) is a concept used by various customers to control service data changes and activations based on dates. The majority of DPN users’ time stream requirements can be categorized into two modes of operation as follows:

- *Provisioning for Immediate Use*
  Provisioning changes are made and activated immediately. This is required for emergency updates. For smaller networks, this could be the regular mode of operation.

- *Provisioning for Future Use*
  In this mode, a large number of provisioning updates are made ahead of time, tested, and are ready to be activated some time in the future.

This chapter provides information on how Configuration for DPN Devices provides the capability to implement Time Stream Management for service data changes.

Configuration for DPN Devices allows the user to assign a date to an individual MCF. This is referred to as a *datekey* and is the basis of TSM. This datekey refers to the module the MCF describes.

Configuration for DPN Devices also allows the user to assign a mnemonic to an individual MCF. This is referred to as a *key* and can be used to design and operate a customized TSM system.

> ***Note:*** For optimal TSM, it is recommended that users use the datekey rather than the mnemonic key.

MCF mnemonic key

TSM supports what is called the *keyed* approach to MCF management.

Through the use of User Preferences, which allows you to control the mode of MCF selection, a mnemonic key naming convention can be implemented. This allows you to assign a mnemonic key which has some significance in the TSM naming convention chosen.

The mnemonic key is used to search for the MCF which matches the pattern: MC.<key>nn.<namsid>.0, where key is a user defined mnemonic key and nn is a system assigned index number between 00..99. Index numbers are automatically assigned sequentially by incrementing the last index number by one. Once the index reaches 99 the MCF can no longer be downloaded. Service data changes to an MCF are automatically accumulated in the MCF with the highest index value. The key can be any user specified string, up to six characters, consisting of letters, numbers and underscores. The NAMS ID used is that of the currently selected module from the Network Model or can be selected manually.

MCF datekey

TSM also supports what is called the *dated* approach to MCF management. This is the recommended mode of operation in order to optimize the implementation of TSM.

A dated MCF is a keyed MCF where the key, known as the *datekey*, is a valid date in the format *yymmdd*. See “Dated algorithm” (page 129) for information on the date format.

When creating an MCF in dated mode, Configuration for DPN Devices essentially operates as if it is in keyed mode. However, when uploading an MCF in dated mode, Configuration for DPN Devices may operate differently. When uploading using a datekey, for example, 920616, Configuration for DPN Devices looks for an MCF matching the pattern: MC.920616nn.<namsid>.0. If any exist, the MCF with the highest index is returned. If one does not exist, Configuration for DPN Devices uploads the dated MCF having the latest date earlier than the given date.

For example, if MC.92050100.<namsid>.0, MC.92060100.<namsid>.0 and MC.92070100.<namsid>.0 exist on the module and a dated upload is requested for 920616, MC.92060100.<namsid>.0 would be uploaded.

Activation date

When an MCF is created in dated mode, the activation date of that MCF is set to the value specified by the date key. For example, if the date is 920616, the activation date of the MCF will be set to 1992061600000000. See “Dated algorithm” (page 129) for information on the date format.

When an MCF is created in any other mode, the default activation date is propagated from the uploaded MCF. The activation date may be changed by selecting *Change activation date* from the *Change Download Preferences* dialog in the Component Provisioning tool.

As stated earlier, Configuration for DPN Devices provisions on a module basis. Modules are the only service data object which can have timing information stored explicitly. Since modules are the root of the Configuration for DPN Devices service data hierarchy, all other components under the module share that timing information. Components that are associated with other components, for example PVC, must have similar timing information.

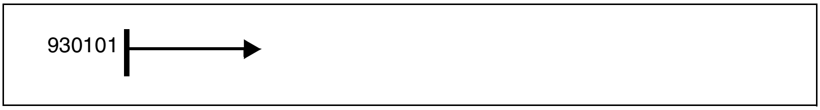
Therefore, if using TSM, when a PVC is provisioned both ends of the PVC (most likely on different modules) must be provisioned in the same time stream.

Using dated MCFs

The idea behind a dated MCF naming convention, is to provide a naming convention in which the MCF name conveys timing information. The MCF name will be used to determine the time interval for which the service data in the MCF is valid.

This section describes the provisioning of current and future views, and the creation of new time streams. Assume all modules have been initially provisioned using dated mode with the datekey 930101. Service data management is based on weekly time streams starting with week 01 in 1993. Datekey 930101 maps to week 01 in 1993. In the figure “Dated MCFs with one stream” (page 82) and the figure “Dated MCFs with two streams” (page 83), we will use module names OTTAWA1 and OTTAWA2.

Figure 7
Dated MCFs with one stream

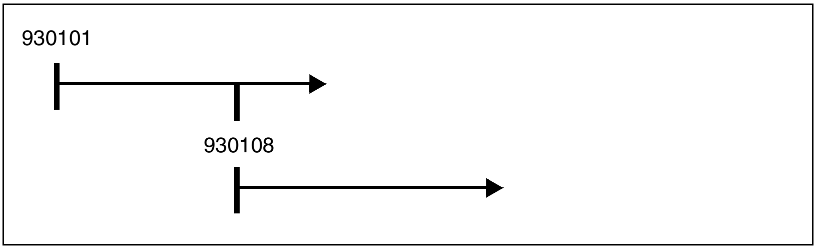


The figure “Dated MCFs with one stream” (page 82) implies that all the provisioned modules are scheduled for activation on 930101. There is currently only one stream, representing week 01 1993.

On January 2, 1993 an update is required for a emergency fix on module OTTAWA1. This change is required immediately. A dated upload is performed using datekey 930101. After the changes are made the MCF key is downloaded with the datekey 930101. The new MCF is then ready to be activated.

On January 4, 1993 an update is required for a new line on module OTTAWA1 to be activated in week 02. Assuming there is no week 02 stream yet, a dated upload is performed using datekey 930101. After the changes are made the MCF key is downloaded with the datekey 930108. A new stream has now been created.

Figure 8
Dated MCFs with two streams



Changes can now be made in the week 02 (930108) stream without affecting any service data in the week 01 (930101) stream and vice-versa. Any changes required in both streams must be made in both streams.

Now assume it is week 02 and a service data change is required to add a direct call between a DNA on OTTAWA1 and a DNA on OTTAWA2 to be activated in week 03. Also assume, that OTTAWA1 only contains dated MCFs from the week 02 (930108) stream and OTTAWA2 only contains dated MCFs from the week 01 (930101) stream.

By using the dated approach, the operator is alleviated from tracking which streams exist on each module. The operator does not need to know what stream is the latest on either module. As long as the migration is from the latest (relative to 930101) on each module, Configuration for DPN Devices takes care of the rest. By using a datekey of 930115 (week 03) to upload and download from/to both OTTAWA1 and OTTAWA2, the correct MCF is picked up and the week 03 stream is created, regardless of whether the modules are at week 03, week 02, or any previous week.

The dated MCF approach is supported by all applicable applications in the DPN Devices configuration toolset.

Chapter 7
Service data conversion

This chapter describes how to convert service data for a new main Preside Multiservice Data Manager (MDM) software release. In this chapter, you can find the following information:

- “About service data conversion” (page 85)
- “Deployment procedures” (page 86)
- “Template conversion” (page 88)
- “Service data conversion requirements” (page 100)

About service data conversion

The evolution of the DPN-100 product often includes enhancements to existing network services. Existing service enhancements require service data changes to allow the services to function properly. These changes raise the requirement for Configuration for DPN Devices to convert older levels of service data to include the new service data changes.

A service data conversion tool is provided to allow automation of service data conversion with each new main Preside Multiservice Data Manager (MDM) software release.

The Service Data Conversion tool converts service data from one level to the current level, one MCF at a time or in batches using a command file. See 241-6001-012 *Preside MDM Configuration Management for DPN User Guide* for more information on command files.

An MCF must be converted to the latest service data level before the following Configuration applications can use the service data:

- Global Data Manager
- Network Reporting System
- Component Provisioning
- Software Substitution

Each of the above applications will detect an MCF that has not been converted and issue a message that service data conversion is required.

Service data conversion is required when:

- a new release of MDM is deployed which introduces a new service data version
- a module is entirely provisioned or has components provisioned by Configuration for DPN Devices

Service data conversion supports forward conversion only. Service data conversion supports conversion from service data level *n* directly to service data level *n*+3.

Service data converted with the conversion tool is compatible with all supported DPN-100 generic software releases without service impact.

Service data conversion supports all methods of service data upload (ACTIVE, COMMITTED, KEYED, DATED, USER_SPECIFIED) and download (KEYED, DATED, USER_SPECIFIED). In the case of incremental downloads, only the components changed by the conversion are downloaded. Incremental downloads are allowed, however *complete downloads* are highly recommended to synchronize the MCFs with the new release. For more information on MCF upload and download modes, see “Uploading and downloading MCFs” (page 123).

Deployment procedures

The following sections detail the recommended deployment procedures for service data conversion.

MCF service data conversion

Service data conversion is required for modules and MCFs. It is recommended that this be done following the installation of a new Preside Multiservice Data Manager (MDM) release which introduces a new service data version for all modules provisioned by Configuration for DPN Devices.

1 Upgrade the MDM software release level.

2 See the release supplement that is applicable to the level running in your network to verify if the new MDM release is at a different service data version level than the MDM release from which you migrated. If there is a difference in the service data version level, convert the active MCF using the Service Data Conversion tool.

3 Activate the converted MCF.

DPN-100/1 service data conversion

DPN-100/1 service data must be converted to the new Preside Multiservice Data Manager (MDM) release level prior to replacing the cartridge. This is to allow for the envelopes which have changed to be upgraded before the cartridge is installed. It is highly recommended that this be done following the installation of a new MDM software release for all DPN-100/1 modules.

1 Upgrade the MDM software release level.

2 Convert the active MCF using the Service Data Conversion tool.

3 Activate the converted MCF.

Global Data Manager

The Global Data Manager prevents synchronization between two MCFs that are not at the latest SD_version. Impacts will be minimal if no Global Data changes are planned during the conversion period. If such changes are required, then two master MCFs, one for each SD_version, will be required. The Global Data Manager would have to be run from two different Preside Multiservice Data Manager (MDM) releases to target each MCF with the appropriate SD_version level.

Template conversion

A template contains the internal names, structure and field values of a specific instance of service data captured at a particular service data level. Thus, a template can only be used to create service data for the service data level in which the template was created. Therefore, when a new software release is installed, the templates must be recreated.

Regenerating templates

1 If the new software release requires explicit service data conversion, use the *egrep* command to create a list of MCFs associated with templates. Select a Unix window, go into the template directory and type the following:

```
egrep  '^MCF=' *
```

This produces a list of template file names and the MCF used to generate each template file. For example:

```
PE_5.sdt08:MCF=MC.95111100.4034.0
PI_5.sdt08:MCF=MC.95111100.4034.0
PO_1.sdt08:MCF=MC.95111100.4034.0
```

2 Convert the MCFs identified, using the service data conversion utility, to the latest SD_version.

If you will be substituting MCFs by MCF name in the next step, keep track of the association between converted and old MCFs.

If the MCFs are keyed and the converted MCFs are downloaded using the same keys, this is not necessary.

3 Regenerate all templates using the *regentemplates* tool.
If an explicit service data conversion was required, or you are consolidating the template MCFs, use the -*smcf* option (or the -*skey* or -*sdatekey* options as desired) to specify the required MCF substitutions.

4 To delete any MCFs that are no longer referenced, use the *deletemcf* command.

Template consolidation

This is a good opportunity to consolidate the MCFs used for managing templates. As MCFs are updated with new service data for new templates, the older templates continue to refer to the older MCFs even though their service data remains unchanged in the new MCFs. To reduce the number of MCFs regenerate the older templates using newer MCFs and then use the *deletemcf* command to delete the MCFs that are no longer referenced.

See 241-6001-012 *Preside MDM Configuration Management for DPN User Guide* for more information on creating and managing templates

The Template regeneration tool (*regentemplates*)

The regentemplates tool uses the <MCF name> and <component id> information in a template to automatically regenerate templates. By default, the MCF specified in a template file is uploaded in order to regenerate that template but there are options to specify a keyed or date keyed upload so that a more up-to-date version of the service data is used. In addition, a set of substitution options allows substituting a new MCF name, key or date key for the existing value. Again, this is to allow the template to be regenerated from the most recent service data.

The template regeneration tool is invoked from a Unix window using the following syntax:

```
/opt/MagellanNMS/bin/regentemplates [ -h | -help ]
```

or

```
   [ -server <server> ] [ -trace ]
   [ -log <logfile> ] [ -quiet ]
   [ -mcf ] [ -key ] [ -datekey ]
   [ -auto ]
   [ -smcf <oldmcf> <newmcf> ]
   [ -skey <oldkey> <newkey> ]
   [ -sdatekey <oldkey> <newkey> ]
   [ -dateboundary <date> ]
   <template file name> ...
```

where:

`-h | -help` displays usage summary.

`-server <server>` overrides the default server /opt/MagellanNMS/ bin/ fa. `<server>` is the name of an executable file that is to be invoked in place of /opt/MagellanNMS/ bin/fa (the default) for debugging purposes.

`-trace` writes debugging information (PDUs as they are sent and received, tokens as they are parsed, and a few other things) to stderr.

`-log <logfile>` duplicates terminal output to <logfile>.

`-quiet` suppresses terminal output. If the -log <logfile> option was specified it is still written.

`-mcf` uploads MCFs for the following templates by MCF name. This is the default upload mode.

`-key` uploads MCFs for the following templates by key.

`-datekey` uploads MCFs for the following templates by date key.

`-auto` uploads MCFs for the following templates by using the upload mode (mcf, key, or datekey) that is appropriate for the MCF name. See “MCF naming” (page 126) for more information on MCF names.

`-smcf <oldmcf> <newmcf>` uses `<newmcf>` in place of `<oldmcf>` when uploading an MCF by name. You need to use either the `-auto` option (with an appropriate MCF name) or the `-mcf` option in order to use `-smcf`.

`-skey <oldkey> <newkey>` uses `<newkey>` in place of `<oldkey>` when uploading an MCF by key. You need to use either the `-auto` option (with an appropriate MCF name) or the `-key` option in order to use `-skey`.

`-sdatekey <oldkey> <newkey>` uses `<newkey>` in place of `<oldkey>` when uploading an MCF by date key. You need to use either the `-auto` option (with an appropriate MCF name) or the `-datekey` option in order to use -`sdatekey`. See “Dated algorithm” (page 129) for information on the date format.

`-dateboundary <date>` substitutes date keys that are earlier than `<date>` but more recent than the nearest lower date boundary with `<date>`. This option is similar to `-sdatekey` but instead of substituting a specific date key, you substitute a range of date keys. You need to use either the `-auto` option (with an appropriate MCF name) or the `-datekey` option in order to use `-dateboundary`. See "Dated algorithm" (page 129) for information on the date format.

`<template file names>` is the list of template files to be regenerated. Wildcarding can be used freely. Any files that are not recognized as template files are skipped with a warning, so there is no need to be very specific when you specify template file names.

The -mcf, -key, -datekey and -auto options affect the templates that are named after them so that you can select different MCF upload modes for different templates.

The -smcf, -sdate, -sdatekey and -dateboundary options are used when one or more templates need to be regenerated using a different MCF from the one that was used to create the current version of the template. The <oldmcf> and <oldkey> option parameters refer to the existing MCF reference in the template file and the <newmcf> and <newkey> option parameters define the MCF that is to be used to regenerate the template file. The regenerated template file will refer to the replacement MCF. These MCF substitution options are normally used when an explicit MCF conversion was required or if the template MCFs are being consolidated.

The -smcf, -sdate, -sdatekey and -dateboundary options work on each template which meets the criteria specified above.

Example 1

```
/opt/MagellanNMS/bin/regentemplates ~/MagellanNMS/
  provisioningTemplates/*
```

The response is:

```
Template Regeneration Tool Version 1.0 Copyright Nortel
Networks 1997

NOTE: MC.95111100.4034.0 uploaded, version = 8.

NOTE: Regenerated /users/chrisw/MagellanNMS/\
      provisioningTemplates/ PE_5.sdt08: PM R34 PE 5
```

```
NOTE: Regenerated /users/chrisw/MagellanNMS/\
      provisioningTemplates/ PI_5.sdt08: PM R34 PE 5
      PI 5

NOTE: Regenerated /users/chrisw/MagellanNMS/\
      provisioningTemplates/ PO_1.sdt08: PM R34 \
      PE 5 PI 5 PO 1

NOTE: Regenerated /users/chrisw/MagellanNMS/\
      provisioningTemplates/ PO_2.sdt08: PM R34 \
      PE 5 PI 5 PO 2

NOTE: Regenerated /users/chrisw/MagellanNMS/\
      provisioningTemplates/ PO_4.sdt08: PM R34 \
      PE 5 PI 5 PO 4

NOTE: Regenerated /users/chrisw/MagellanNMS/\
      provisioningTemplates/ PO_5.sdt08: PM R34 \
      PE 5 PI 5 PO 5

NOTE: Regenerated /users/chrisw/MagellanNMS/\
      provisioningTemplates/ PO_6.sdt08: PM R34 \
      PE 5 PI 5 PO 6

NOTE: Regenerated /users/chrisw/MagellanNMS/|
      provisioningTemplates/ PO_7.sdt08: PM R34 \
      PE 5 PI 5 PO 7

NOTE: Regenerated /users/chrisw/MagellanNMS/\
      provisioningTemplates/ PO_8.sdt08: PM R34 \
      PE 5 PI 5 PO 8

NOTE: MC.CHRISW03.4034.0 uploaded, version = 8.

NOTE: Regenerated /users/chrisw/MagellanNMS/\
      provisioningTemplates/ PI_6.sdt08: PM R34
      PE 6 PI 6

NOTE: MC.CHRISW02.4034.0 uploaded, version = 8.

NOTE: Regenerated /users/chrisw/MagellanNMS/\
      provisioningTemplates/ PI_7.sdt08: PM R34
      PE 7 PI 7
```

```
NOTE: MC.CHRISW01.4034.0 uploaded, version = 8.

NOTE: Regenerated /users/chrisw/MagellanNMS/\
      provisioningTemplates/ PO_3.sdt08: PM R34 \
      PE 5 PI 5 PO 3
```

When explicit MCF conversion has been required for new software or the template MCFs are being consolidated, the regentemplates command needs to specify how MCFs are to be substituted. If the template MCFs are being downloaded using keys or date keys then the -key and -datekey options can be used to automatically select the latest MCF. These options apply to the template files that are named after them. The -mcf option can be used to switch back to uploading by MCF name. You can choose what kind of upload you want for each template. But even keys need to change from time to time and the -skey, -sdatekey and -dateboundary options are used to indicate how keys are to be substituted. And the -smcf option substitutes a whole MCF name.

The different types of substitution apply only to uploads of the same type. For example, -skey options only apply to template files that have been preceded by a -key option. The command line could get long and complicated so it is a good idea to keep it in a shell script. It can be coded across multiple lines by using backslash (\) at the end of each line except the last. For efficiency, the template regeneration tool groups templates by upload specification to minimize the number of uploads required to regenerate all the listed templates.

Example 2

Assume that the PE_*.std08 templates contain the MCF name MC.95111100.4034.0.

```
cd ~/MagellanNMS/provisioningTemplates
/opt/MagellanNMS/bin/regentemplates \
-logfile regentemplates.log \
-skey qwerty asdfg -skey xxxx yyyy \
-sdatekey 971002 971210 \
-dateboundary 970107 -dateboundary 980408 \
-smcf mc.xyz.4444.0 mc.abc.5555.0 \
-mcf PE_*.sdt08 \
-key PI_*.sdt08 \
-datekey PO_*.sdt08
```

In this example the PE_*.sdt08 templates have their MCFs uploaded by MCF name without substitution because the -smcf option has an MCF name that does not match the MCF name of the -mcf option. The PI_*.std08 templates have their MCFs uploaded by key, with the qwerty key replaced by asdfg and the xxxx key replaced by yyyy. The PO_*.std08 templates have their MCFs uploaded by date key with the 971002 date key replaced by 971210. Date keys other than 971002 are replaced by 970107 or 980408, whichever is the closest higher date. Date keys greater than 980408 are left alone.

Alternative Template Management using MCF backups

If the template management recommendation cannot be implemented, an alternative procedure is available when both the following are true:

- templates are being generated from MCFs downloaded to a Packet Module
- downloaded MCFs are backed up to NMS disk

Automatic MCF backup to NMS disk should be configured to ensure all MCFs used in creating templates are saved. In addition, manual MCF backup is useful for creating the initial set of backed up MCFs and for recovering from situations in which backup MCFs are missing from NMS disk, but still reside on a Packet Module.

If some templates are generated from MCFs downloaded to NMS disk, they can be included in the following procedure.

Templates requiring an MCF that hasn’t been downloaded to NMS disk or that hasn’t been backed up to NMS disk cannot be regenerated using this procedure.

Backup MCF maintenance for template regeneration

Since regenerating a template requires the presence of the source MCF, it is necessary to avoid deleting such MCFs from the backup directory during cleanup of the backup directory. Use the *templatemcfs* tool to provide a list of MCFs that need to be kept for template regeneration. The command syntax is:

```
/opt/MagellanNMS/bin/templatemcfs <template file name> ...
```

where:

`<template file name>...` is a list of template files that may require regeneration at some time in the future. Wildcarding can be used to reduce the amount of typing required to invoke this tool.

The list of MCFs to keep is placed in the template.mcfs file in the local directory. This file is then specified in the -mcffile option of the bdtidymcf command, so that these MCFs are not deleted during backup MCF cleanup, as follows.

```
/opt/MagellanNMS/bin/bdtidymcf -mcffile template.mcfs ...
```

See 241-6001-012 *Preside MDM Configuration Management for DPN User Guide* for information on the *bdtidymcf* tool.

> ***Note:*** The C shell has a limit of 1706 arguments after file name expansion, so it may be necessary to switch to the Bourne shell to invoke the templatemcfs command. Alternatively, the invocation can be placed in a Bourne shell script.

For example, if template files are kept in the ~/Magellan NMS/ provisioningTemplates directory, then the following cleans up the MCF backup directory while keeping the MCFs needed to regenerate the templates in the ~/MagellanNMS/provisioningTemplates directory.

```
/opt/MagellanNMS/bin/templatemcfs ~/MagellanNMS/
provisioningTemplates/*
/opt/MagellanNMS/bin/bdtidymcf -mcffile ./
template.mcfs -ncs corencs iwsa password
```

Template regeneration during software installation

When it is time to regenerate templates (usually during the installation of a new software release), there is a tool, *regenfrombkup*, which regenerates templates from backup MCFs. It is invoked as follows:

```
regenfrombkup <backup dir> <NMS disk dir> <template file
  name> ...
```

where:

`<backup dir>` is the directory where the backup MCFs are stored. Make sure that this directory is accessed by the workstation being used to run the regenfrombkup command, that is, it is mounted or it is a local directory.

`<NMS disk dir>` is the directory where MCFs are stored when they are downloaded to NMS disk. Make sure that this is the same as what the currently accessible PFAS server is using and that this directory is accessed by the workstation being used to run the regenfrombkup command, that is, it is mounted or it is a local directory.

`<template file name>...` is a list of template files that are to be regenerated. Wildcarding can be used to reduce the amount of typing required to invoke this tool.

> ***Note:*** The C shell has a limit of 1706 arguments after file name expansion, so it may be necessary to switch to Bourne shell to invoke the *regenfrombkup* command. Alternatively, the invocation can be placed in a Bourne shell script.

This tool:

1. goes through the specified templates to get the names of the root MCFs
2. goes through the backed up versions of the root MCFs to get a full list of MCF files needed for template regeneration
3. goes through the NMS disk directory for the MCF files that are already there
4. restores the MCF files that are needed that aren’t already on NMS disk
5. regenerates the templates by invoking the *regentemplates* tool
6. deletes the restored MCF files from NMS disk.

Only error messages are output to the terminal. The regenfrombkup.log file contains these error messages, and also error messages about templates that were successfully regenerated. Messages from previous runs are retained. Remember to look at the last entry in the regenfrombkup.log file for messages from the most recent run.

Errors do not stop the regenfrombkup tool. It regenerates as many of the specified templates as it can during a run.

If any error messages were displayed, review the error messages, make any required corrections, and rerun the regenfrombkup tool, run the regentemplates tool, or manually recreate the templates that could not be generated automatically.

Regeneration example

The following command regenerates provisioning templates from backup MCFs.

```
regenfrombkup bkup sd ~/MagellanNMS/
  provisioningTemplates/*
```

The command outputs the following to the terminal.

```
restoring needed MCFs
cp: cannot access bkup/B_MC.95111100.4034.ACR
invoking /opt/MagellanNMS/bin/regentemplates
ERROR: Invalid component id PM R34 PE 1 PE_Loader in
  /users/chrisw/MagellanNMS/provisioningTemplates/ \
  PE_Loader2.sdt08
  .save. Skipped.
ERROR: ERROR (pfa_server@bcarsdeb): Cannot open file
  sd/MC.95111100.4034.ACR for read.
ERROR:  (Unix) No such file or directory
ERROR: Skipping: /users/chrisw/MagellanNMS/ \
  provisioningTemplates/PE_5.s dt08 PM R34 PE 5
   .
   .
   .
ERROR: Skipping: /users/chrisw/MagellanNMS/ \
  provisioningTemplates/PO_8.s dt08 PM R34 PE 5 PI 5 PO 8
ERROR: ERROR (pfa_server@bcarsdeb): Cannot open file
  sd/MC.95111100.4034.ACR for read.
ERROR:  (Unix) No such file or directory
ERROR: Skipping: /users/chrisw/MagellanNMS/ \
  provisioningTemplates/PO_3.s dt08 PM R34 PE 5 PI 5 PO 3
deleting restored MCFs

template regeneration from backup MCFs ended
```

Also an entry is added at the end of the *regenfrombkup.log* file. In addition to the error messages that were displayed on the terminal, the log file contains informational messages showing the templates that were regenerated successfully. The log file is as follows:

```
*****************************************************
*     Template Regeneration from NMS DISK backup    *
*     v1.0  Copyright Nortel Networks 1997          *
*     Time: Fri May 9 10:58:36 EDT 1997             *
*                                                   *
*****************************************************

restoring needed MCFs
cp: cannot access bkup/B_MC.95111100.4034.ACR
invoking /opt/MagellanNMS/bin/regentemplates

------------------------------------------------------

Template Regeneration Tool  Version 1.1  Copyright
Nortel Networks 1997

Time: 1997.05.09 10:58

Command: /opt/MagellanNMS/bin/regentemplates -logfile
regenfrombkup.log -quiet
/users/chrisw/MagellanNMS/provisioningTemplates/ \
   DNA_CUG.sdt08
/users/chrisw/MagellanNMS/provisioningTemplates/ \
   NCUG_Index_1.sdt08
   .
   .
   .
/users/chrisw/MagellanNMS/provisioningTemplates/ \
   X_Prefix_DNAs.sdt08
/users/chrisw/MagellanNMS/provisioningTemplates/save

ERROR: Invalid component id PM R34 PE 1 PE_Loader in
   /users/chrisw/MagellanNMS/provisioningTemplates/ \
   PE_Loader2.sdt08
   .save. Skipped.
WARNING: /users/chrisw/MagellanNMS/ \
   provisioningTemplates/save is not a template file.
   Skipped.
NOTE: MC.NMS02.4034.0 uploaded, version = 8.
```

```
NOTE: Regenerated /users/chrisw/MagellanNMS/ \
   provisioningTemplates/DNA_C UG.sdt08: PM R34 PE 9 PI 9 \
   PO 1 FRAMERELAY 0 FRLDNA X12345 FRDNACUG X12345
NOTE: Regenerated /users/chrisw/MagellanNMS/ \
   provisioningTemplates/NCUG_ Index_1.sdt08: PM R34 \
   CONMAN_SDA 0 ICON_SDA ACCOUNT ICONDNACUG \
   X30214034401002 CUGS NULL CUG 1
   .
   .
   .
NOTE: Regenerated /users/chrisw/MagellanNMS/ \
   provisioningTemplates/X_Prefix_DNAs.sdt08: PM R34 PE \
   4 PI 4 PO 1 X75_SL_GATEWAY 0 X_ADDRESSSCREENING 0 \
   X_PFXDNAGROUP NULL
ERROR: ERROR (pfa_server@bcarsdeb): Cannot open file
   sd/MC.95111100.4034.ACR for read.
ERROR:  (Unix) No such file or directory
ERROR: Skipping: /users/chrisw/MagellanNMS/ \
   provisioningTemplates/PE_5.s dt08 PM R34 PE 5
   .
   .
   .
ERROR: Skipping: /users/chrisw/MagellanNMS/ \
   provisioningTemplates/PO_8.s dt08 PM R34 PE 5 PI 5 \
   PO 8
NOTE: MC.CHRISW03.4034.0 uploaded, version = 8.
NOTE: Regenerated /users/chrisw/MagellanNMS/ \
   provisioningTemplates/PI_6. sdt08: PM R34 PE 6 PI 6
NOTE: MC.CHRISW02.4034.0 uploaded, version = 8.
NOTE: Regenerated /users/chrisw/MagellanNMS/ \
   provisioningTemplates/PI_7. sdt08: PM R34 PE 7 PI 7
ERROR: ERROR (pfa_server@bcarsdeb): Cannot open file
   sd/MC.95111100.4034.ACR for read.
ERROR:  (Unix) No such file or directory
ERROR: Skipping: /users/chrisw/MagellanNMS/ \
   provisioningTemplates/PO_3.s dt08 PM R34 PE 5 PI 5 PO 3
deleting restored MCFs

template regeneration from backup MCFs ended on Fri May  9
10:59:19 EDT 1997
```

In this example, not all templates were regenerated. A couple of the template files: `PE_Loader2.sdt08.save` and `save`, are not proper template files (and are no cause for concern). But you need to look at the template files that require MC.95111100.4034.ACR for regeneration. Near the start of the log file, it shows that the backup of MC.95111100.4034.ACR is not in the backup directory.

If this file can be found and restored to the backup directory as B_MC.95111100.4034.ACR or to the NMS disk directory as MC.95111100.4034.ACR, the regenfrombkup tool can be rerun. This can be a complete rerun or the list of templates can be reduced to only those that haven't been regenerated successfully. If the missing file cannot be found, suitable alternative service data needs to be found and the corresponding template files need to be regenerated by invoking the regentemplates tool with one or more substitution options. It may be necessary to recreate the templates manually.

Service data conversion requirements

The following sections detail the possible actions and message responses associated with service data conversion.

Add a field

When the conversion must add a new field, and that new field causes the extension of an existing component, the field is created according to the information (value and position) obtained from the current service data description files. The format of the message is:

```
NOTE: The field <field:value> has been added.
```

If this field is added in the service data but is not visible through Component Provisioning, the format of the message is:

```
NOTE: An invisible field has been added under the
component <component:value>*
```

Replace a field

The value of a field can be replaced by a new value. The replacement can be a simple value replacement, an out of range detection, or based on some limited conditions. Normally, a conditional replacement is based on the result of calculations or the value of other related fields. The format of the message is one of:

```
NOTE: The field <field:value> has been set to the
specified value.
```

```
NOTE: The field <field:value> was out of range and has been
set to the specified value.
```

If the field is replaced in the service data but is not visible through Component Provisioning, the format of the message would be one of:

```
NOTE: An invisible field has been replaced under the
component <component:value>.
```

```
NOTE: An invisible field was out of range and has been
replaced under the component <component:value>.
```

Examples

In the following example, the field TPUTCUTOFF had a value other than 245 and the Service Data Conversion tool reset it to 245.

```
The field RID_ROUTING 1 SYSTEM_ATTRIBUTES 1
TPUTCUTOFF:245 has been replaced with the specified
value.
```

In the following example, the field DIALINTIMER is using a spare location in the service data. Its value was not in the allowed range of values and the Service Data Conversion tool has reset it to its default value of 30.

```
The field PE 23 PI 24 PO 7 ITI 0 ITILINK 0
DIALINTIMER:30 was out of range and has been reset to
the specified value.
```

Add an envelope

When a conversion requires a new mandatory envelope to be added to an existing service, the envelope itself and its subordinates are created. The hierarchical structure of the new envelope and the default values of the envelope and its fields are obtained from the new current service data description files. The format of the message is:

```
NOTE: The component <component:value> has been added along
with its associated fields.
```

If the component is added in the service data but is not visible through Component Provisioning, the format of the message is:

```
NOTE: An invisible component has been added under the
component <component:value>.
```

No messages are produced for individual fields within a new envelope.

Obsolete an envelope

If an envelope is no longer required in the service data or needs to be removed under some conditions, a delete envelope operation takes place in the conversion process. The delete operation removes the envelope and its children accordingly from the hierarchy. The format of the message is:

```
NOTE: The component <component:value> has been deleted.
```

If the component is deleted in the service data but is not visible through Component Provisioning, the format of the message is:

```
NOTE: An invisible component has been deleted under the
component <component:value>.
```

No messages are displayed for individual fields within a deleted envelope.

SD_version =5

In the following sections the term *offset* is used. The *offset* contains three numbers, from left to right, the first digit indicates the word number, the second digit indicates the bit number and the third digit indicates the length of the field. See 241-2001-340 *DPN-100 Envelope Definitions* for more information on service data fields.

The following components make up the service data conversion requirements for SD_version = 5 (R7).

Broadcast Server

BROADCASTDNACUG component

Field:	N/A (invisible field)
UI_Group:	N/A
UI_Prompt:	N/A
Offset:	Variable
Action:	Replace a field
Value:	The value of the Packet Size field
Critical:	No

BROADCASTSERVER component

Field:	BSCTMULTICASTTYPE
UI_Group:	Broadcast service parameters
UI_Prompt:	Multicast type
Offset:	8,0,4
Action:	Add a field
Value:	Uni-directional
Critical:	Yes

Field:	BCSTACCEPTDESCENDANT
UI_Group:	Broadcast service parameters
UI_Prompt:	Accept descendant calls
Offset:	8,4,1,
Action:	Add a field
Value:	0
Critical:	Yes

Field:	BCSTOUTGOINGCALLTYPE
UI_Group:	Broadcast service parameters
UI_Prompt:	Outgoing call type
Offset:	8,5,2
Action:	Add a field
Value:	SVC
Critical:	Yes

Field:	N/A (invisible field)
UI_Group:	N/A
UI_Prompt:	N/A
Offset:	8,7,9

Action:	Add a field
Value:	0
Critical:	No

Field:	BCSTOUTGOINGCALLLCN
UI_Group:	Broadcast service parameters
UI_Prompt:	Outgoing call LCN
Offset:	9,0,16
Action:	Add a field
Value:	0
Critical:	Yes

Frame Relay

FRDLCI component

Field:	COMMITBURST
UI_Group:	Bandwidth management
UI_Prompt:	Bc (bits)
Offset:	6,0,32
Action:	Replace a field when the invisible field described below is set to 0
Value:	64000
Critical:	No

Field:	N/A (invisible field)
UI_Group:	N/A
UI_Prompt:	N/A
Offset:	12,2,2
Action:	Replace a field
Value:	1
Critical:	No

FRDNACUG component

Field:	FRROUTINGDEFAULT
UI_Group:	Class of service parameters
UI_Prompt:	Routing Default
Offset:	21,0,1
Action:	Replace a field
Value:	Throughput
Critical:	No

Field:	N/A (invisible field)
UI_Group:	N/A
UI_Prompt:	N/A
Offset:	Variable
Action:	Replace a field
Value:	The value of the Packet Size field
Critical:	No

Console Manager

ICONDNACUG component

Field:	N/A (invisible field)
UI_Group:	N/A
UI_Prompt:	N/A
Offset:	Variable
Action:	Replace a field
Value:	The value of the Packet Size field
Critical:	No

ITI service

ITIDNACUG component

Field:	N/A (invisible field)
UI_Group:	N/A
UI_Prompt:	N/A
Offset:	Variable
Action:	Replace a field
Value:	The value of the Packet Size field
Critical:	No

Point of Sale

POSDNACUG component

Field:	N/A (invisible field)
UI_Group:	N/A
UI_Prompt:	N/A
Offset:	Variable
Action:	Replace a field
Value:	The value of the Packet Size field
Critical:	No

RID Routing

RID_ROUTING component

Field:	N/A (invisible field)
UI_Group:	N/A
UI_Prompt:	N/A
Offset:	x,0,8 where x = 135 to 263
Action:	Add a field
Value:	127
Critical:	No

Field:	N/A (invisible field)
UI_Group:	N/A
UI_Prompt:	N/A
Offset:	x,8,7 where x = 135 to 263
Action:	Add a field
Value:	127
Critical:	No

Field:	N/A (invisible field)
UI_Group:	N/A
UI_Prompt:	N/A
Offset:	x,15,1 where x = 135 to 263
Action:	Add a field and Replace a field
Value:	Added with a value of 0 and Replaced with the value of the Overflow x field
Critical:	No

SYSTEM_ATTRIBUTES component

Field:	DELAYCUTOFF
UI_Group:	System attribute options
UI_Prompt:	Delay cutoff
Offset:	16,0,8
Action:	Add a field
Value:	250
Critical:	No

Field:	TPUTCUTOFF
UI_Group:	System attribute options
UI_Prompt:	Throughput cutoff
Offset:	6, 8, 8
Action:	Replace a field
Value:	245
Critical:	No

Field:	N/A (invisible field)
UI_Group:	N/A
UI_Prompt:	N/A
Offset:	16,8,8
16,8,8	Add a field
Value:	0
Critical:	No

Field:	DELAYMETICCONSTANT
UI_Group:	System attribute options
UI_Prompt:	Delay metric constant
Offset:	17,0,16
Action:	Add a field
Value:	6
Critical:	No

SABRE service

SBRDNACUG component

Field:	N/A (invisible field)
UI_Group:	N/A
UI_Prompt:	N/A
Offset:	Variable
Action:	Replace a field
Value:	The value of the Packet Size field
Critical:	Yes (applies to whole envelope)

SBSC service

SBSCDNACUG component

Field:	N/A (invisible field)
UI_Group:	N/A
UI_Prompt:	N/A
Offset:	Variable
Action:	Replace a field
Value:	The value of the Packet Size field
Critical:	Yes (applies to whole envelope)

SNA service

SNADNACUG component

Field:	N/A (invisible field)
UI_Group:	N/A
UI_Prompt:	N/A
Offset:	Variable
Action:	Replace a field
Value:	The value of the Packet Size field
Critical:	Yes (applies to whole envelope)

Token Ring service

TRSNAPADDNACUG component

Field:	N/A (invisible component)
UI_Group:	N/A
UI_Prompt:	N/A
Offset:	Variable
Action:	Replace a field
Value:	The value of the Packet Size field
Critical:	Yes (applies to whole envelope)

X.25 service

X25DNACUG component

Field:	N/A (invisible field)
UI_Group:	N/A
UI_Prompt:	N/A
Offset:	Variable
Action:	Replace a field
Value:	The value of the Packet Size field
Critical:	No

X.25 Gateway service

X2GTYDNACUG component

Field:	N/A (invisible field)
UI_Group:	N/A
UI_Prompt:	N/A
Offset:	Variable
Action:	Replace a field
Value:	The value of the Packet Size field
Critical:	No

Trunk and UTP Network Link

UTPLINK component

Field:	N/A (invisible field)
UI_Group:	N/A
UI_Prompt:	N/A
Offset:	21,0,8
Action:	Add a field
Value:	70
Critical:	No

Field:	HIGHUTILLEVEL
UI_Group:	Adaptive routing parameters
UI_Prompt	High level threshold
Offset:	21,8,8
Action:	Add a field
Value:	80
Critical:	No

Field:	TIMEINTLOWUTIL
UI_Group:	Adaptive routing parameters
UI_Prompt:	Low time interval (min)
Offset:	22,0,8
Action:	Add a field
Value:	2
Critical:	No

Field:	TIMEINTHIGHUTIL
UI_Group:	Adaptive routing parameters
UI_Prompt:	High time interval (min)
Offset:	22,8,8
Action:	Add a field
Value:	5
Critical:	No

Field:	THMETRICCHANGE
UI_Group:	Adaptive routing parameters
UI_Prompt:	Throughput metric increase (%)
Offset:	23,0,8
Action:	Add a field
Value:	15

Critical:	No

Field:	DLMETRICCHANGE
UI_Group:	Adaptive routing parameters
UI_Prompt:	Delay metric increase (%)
Offset:	23,8,8
Action:	Add a field
Value:	0
Critical:	No

Field:	DELAYOVERRIDE
UI_Group:	Delay override (one way)
UI_Prompt:	Delay override (ms)
Offset:	24,0,16
Action:	Add a field
Value:	Undefined
Critical:	Yes

SD_version = 6

The following components make up the service data conversion requirements for SD_version=6 (R8).

DSOB service

DS0BLINK component

Field:	LINKSETUPTIMER
UI_Group:	Timers
UI_Prompt:	Link Setup Timer
Offset:	6,0,8
Action:	Replace a field when out of the range
Value:	1
Critical:	Yes

Field:	DIALINTIMER
UI_Group:	Timers
UI_Prompt:	Dial in Timer
Offset:	6,8,8
Action:	Replace a field when out of the range
Value:	30
Critical:	Yes

ITI service

ITILINK component

Field:	LINKSETUPTIMER
UI_Group:	Timers
UI_Prompt:	Link setup timer
Offset:	6,0,8
Action:	Replace a field when out of the range
Value:	1
Critical:	Yes

Field:	DIALINTIMER
UI_Group:	Timers
UI_Prompt:	Dial in timer
Offset:	6,8,8
Action:	Replace a field when out of the range
Value:	30
Critical:	Yes

SNA service

SNALINK component

Field:	SNALINESPEED
UI_Group:	Other parameters
UI_Prompt:	Line speed
Offset:	14,0,8
Action:	Replace a field
Value:	Value is copied from the old line speed location (5,11,5)
Critical:	Yes

SD_version = 7

The following components make up the service data conversion requirements for SD_version=7 (R8.3).

ITI service

ITILINK component

Envelope expanded by one word, two new fields added.

Field:	INPUTSPEED
UI_Group:	Interface parameters
UI_Prompt:	Input speed
Offset:	10,0,8
Action:	Add a field
Value:	Value is copied from the old input speed location (5,5,4)
Critical:	Yes

Field:	OUTPUTSPEED
UI_Group:	Interface parameters
UI_Prompt:	Output speed
Offset:	10,8,8
Action:	Add a field
Value:	Value is copied from the old output speed location (5,9,4)
Critical:	Yes

DS0B service

DS0BLINK component

Envelope expanded by one word, two new fields added.

Field:	INPUTSPEED
UI_Group:	Interface parameters
UI_Prompt:	Input speed
Offset:	10,0,8
Action:	Add a field
Value:	Value is copied from the old input speed location (5,5,4)
Critical:	Yes

Field:	OUTPUTSPEED
UI_Group:	Interface parameters
UI_Prompt:	Output speed
Offset:	10,8,8

Action:	Add a field
Value:	Value is copied from the old output speed location (5,9,4)
Critical:	Yes

Token Ring service

ISRB component

Envelope expanded by 24 words, 4 fields and 19 invisible fields.

Field:	NUMCOMPRESSMACHEADER
UI_Group:	N/A
UI_Prompt:	Total of compressible MAC headers
Offset:	30,0,16
Action:	Add a field
Value:	512
Critical:	Yes

Field:	NUMLWVCCOMPRESSMACHEADER
UI_Group:	N/A
UI_Prompt:	Compressible MAC headers per LWVC
Offset:	31,0,8
Action:	Add a field
Value:	10
Critical:	Yes

Field:	N/A (invisible field)
UI_Group:	N/A
UI_Prompt:	N/A
Offset:	Offset:
Action:	Add a field
Value:	0
Critical:	No

Field:	SLRESERVICEMAXDATARATE
UI_Group:	N/A
UI_Prompt:	Service maximum data rate
Offset:	32,0,32
Action:	Add a field
Value:	64000
Critical:	Yes

Field:	SLRESERVICENORMALDATARATE
UI_Group:	N/A
UI_Prompt:	Service normal data rate
Offset:	34,0,32
Action:	Add a field
Value:	64000
Critical:	Yes

18 times this invisible field of 16 bits:

Field:	N/A (invisible field)
UI_Group:	N/A
UI_Prompt:	N/A
Offset:	x,0,16 where x = 36 to 53
Action:	Add a field
Value:	0
Critical:	No

UTP service

Dial Up component

An invisible field becomes visible with a new default of 1.

Field:	UTPENAAUTODISABLE
UI_Group:	UTP specific dial parameters
UI_Prompt:	Automatic disabling allowed
Offset:	5,11,1
Action:	Replace a field
Value:	1
Critical:	No

Envelope expanded by two words. (1 field and 2 invisible fields)

Field:	UTPHEARTBEATALARMINT
UI_Group:	UTP specific dial parameters
UI_Prompt:	Heart beat alarm
Offset:	156,0,8
Action:	Add a field
Value:	5

Critical:	No

Field:	N/A (invisible field)
UI_Group:	N/A
UI_Prompt:	N/A
Offset:	156,8,8
Action:	Add a field
Value:	0
Critical:	No

Field:	N/A (invisible field)
UI_Group:	N/A
UI_Prompt:	N/A
Offset:	157,0,16
Action:	Add a field
Value:	0
Critical:	No

Frame Relay service

DNA/CUG component

This invisible field has a new default of 30

Field:	N/A (invisible field)
UI_Group:	N/A
UI_Prompt:	N/A
Offset:	10,0,8
Action:	Replace a field
Value:	30
Critical:	No

SD_version = 8

The following components make up the service data conversion requirements for SD_version=8 (R9.0).

ITI service

The ITILINK component

This envelope has been expanded by 3 word (2 new fields and 2 invisible fields). Following fields are new:

Field:	BTimer
UI_Group:	Timers
UI_Prompt:	B time
Offset:	11,0,8
Action:	Add a field
Value:	0

Field:	CallSetupTimer_ext
UI_Group:	Timers
UI_Prompt:	Call setup timer extension
Offset:	11,8,8
Action:	Add a field
Value:	0

Field:	N/A (invisible field)
UI_Group:	N/A
UI_Prompt:	N/A
Offset:	12,0,16
Action:	Add a field
Value:	0

Field:	N/A (invisible field)
UI_Group:	N/A
UI_Prompt:	N/A
Offset:	13,0,16
Action:	Add a field
Value:	0

DS0B service

The DS0BLINK component

This envelope has been expanded by 3 word (1 new field and 3 invisible fields). Following fields are new:

Field:	BTimer
UI_Group:	Timers
UI_Prompt:	B timer
Offset:	11,0,8
Action:	Add a field
Value:	0

Field:	N/A (invisible field)
UI_Group:	N/A
UI_Prompt:	N/A
Offset:	11,8,8
Action:	Add a field
Value:	0

Field:	N/A (invisible field)
UI_Group:	N/A
UI_prompt:	N/A
Offset:	12,0,16
Action:	Add a field
Value:	0

Field:	N/A (invisible field)
UI_Group:	N/A
UI_prompt:	N/A
Offset:	13,0,16
Action:	Add a field
Value:	0

Chapter 8 Operational recommendations

This chapter provides some guidelines to follow if you are having trouble with the provisioning applications. The provisioning applications and facilities must be operational and accessible in order for Configuration for DPN Devices provisioning to operate properly.

In this chapter, you can find the following information:

- “Component Provisioning operational constraints” (page 119)
- “Troubleshooting” (page 120)

Component Provisioning operational constraints

The following sections describe some of the operational constraints associated with Configuration for DPN Devices.

Memory usage

There are no warnings when the memory utilization is nearing its maximum limit, but this state can be monitored by the Memory Utilization tool. To access the Memory Utilization tool, in the Preside MDM window, select System -> Utilities -> Memory Utilization.

If the memory tool shows excessive memory use (>90%), then there are two options available:

- Expand only at the lowest component level that is being modified and do not fully expand a module.

- Increase the swap space on the workstation. Increasing the real memory on the workstation will improve the performance, but it will not solve the memory shortage.

Error messages

Component Provisioning will not permit service data with an error to be downloaded to a module. Therefore, it is necessary to correct all errors before being able to perform a successful download.

The error messages issued by the system are very technical in some cases. Often they are preceded with a string identifying the process which detected the error. FA, SEA or PFA are common message prefixes. In addition, some messages contain internal field names for service data elements which are involved in a semantic error.

Multiple instances

Multiple instances of the Component Provisioning tool can be started to facilitate cut, copy, and paste operations between modules or between MCFs on a single module. However, when working with multiple instances of the tool, ensure that sufficient memory and swap space is available.

Troubleshooting

When selecting an application from the DPN Devices configuration toolset, the following items must be accessible to ensure correct operation of the tool.

Provisioning File Access server

The Provisioning File Access server (PFAS) must be operating and must be accessible from the workstation trying to access the provisioning tools. This process provides access to the network for both NCS operator authentication and file transfer operations to the packet modules. The following is a list of potential problems:

- Provisioning Agent cannot be activated by PFAS. The most common error is from mismatched X.25 parameters between SunLink and the X.25 port on the DPN-100 module. Another common error is that the incoming reverse facility is disabled. This should be corrected with the appropriate SunLink tools.

- PFAS is not operating. We recommend that you use the Server Administration tool to ensure that the server is operating. There are two ways to start the Server Manager administrator tool:
  - — Logged as *root*, start the Server Administration tool from the Preside Multiservice Data Manager (MDM) Administration toolset.
  - — Logged as provisioning user, start the tool manually from a UNIX window:

    ```
    % /opt/MagellanNMS/bin/svmadm &
    ```

    In both cases, ensure that the PM File Access Server is running, if not start it.

- The workstation is not operating a local multi nodal name server (MNS) process. This prevents the provisioning applications from starting up since they cannot register with MNS to find the required server processes.
  - — Verify that MNSD is operating. From the UNIX window enter

    ```
    % ps -def | grep mnsd
    ```

  - — If MNSD is not present, try to start it manually, enter

    ```
    % /opt/MagellanNMS/bin/mnsd
    ```

    If MNS is still not present, re-install the MDM software or contact the System Administrator.

The points listed above also apply to the Software Download Server.

Access to the NCSMGR process

Configuration for DPN Devices requires access to NCS to perform operator authentication and to issue a command to the target module to request a provisioning session. The following is a list of potential problems:

- Ensure that the configuration parameter NCS_HOST in the configuration file for PFAS indicates a host where NCSMGR is operating. Ensure that NCSMGR is operating on the indicated host. To do this, start the Server

Administrator tool and ensure that the NCS Communication Manager is running. Any problems in this area results in authentication failure error messages.

- Another method to verify that NCSMGR is available is to use the NCS Connect Console and verify that the operator can logon using this tool. If this method succeeds, then there is no problem with NCSMGR.

SunLink on the Preside Multiservice Data Manager (MDM) workstation

Both the NCSMGR and PFA servers require access to an operational X.25 port on the workstation where the processes are running. Special considerations for provisioning include:

- Ensure that any CUGs used on the DNA for the X.25 port on the workstation matches the preferred CUG(s) on the DNA of the Provisioning Agent (PAGENT) for the module you are attempting to provision. Any mismatches result in a time-out message from the provisioning tools.
- If provisioning across an X.75 link, ensure that International Calls are allowed for the PAGENT’s DNA of the target module. The parameter Out international must be turned ON.
- Ensure that the full X121 DNA is specified in the PFA configuration file for the HOST_ADDRESS parameter. This is the DNA of the X.25 port on the workstation that the PFA server uses in the call request command that it uses from the PAGENT on the target module.
- Ensure the Incoming Reverse Charging facility is set in order to accept calls from the Provisioning Agent.

Appendix A
Uploading and downloading MCFs

This appendix describes the procedures for uploading and downloading master configuration files (MCFs). In this appendix, you can find the following information:

Service data for the DPN-100

Service data created by Configuration for DPN Devices is stored on disks attached to either an Preside Multiservice Data Manager (MDM) workstation or a DPN-100 module. When a provisioning session is started, the service data must be retrieved from the disk before it can be changed. This operation is called uploading. When a provisioning session is completed, the service data is placed back on the disk. This operation is called downloading.

Many DPN Devices configuration applications involve uploading and downloading service data from DPN-100 modules. Some important fundamentals involved with the upload and download operations are described in the following sections.

Master configuration files

An MCF is a hierarchical set of MC (master configuration) files containing binary service data used by DPN switches. At the top of the hierarchy is the root MCF. Usually, the entire MCF is referred to by the name of the root MCF. A root MCF is specified as follows:

```
MC.<bundle>.<namsid>.0
```

where:

`bundle` is 1 to 8 alphanumeric or underscore characters.

`namsid` is a numeric value from 256 to 49151.

Example
MC.93061603.9999.0

Each MC file, including the root MCF, contains references to other MC files and/or service data; this forms the hierarchy. This paradigm allows multiple MCFs to share subordinate MC files.

NAMS ID

The NAMS ID is an integer between 256 and 49151 which uniquely identifies a packet module in the network. Configuration for DPN Devices treats the NAMS ID as an optional field when uploading and downloading MCFs.

> ***Note:*** When uploading an MCF from an NMS disk the NAMS ID is mandatory.

When uploading MCFs, if the NAMS ID is not specified, the NAMS ID of the COMMITTED MCF on the module is used.

When downloading MCFs, if the NAMS ID is not specified, the NAMS ID of the uploaded MCF is used.

Location

MCFs are typically stored on the local disk of the packet module (PM) to which they apply. They may also be stored on the disks of other PMs or an NMS disk.

When uploading or downloading MCFs to an NMS disk, the actual directory used is the one specified in the configuration file /opt/MagellanNMS/cfg/PFA.cfg. The directory is locally accessible from the Preside Multiservice Data Manager (MDM) host on which the PFA server is running.

Authentication

During upload and download the Preside Multiservice Data Manager (MDM) issues an NCS command to the module containing the DNA of the MDM host. The module then places a call to the Preside MDM. This activity can be done automatically by the application if the module is reachable via NCS and the authentication information is supplied by the user for validation. Authentication is required to associate the Configuration for DPN Devices session with an OA in the network. The Destination, User ID and Password must be specified before authentication occurs.

All DPN Devices configuration tools use the Connection Manager for authentication.

Manual access mode

Configuration for DPN Devices has the capability to access a module manually. Manual access mode is used when the call from the module to the Preside Multiservice Data Manager (MDM) needs to be placed manually. For example, this would be necessary if a module was unreachable by NCS.

How to establish a manual call

1. In the Preside MDM window, select Configuration -> DPN Devices -> the Component Provisioning.

   The Connection Manager dialog appears.

2. Select the *Manual Access Mode* check box.
3. Click *Authenticate* or hit the return key.

   The Connection Manager dialog is closed.

4. Enter the component id in the *Component* area field.
5. Click *Expand* or press enter.
6. Issue the CALL command either from the local operator console for the module or via the NCS command console. The command syntax is a follows:

```
con pagent call x <x121 address> pagent <lock id>
```

The time in which this command must be entered is specified in the *PFA.cfg* file, the default is 30 seconds.

Note: The userdata field must be PAGENT in order for the call to be established, otherwise the call request will be cleared with a code of C1.

NCS routing

When uploading or downloading MCFs to a module, the specific NCS routing information for locating that module can be specified. This explicit routing information is used to navigate through the NCS hierarchy from the OA in which the DPN Devices configuration session is logged to the module. For example, the path to <pm_mnemonic> in TOR5 can be specified as:

```
TOR1-TOR2-TOR5-<pm_mnemonic>
```

When a backup OA is created it is usually given the same name as the primary OA, except *-B* is added to the end. For example, *OANAME -B* is the backup for *OANAME*. DPN-100 NCS supports path routing to the module when the backup OA becomes the primary OA. Configuration for DPN Devices allows the use of this implicit path routing to access a module. However, Configuration for DPN Devices does not support explicit naming of a path route that includes a backup OA name with the form <*OANAME*> *-B*.

MCF naming

There are five different naming conventions for identifying MCFs when uploading and downloading from the modules.

Active

Active mode allows you to upload the active MCF on a given module.

> ***Note:*** Active mode does not apply when downloading or to MCFs stored on an NMS disk.

Committed

Committed mode allows you to upload the committed MCF on a given module.

> ***Note:*** Committed mode does not apply to downloading or to MCFs stored on an NMS disk.

User specified

The most flexible mode, user specified, allows you to explicitly specify the bundle and NAMS ID which comprise the MCF. This mode can be used to name any MCF. For example:

```
MC.MYBUNDLE.9999.0
```

When uploading MCFs in user specified mode, you supply the upload bundle (and optionally NAMS ID). If the specific MCF exists, it is uploaded.

When downloading MCFs in user specified mode, you supply the download bundle (and optionally NAMS ID). If the specific MCF does not exist, it is downloaded. If the specific MCF already exists, the download will fail.

Keyed

Keyed mode interprets the bundle field as a key concatenated with an index. A keyed MCF is specified as follows:

`MC.<key><index>.<namsid>.0`

where:

`key` is 1 to 6 alphanumeric or underscore characters.

`index` is a numeric value from 00 to 99.

`namsid` is a numeric value from 256 to 49151.

Example

```
MC.MYKEY44.9999.0
```

When uploading MCFs in keyed mode, you supply the upload key (and optionally NAMS ID). The latest MCF (the one with the greatest index) is retrieved based on the MCFs matching the pattern as follows:

```
MC.<key>*.<namsid>.0
```

> ***Note:*** If no MCFs exist with that key, the upload fails.

When downloading, the same logic is applied and the next in the sequence is created. The user specifies a download key (and optionally NAMS ID) and Configuration for DPN Devices determines which MCF is to be downloaded.

For example, given the key *MYKEY*, if the following MCFs exist at the specified location,

```
MC.92010110.9999.0
MC.92010120.9999.0
MC.MYKEY19.9999.0
MC.MYKEY20.9999.0
MC.0001019.9999.0
```

MCF *MC.MYKEY20.9999.0* would be the latest.

Dated

Dated mode is similar to keyed mode except the key is a date in the form: yymmdd where yy represents the year, mm the month, and dd the day. A dated MCF is specified as follows:

`MC.<yymmdd><index>.<namsid>.0`

Example

```
MC.93061612.9999.0
```

When uploading MCFs in dated mode, you supply the upload date (either a valid date or *today*) and optionally, NAMS ID, and the latest MCF is retrieved based on the MCFs matching the pattern as follows:

```
MC.*.<namsid>.0
```

The bundle portion is processed and those MCFs having an 8-digit bundle where the first 6 digits are a valid date are considered. The MCF with the greatest date not exceeding the upload date is selected.

> ***Note:*** If no MCFs exist with the upload date, the upload does not necessarily fail; the next latest MCF is chosen.

When downloading, the same logic is applied and the next in the sequence is created. You specify a download date (either a valid date or *today*) and the system determines which MCF is to be downloaded.

This allows the provisioning system to determine, given a date, the *latest* MCF relative to that date.

For example, given the date *970101*, if the following MCFs exist at the specified location,

```
MC.92010100.9999.0
MC.92010110.9999.0
MC.92010120.9999.0
MC.99010120.9999.0
MC.0001019.9999.0
```

MCF *MC92010120.9999.0* would be the latest.

Dated algorithm

To account for the year 2000 and beyond, Configuration for DPN Devices interprets 000101 as later than 991231. The base year has been chosen as 1980. This means 800101 precedes 900101 which precedes 000101 which precedes 790101. In other words:

```
if (yy < 80)
   year = 2000 + yy
else
   year = 1900 + yy
```

This means:

80 => 1980
90 => 1990
99 => 1999
00 => 2000
10 => 2010
79 => 2079

Activation date

In Preside Multiservice Data Manager (MDM), an edition-issue has an activation date associated with it. The activation date is in the following format in the SDA header: yyyymmddhhmmssss where yyyy is the year, mm is the month, dd is the day, hh is the hour, mm is the minute and ssss is the seconds.

When a download takes place this activation date is stored in each SDA created (in the SDA header). Although no support exists on-switch to activate service data based on this date, some customers use it to manage their service data and to run network service integrity checks based on activation date.

When uploading MCFs the activation date may or may not be displayed.

When downloading MCFs in dated mode, the activation date will be set to the download date or it may be specified (either a valid date or *today*). If the date is not specified, the activation date of the uploaded MCF is propagated to the downloaded MCF.

Download type

There are two different download types: incremental and complete. Incremental download means only the MC files containing modified components are created. Complete download means all MC files are created.

Appendix B
ICONS

This appendix describes the Data Spooling Dump, PAGENT and RDS ICONs. In this appendix, you can find the following information:

- “Data Spooling Dump ICON” (page 131)
- “PAGENT ICON” (page 132)
- “RDS ICON” (page 133)

Data Spooling Dump ICON

The following components and parameters are required on a module for service data backup to the Backup disk.

- Add component DataSpooling under PM. The system will automatically add the DNACUG component, which must be completed.
- Add DataSpoolingDump and DataSpoolingDirectCall components under DataSpooling component, and complete the service data.
- Parameters for DataSpoolingDump
  - — DataSpoolingDump key is an index from 0 to 4.
  - — DNA of data collection system, that is, the X.25 port DNA where the backup workstation is located.
  - — Start and End busy times for automatic dump based on the disk occupancy.
  - — Daily Scheduled Dump Time for automatic daily dump.
  - — Data Spooling Dump Applications - Select *Backup* and *Schedule dump flag*. The *Schedule dump flag* is to activate daily dump. DPN-

100/1 does not support local collection of alarms, Accounting, etc. DPN-100/1 also does not support Dump based on the disk occupancy.

- Parameters for Data Spooling Direct Call

  The key for the direct Call must be the same as the key for the corresponding DataSpoolingDump component.

  - DNA of data collection system - this must be the same DNA as in DataSpoolingDump.
  - User data - must contain the hex value C4.

PAGENT ICON

The following information is necessary for the correct functioning of the PAGENT ICON.

- ICON service type Provisioning agent (PA) must be set.
- Maximum number of LCNs - 2 to 8 based on the number of simultaneous accesses allows to/from the Preside Multiservice Data Manager (MDM) workstation.
- NAMs data streams - all off.
- NAMs critical streams - are all unset.
- Add an ICON_Direct_Call component to the PAGENT ICON for the Software Distribution system. The remote DNA field should be the DNA of the RDS ICON in the RDS. In the ICON Direct Call Facilities, Facilities should be set to *off* and all other fields left blank.
- If the X.25 which connects to MDM workstation has a CUG, then the PAGENT ICON must have a preferred CUG which matches.
- The DNA of the PAGENT ICON must have the reverse charge facility set ON.
- The key value for the ICON must be called PAGENT.
- Calls on command must be set.

RDS ICON

None of the ICON options should be set in the ICON Service Envelope for the RDS ICON.

Index

Preside Multiservice Data Manager

Configuration Management for DPN Administration

Release: R15.1

Publication: 241-6001-304
Document status: Standard
Document version: 15.1RSUP
Document date: August 2004
Printed in Canada