Critical Release Notice

Publication number: 297-2621-320
Publication release: Standard 10.01

The content of this customer NTP supports the SN06 (DMS) software release.

Bookmarks used in this NTP highlight the changes between the NA015 baseline and the current release. The bookmarks provided are color-coded to identify release-specific content changes. NTP volumes that do not contain bookmarks indicate that the NA015 baseline remains unchanged and is valid for the current release.

Bookmark Color Legend

Black: Applies to content for the NA015 baseline that is valid through the current release.

Red: Applies to new or modified content for NA017 that is valid through the current release.

Blue: Applies to new or modified content for NA018 (SN05 DMS) that is valid through the current release.

Green: Applies to new or modified content for SN06 (DMS) that is valid through the current release.

Attention!
Adobe Acrobat Reader 5.0 is required to view bookmarks in color.

Publication History

May 2003

For the SN06 (DMS) release, 10.01, updates were made according to CR Q00534114.

297-2621-320

Digital Switching Systems

UCS DMS-250

Billing Server Application Guide

UCS15 Standard 09.01 May 2001

How the world shares ideas.

Digital Switching Systems

UCS DMS-250

Billing Server Application Guide

Publication number: 297-2621-320
Product release: UCS15
Document release: Standard 09.01
Date: May 2001

Printed in the United States of America

Publication history

May 2001

Standard release 09.02 for software release UCS15. Add chapter “FP Disable procedure.”

August 2000

Standard release 07.02 for software release UCS13. Removed the column ”File size KB” from MAP response example in the ”Determine FTAM billing transfer” section of chapter ”Teleprocessing.”

March 2000

Preliminary release 07.01 for software release UCS13. Removed reference to FP1 in the DRM commands appendix (DAT and MOUNT commands). Updated ordering infomation.

November 1998

Standard release 06.02 for software release UCS09. Replaced detailed FCDRSRCH information in Appendix C with a general list of commands, and references to appropriate documents.

October 1998

Preliminary release 06.01 for software release UCS09.

August 1998

Standard release 05.03 for software release UCS08. This document was up-issued to revise the FCDRSRCH commands and to add minor technical changes.

Contents

About this document

When to use this document

This document describes Billing Server hardware and software and provides detailed information for understanding, provisioning, and maintaining your Billing Server.

This publication is written under the following assumptions:

- The customer's switch is already installed, commissioned, and active.
- The personnel using this publication are trained in the use of the Table Editor and have completed Nortel Networks approved datafill and maintenance training.

How to check the version and issue of this document

The version and issue of the document are indicated by numbers, for example, 01.01.

The first two digits indicate the version. The version number increases each time the document is updated to support a new software release. For example, the first release of a document is 01.01. In the *next* software release cycle, the first release of the same document is 02.01.

The second two digits indicate the issue. The issue number increases each time the document is revised but rereleased in the *same* software release cycle. For example, the second release of a document in the same software release cycle is 01.02.

To determine which version of this document applies to the software in your office and how documentation for your product is organized, check the release information in the *DMS-10 and DMS-100 Family Product Documentation Directory*, 297-8991-001.

This document is written for all DMS-100 Family offices. More than one version of this document may exist. To determine whether you have the latest version of this document and how documentation for your product is organized, check the release information in the *DMS-10 and DMS-100 Family Product Documentation Directory*, 297-8991-001.

How this document is organized

The chapters in this document provide the following:

Chapter 1, Understanding Billing Server

Chapter 1 contains an overview of Billing Server.

Chapter 2, Billing Server hardware

Chapter 2 provides detailed Billing Server hardware information, including circuit pack layout examples.

Chapter 3, Billing Server software

Chapter 3 provides an overview of Billing Server software (discussed in more detail in Chapters 4 through 8).

Chapter 4, Billing Server hardware datafill

Chapter 4 provides procedures to datafill Billing Server hardware.

Chapter 5, Billing Server SCSI device management

Chapter 5 provides detailed information about SCSI disk management using DRM, DISKADM, DISKUT, and SHADOWUT. Procedures for disk management are included.

Chapter 6, Teleprocessing

Chapter 6 describes the protocols used and procedures required to transport data to a downstream processor.

Chapter 7, SBS menu

Chapter 7 describes the SBS menu required to activate and deactivate Billing Server.

Chapter 8, FCDRSRCH menu

Chapter 8 describes the call detail record search function.

Appendix A, User interface quick reference guide

Appendix A provides a shortened version of all procedures described in Chapters 1 through 8.

Appendix B, Billing Server data schema

Appendix B provides detailed Billing Server-specific data schema information.

Appendix C, FCDRSRCH menu commands

Appendix C provides general information on the FCDRSRCH commands.

Appendix D, DRM commands

Appendix D provides a quick reference for the DRM commands.

Appendix E, LOGUTIL commands

Appendix E provides a quick reference for the LOGUTIL commands.

References in this document

The following documents are referred to in this document:

UCS DMS-250 related documentation

The following UCS DMS-250 documents are referred to in this document:

Title	Number
Device Independent Recording Package (DIRP) Administration Guide	297-1001-345
Product Documentation Directory	297-8991-001
DMS-100 Customer Data Schema Reference Manual	297-8001-351
UCS DMS-250 Billing Records Application Guide	297-2621-395
UCS DMS-250 Commands Reference Manual	297-2621-819
UCS DMS-250 Data Schema Reference Manual	297-2621-851
UCS DMS-250 Log Reports Reference Manual	297-2621-840
UCS DMS-250 Operational Measurements Reference Manual	297-2621-814
UCS DMS-250 Master Index of Publications	297-2621-001

File processor (FP) related documentation

The following FP documents are referred to in this document:

Title	Number
UCS DMS-250 Data Schema Reference Manual	297-2621-851
UCS DMS-250 Log Reports Reference Manual	297-2621-840
UCS DMS-250 Office Parameters Reference Manual	297-2621-855
UCS DMS-250 Operational Measurements Reference Manual	297-2621-814

Fiberized link interface (FLIS) and Ethernet internet unit (EIU) related documentation

The following FLIS amd EIU documents are referred to in this document:

Title	Number
Routine Maintenance Procedures	297-2621-546
Menu Commands Reference Manual	297-1001-821
Peripheral Modules Maintenance Guide	297-1001-592
UCS DMS-250 Data Schema Reference Manual	297-2621-851
UCS DMS-250 Log Reports Reference Manual	297-2621-840
UCS DMS-250 Operational Measurements Reference Manual	297-2621-814

Logs and Operational measurement (OM) related design documents

The following Logs and OM documents are referred to in this document:

Title	Number
DRM Move and DRM User Information Area	AD4333
FTAM File Transfer Service (I) for Billing	AD3500
S/DMS DRM	AD3363
Storage Agent/Collector for Billing System	AD3637

What precautionary messages mean

The types of precautionary messages used in Nortel Networks documents include attention boxes and danger, warning, and caution messages.

An attention box identifies information that is necessary for the proper performance of a procedure or task or the correct interpretation of information or data. Danger, warning, and caution messages indicate possible risks.

Examples of the precautionary messages follow.

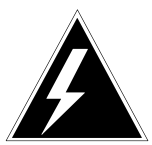
ATTENTION Information needed to perform a task

> **ATTENTION**
> If the unused DS-3 ports are not deprovisioned before a DS-1/VT Mapper is installed, the DS-1 traffic will not be carried through the DS-1/VT Mapper, even though the DS-1/VT Mapper is properly provisioned.

DANGER Possibility of personal injury

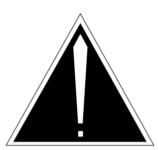
> **DANGER**
> **Risk of electrocution**
> Do not open the front panel of the inverter unless fuses F1, F2, and F3 have been removed. The inverter contains high-voltage lines. Until the fuses are removed, the high-voltage lines are active, and you risk being electrocuted.

WARNING Possibility of equipment damage



> **WARNING**
> **Damage to the backplane connector pins**
> Align the card before seating it, to avoid bending the backplane connector pins. Use light thumb pressure to align the card with the connectors. Next, use the levers on the card to seat the card into the connectors.

CAUTION Possibility of service interruption or degradation



> **CAUTION**
> **Possible loss of service**
> Before continuing, confirm that you are removing the card from the inactive unit of the peripheral module. Subscriber service will be lost if you remove a card from the active unit.

How commands, parameters, and responses are represented

Commands, parameters, and responses in this document conform to the following conventions.

Input prompt (>)

An input prompt (>) indicates that the information that follows is a command:

>BSY

Commands and fixed parameters

Commands and fixed parameters that are entered at a MAP terminal are shown in uppercase letters:

>BSY CTRL

Variables

Variables are shown in lowercase letters:

>BSY CTRL ctrl_no

The letters or numbers that the variable represents must be entered. Each variable is explained in a list that follows the command string.

Responses

Responses correspond to the MAP display and are shown in a different type:

```
FP 3 Busy CTRL 0: Command request has been submitted.
FP 3 Busy CTRL 0: Command passed.
```

The following excerpt from a procedure shows the command syntax used in this document:

1 Manually busy the CTRL on the inactive plane by typing

>BSY CTRL ctrl_no
and pressing the Enter key.

where

ctrl_no is the number of the CTRL (0 or 1)

Example of a MAP response:

```
FP 3 Busy CTRL 0: Command request has been submitted.
FP 3 Busy CTRL 0: Command passed.
```

Understanding Billing Server

Welcome to Billing Server. Billing Server is an optional set of hardware and software that interfaces with the computing module (CM). If you choose to add the Billing Server to your network, you can relieve the CM of billing data formatting and storage responsibilities.

What is Billing Server?

By adding the Billing Server to your network, the processing and storage of call detail record (CDR), operator services record (OSR), and event records are moved from the CM to the Billing Server. Operational measurements can also be stored on the Billing Server.

Benefits

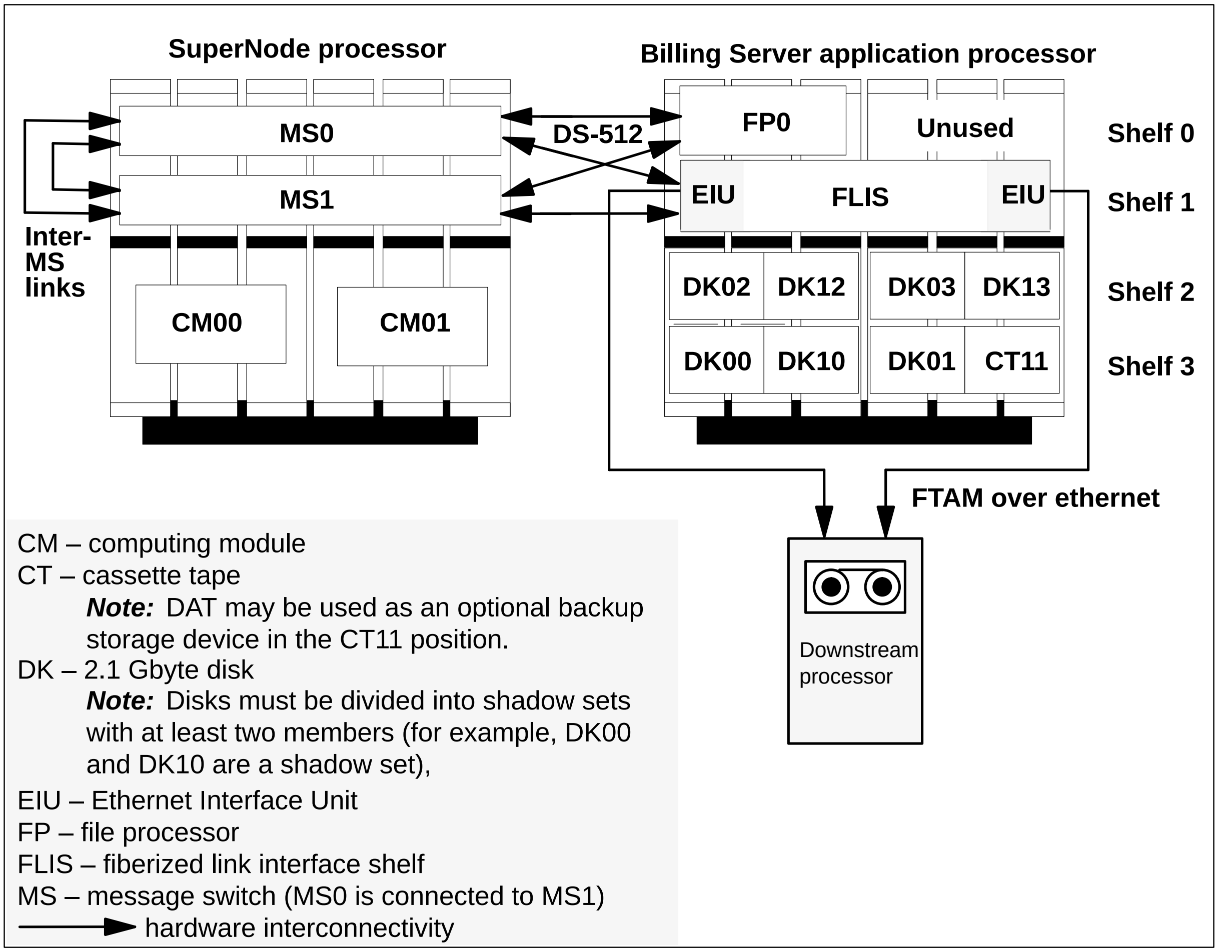
Billing Server provides the following benefits to your UCS DMS-250 switch:

- increased billing capacity with ability to process 800,000 billing records (CDRs and OSRs) per hour
- increased storage capacity for CDRs, OSRs, and OMs
- high-speed reliable access to CDRs, OSRs, and OMs available at the downstream processor through File Transfer, Access, and Management (FTAM)
- flexible modular architecture to accommodate future billing requirements effectively and economically
- a base platform capable of storing billing records (CDRs, OSRs, and event records) and OMs
- disk shadowing
- fault tolerance
- ability to back up files to core input/output controller (IOC) 9-track tape
- ability to back up shadowed disks to a digital audio tape (DAT) located in the application processor cabinet (APC) on the device storage shelf
- multiple billing stream functionality

- relieves UCS DMS-Core of formatting responsibilities creating real-time savings to the call processor
- supports 20 simultaneous searches for CDRs residing on the file processor (using FCDRSRCH)
- reverts to computing module (CM)-based billing (device independent recording package [DIRP] or equivalent) in the event of Billing Server failure
- supports the flexible billing framework (For more information, refer to the *UCS DMS-250 Billing Records Application Guide*.)

Billing Server uses the SuperNode multicomputing base platform. The Billing Server relieves the UCS DMS-Core by formatting billing data and providing storage capabilities and transfer protocols. Figure 1-1 is a high-level representation of the UCS DMS-Core, Billing Server, and downstream processor interconnectivity.

Figure 1-1
UCS DMS-Core, Billing Server, and downstream processor connectivity



Dependencies

Billing Server depends on the following:

- DMS-Core hardware
  - — message switch (MS)
  - — inter-MS links
- Application Processor Cabinet (APC) hardware
  - — file processor (FP)
  - — Fiberized Link Interface Shelf (FLIS)
    - – Ethernet Interface Unit (EIU) with an 8 Mbyte micro-controller subsystem (supports FTAM)
- EIU software to support FTAM

- Small Computer Standard Interface (SCSI) devices
  - 2.1 Gbyte disk drives
  - 1.3 Mbyte DAT
- FP software/applications
  - CDR/OSR formatter
  - DISKADM and DISKUT for maintenance and configuration
  - table SHADOW and SHADOWUT for disk shadowing
  - FTAM for:
    - file integrity on FP restart
    - cache access to most recently written billing
    - high reliability
- DMS-Core software to support FP and EIU

Data transfer

The following steps show how data is handled and stored by Billing Server:

1. CDRs, OSRs, and OMs are collected on the core as calls are made.
2. When a call is completed, billing data is sent from the CM to Billing Server through the MS. When an OM dump is made, OMs are delivered to Billing Server.
3. The data is then transferred through the Billing Server in a user-defined stream. This stream is defined in tables SBSMAP and SBSFMT.
4. CDRs and OSRs are formatted by the UCS DMS-250 billing formatters.

   Note: Refer to the *UCS DMS-250 Billing Records Application Guide* for more information on the formatters.

5. The formatted CDRs and OSRs are passed to DRM and placed into billing volumes created by the user. OMs are sent directly to distributed recording manager (DRM) and placed in OM volumes.

   Note: OMs are not formatted by Billing Server.

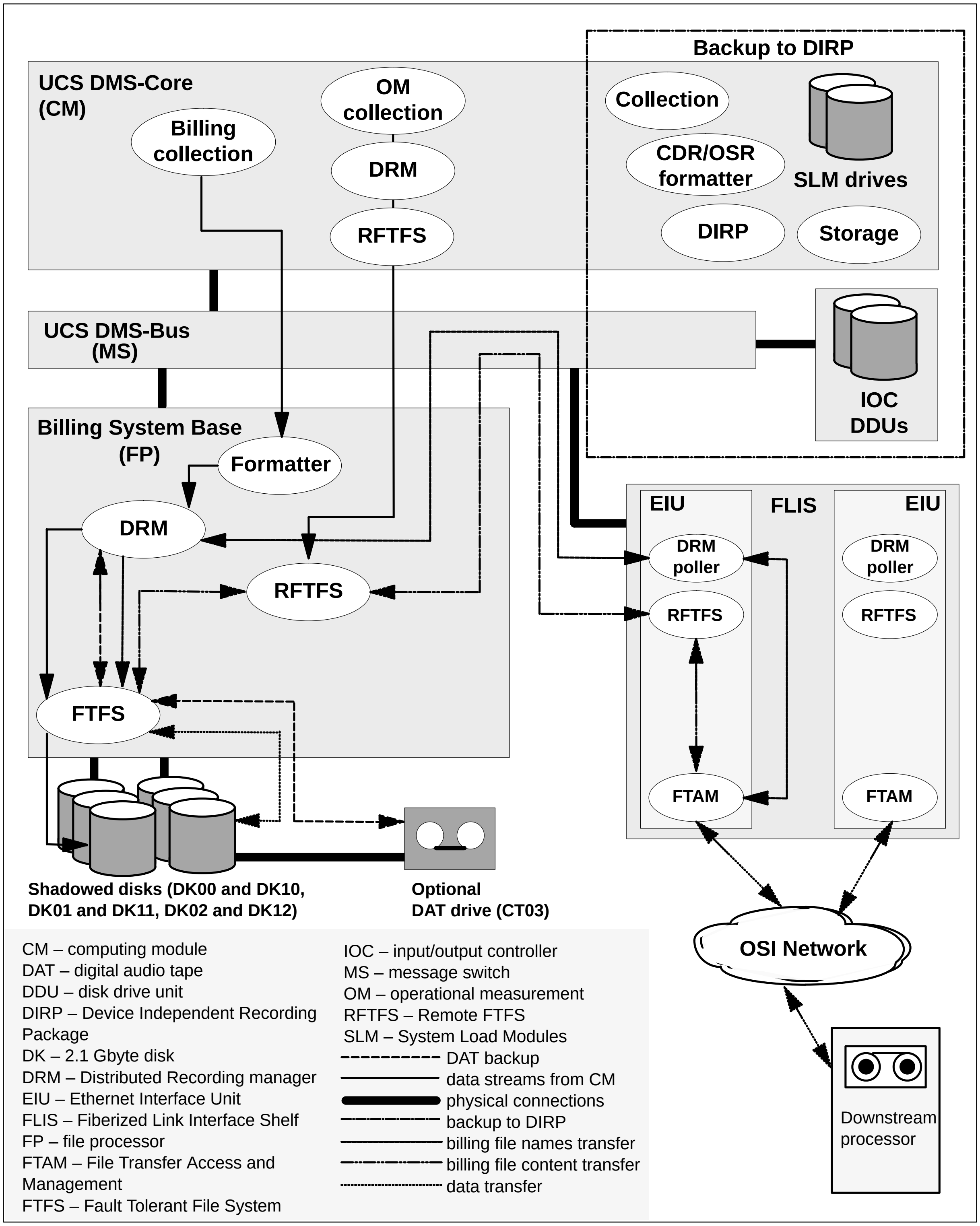
6. DRM manages and creates files to hold the data.
7. Each file holds a user-defined number of 2K blocks of data.
8. Each block holds the individual records.

Figure 1-2 is a high-level representation of the interaction between Billing Server hardware and software.

Feature Interactions

Billing Server supports the UCS DMS-250 EADAS interface. Refer to the *UCS07 Software Release Document*, feature AD9651, for more information.

Figure 1-2
Billing Server hardware and software interaction



Limitations and restrictions

The following are Billing Server limitations and restrictions:

- A single FLIS-equipped FP supports up to eight SCSI devices (disk drives or DAT).
- Billing Server does not use (other than core IOC 9-track tape) devices that belong to the IOC or the System Load Module (SLM) file systems.
- The use of the shadowed disk feature of Fault Tolerant File System (FTFS) is highly recommended.
- Billing Server is dedicated to FTAM.

  Note: Transmission Control Protocol/Internet Protocol (TCP/IP), Telnet, Network File System (NFS), and File Transfer Protocol (FTP) are not supported for Billing Server.

- Only one FP (FP0) per UCS DMS-250 switch is supported.
- The CM-based DIRP serves as the backup billing system.
- The flexible billing framework is supported. Refer to the *UCS DMS-250 Billing Records Application Guide* for more information.
- A maximum of four logical FTAM connections per EIU are supported.
- FTAM supports a maximum of two EIUs.
- A maximum of sixteen stream names are supported.
- Only one collector is supported and must reside on the CM.
- Billing records are not written to disk on a per-call basis. They are bundled in the disk buffer and the group of records is written to disk when the disk buffer is full or when the billing file is rotated. During periods of low traffic, a perceptible delay may occur between the time the call is disconnected and when the record is written to the disk.
- Records are not available to the downstream processor until a block of data is filled.
- Billing Server does not support data volumes on the CM.

Billing Server hardware

Billing Server hardware resides on the DMS-Core and in an Application Processor Cabinet (APC).

DMS-Core hardware

The DMS-Core transports billing data (call detail records [CDR], operator services records [OSR], and event records) and operational measurements (OM) through the DMS-Bus (also known as the message switch [MS]) to the Billing Server. The MS also performs the routing of messages between Billing Server and the downstream processor.

Message switch

The MS is the DMS-250 SuperNode messaging component which transfers data in the UCS DMS-250 system. The message switch is fully redundant with the two units (MS0 and MS1) operating in a load-sharing mode.

The MS handles all data transferred from the computing module (CM) to the file processor (FP) and all data requested from the downstream processor.

Table 2-1 shows the MS hardware components specific to Billing Server, the respective Product Engineering Codes (PEC), the physical slot locations (user definable), and the quantity required for operation. Figure 2-1 shows the MS circuit pack layout. (Required hardware is shaded.)

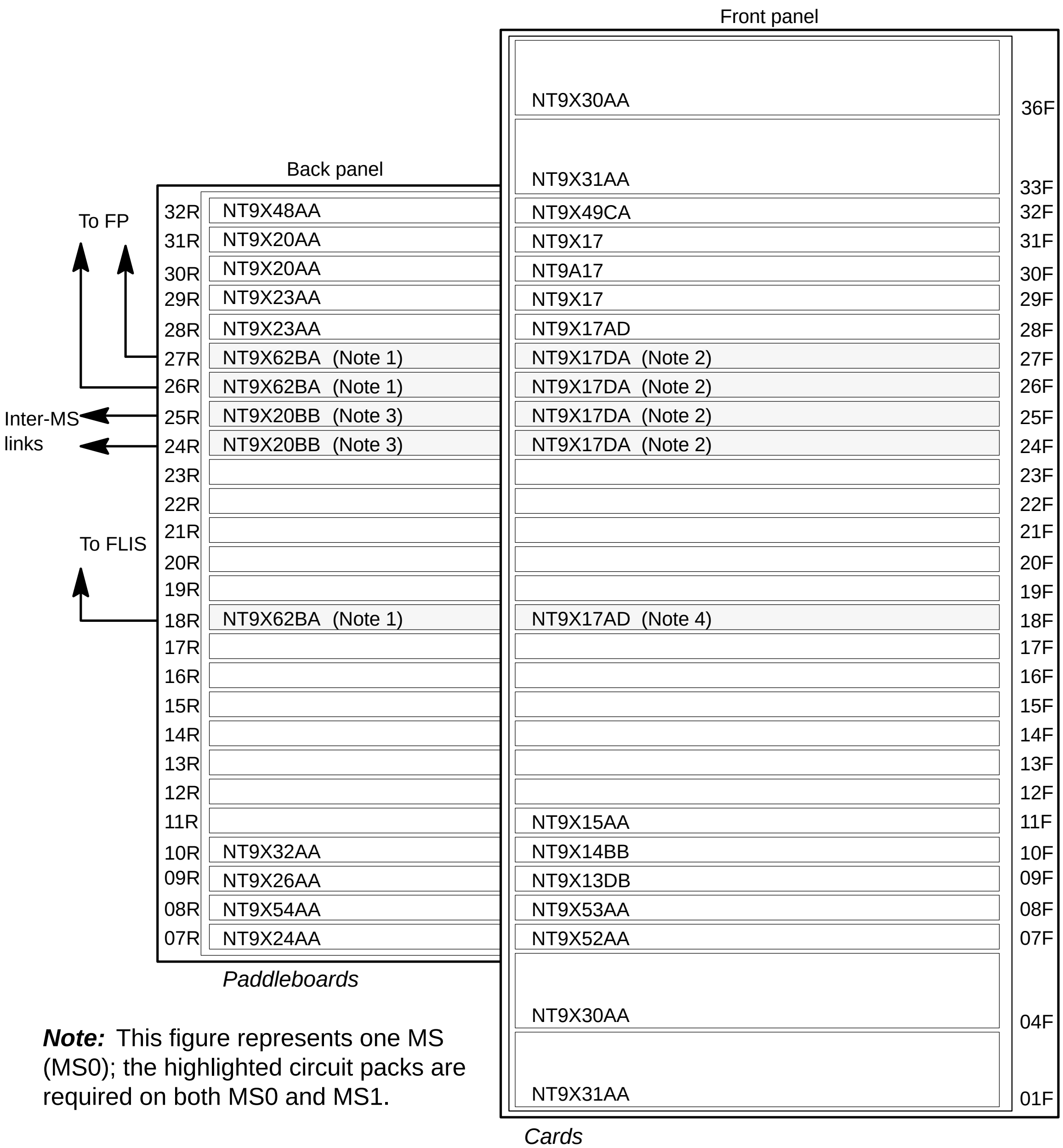
The slot assignments shown in Table 2-1 and Figure 2-1 are examples only; actual slot assignments are engineered to support specific site needs.

Table 2-1
MS0 hardware requirements specific to Billing Server

MS hardware	PEC	Physical slot examples	Quantity
MS 4 port card circuit pack	NT9X17AD	18F	1
DMS-Bus 64 port circuit pack	NT9X17DA	24F, 25F, 26F, and 27F	4
FLIS links			
4-port subrate DS512 paddleboard	NT9X62BA	18R	1
FP links			
4-port subrate DS512 paddleboard	NT9X62BA	26R and 27R	2
Inter-MS links			
ENET/MS fiber interface paddleboard	NT9X20BB	24R and 25R	2
Inter-MS link dual fiber cable	NT0X97AE	not applicable	2

Note 1: The hardware listed here is for MS0 only; MS1 requires the same hardware.
Note 2: The slot assignments shown in the table are examples only; actual slot assignments are engineered on a per-site basis.
Note 3: All empty slots contain filler packs.

Figure 2-1
MS0 circuit pack layout



Note: This figure represents one MS (MS0); the highlighted circuit packs are required on both MS0 and MS1.

FP – file processor
MS – message switch
FLIS – Fiberized Link Interface Shelf

Note 1: 4-port subrate DS512 paddleboard (MS)
Note 2: DMS-Bus 64-port circuit pack
Note 3: ENET/MS fiber interface paddleboard
Note 4: MS 4-port card circuit pack

MS links

Billing Server requires the following message switch (MS) links (a link represents a fiber pair):

- MS to FP (four links: MS0 to PL0, MS0 to PL1, MS1 to PL0, MS1 to PL1); see Figure 2-2. For information on FP hardware, refer to “File Processor shelf” in this chapter.
- MS to FLIS (two links: MS0 to one EIU, MS1 to other EIU); see Figure 2-3. For information on Fiberized Link Interface Shelf (FLIS) hardware, refer to “Fiberized Link Interface Shelf” in this chapter.
- Inter-MS links (linking MS0 and MS1)

Figure 2-2
MS-FP fiber links

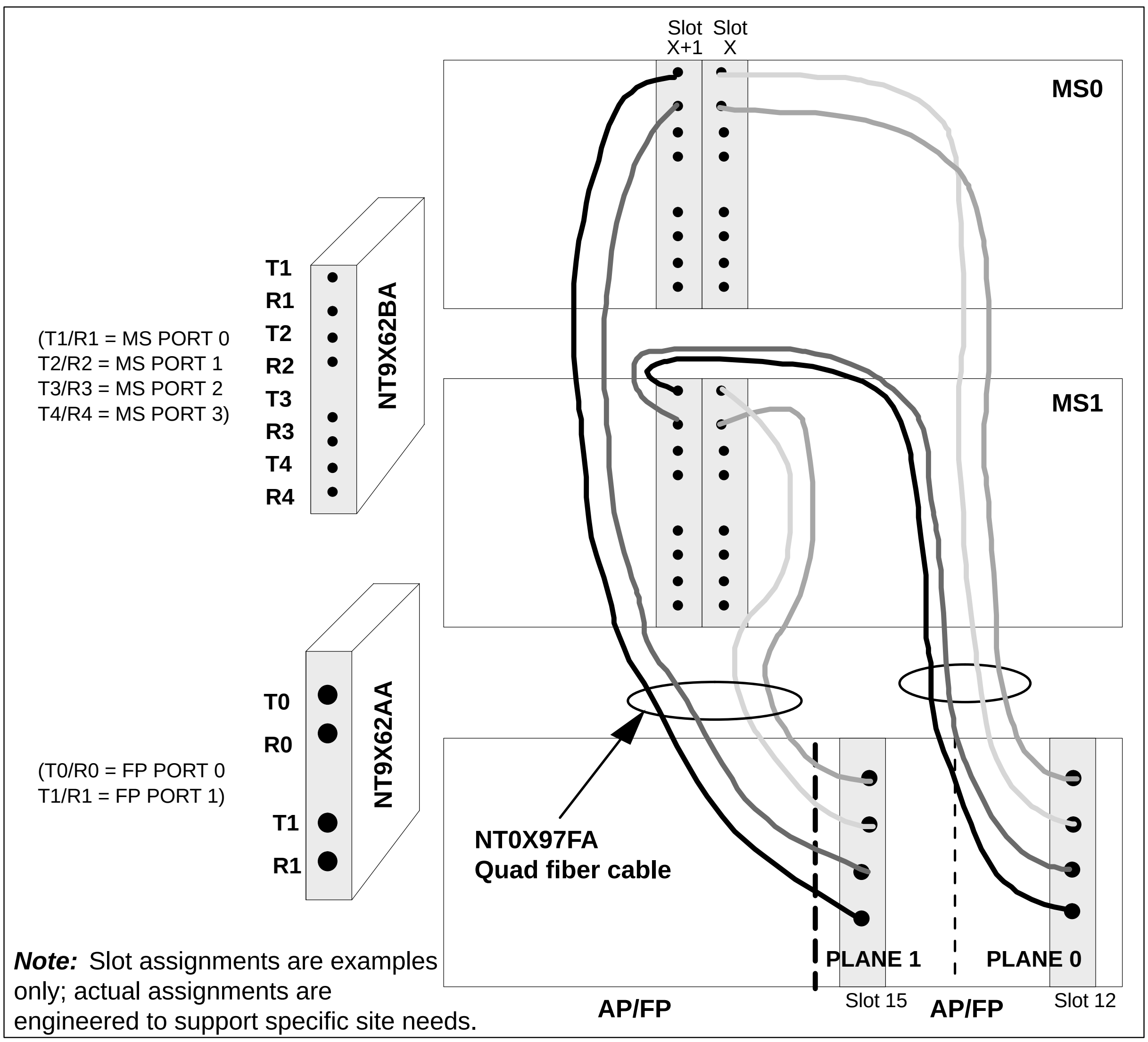
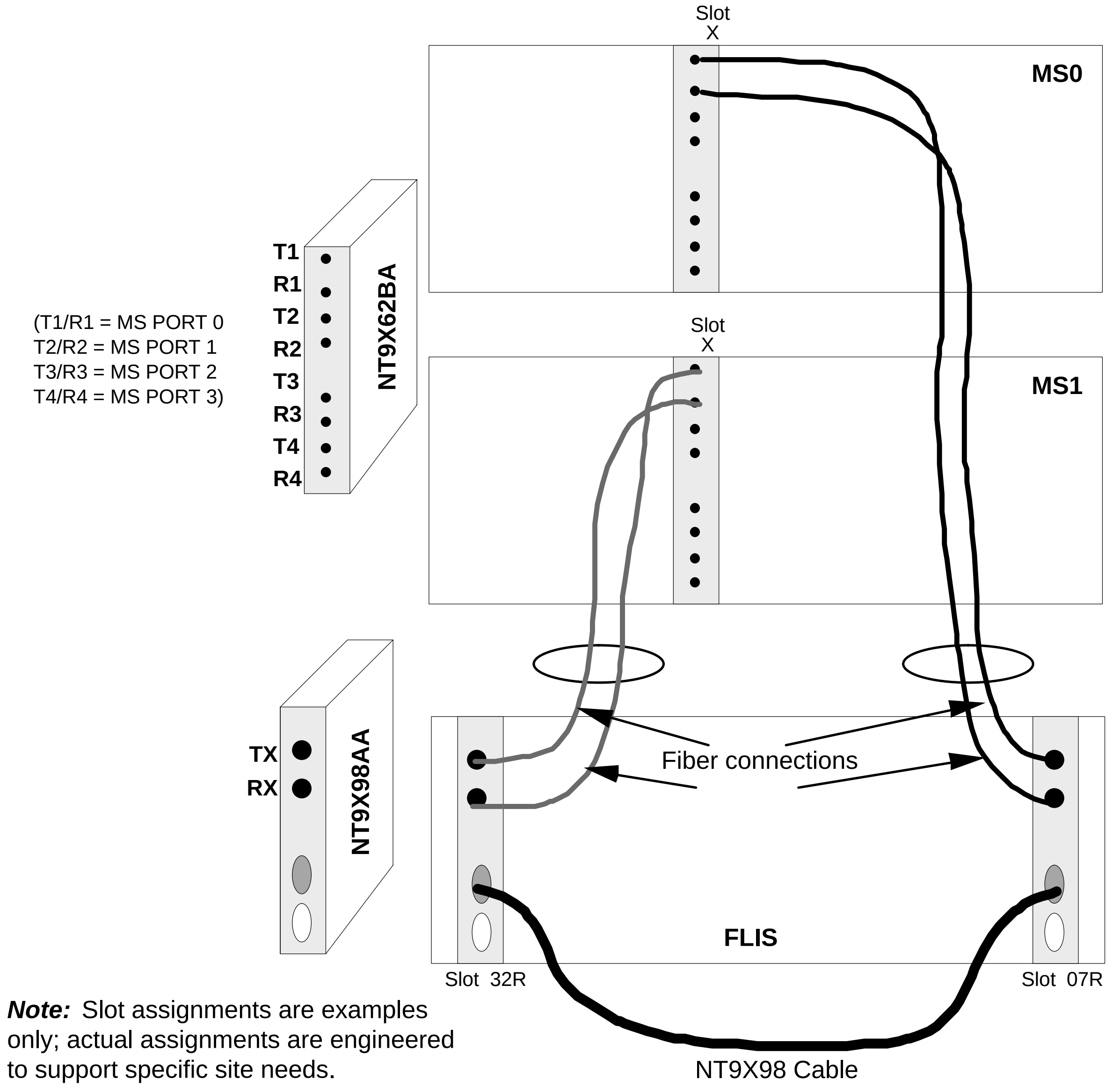


Figure 2-3
MS-FLIS fiber links

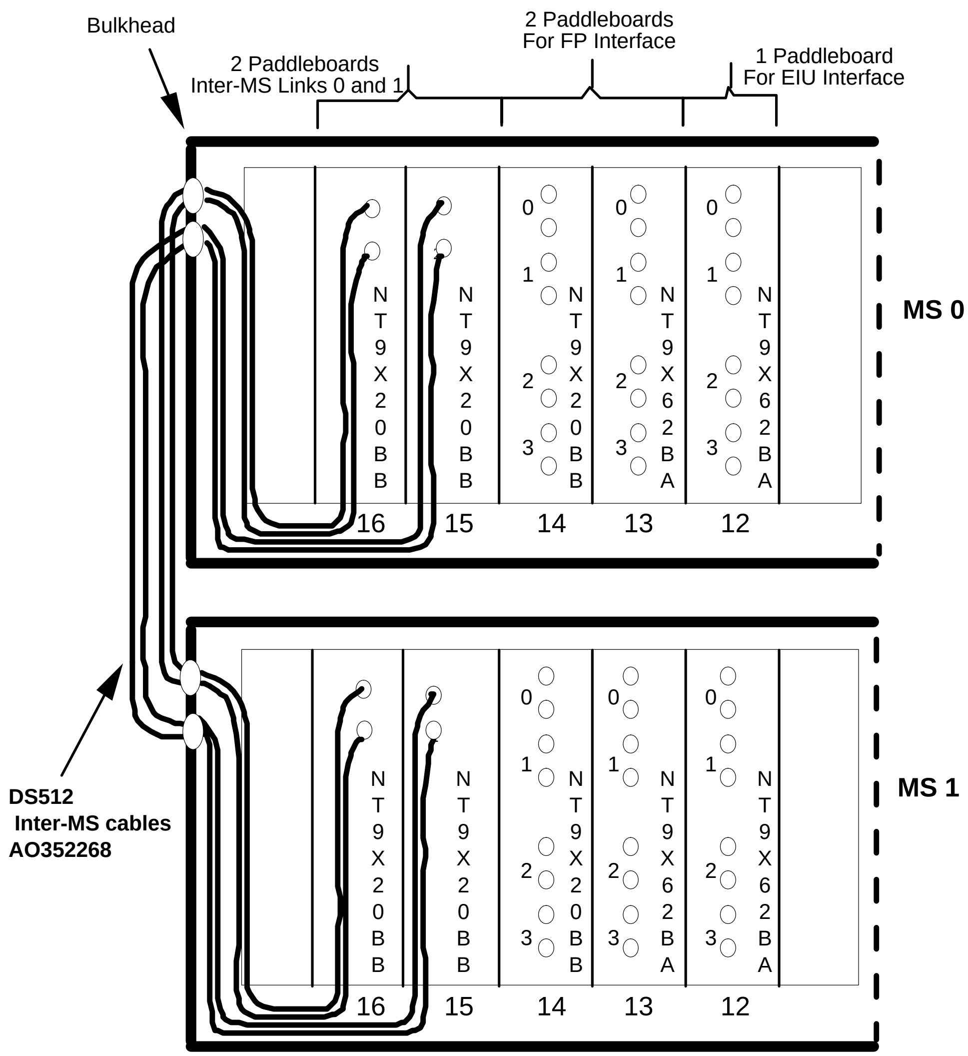


Note: Slot assignments are examples only; actual assignments are engineered to support specific site needs.

Inter-MS links

Billing Server requires inter-MS links. Using inter-MS links improves the reliability of the UCS DMS-Bus. These links require the provisioning of two ports for each MS plane (MS0 and MS1) to be interconnected. Interconnecting MS0 and MS1 creates an emergency path for rerouting frame transport system messages when normal paths are blocked due to hardware failures. Figure 2-4 shows the inter-MS link cabling.

Figure 2-4
Inter-MS link cabling



Note: Slot assignments are examples only; actual assignments are engineered to support specific site needs.

Application Processor Cabinet

The Application Processor Cabinet (APC) holds the file processor (FP) shelf, Fiberized Link Interface Shelf (FLIS), and storage devices. Components in the APC communicate to the DMS-Core and the downstream processor over fiber links through the MS. Billing Server hardware is housed in an APC and is comprised of the following components:

- FP shelf
- FLIS
  — Ethernet Interface Unit (EIU)
- two device storage shelves

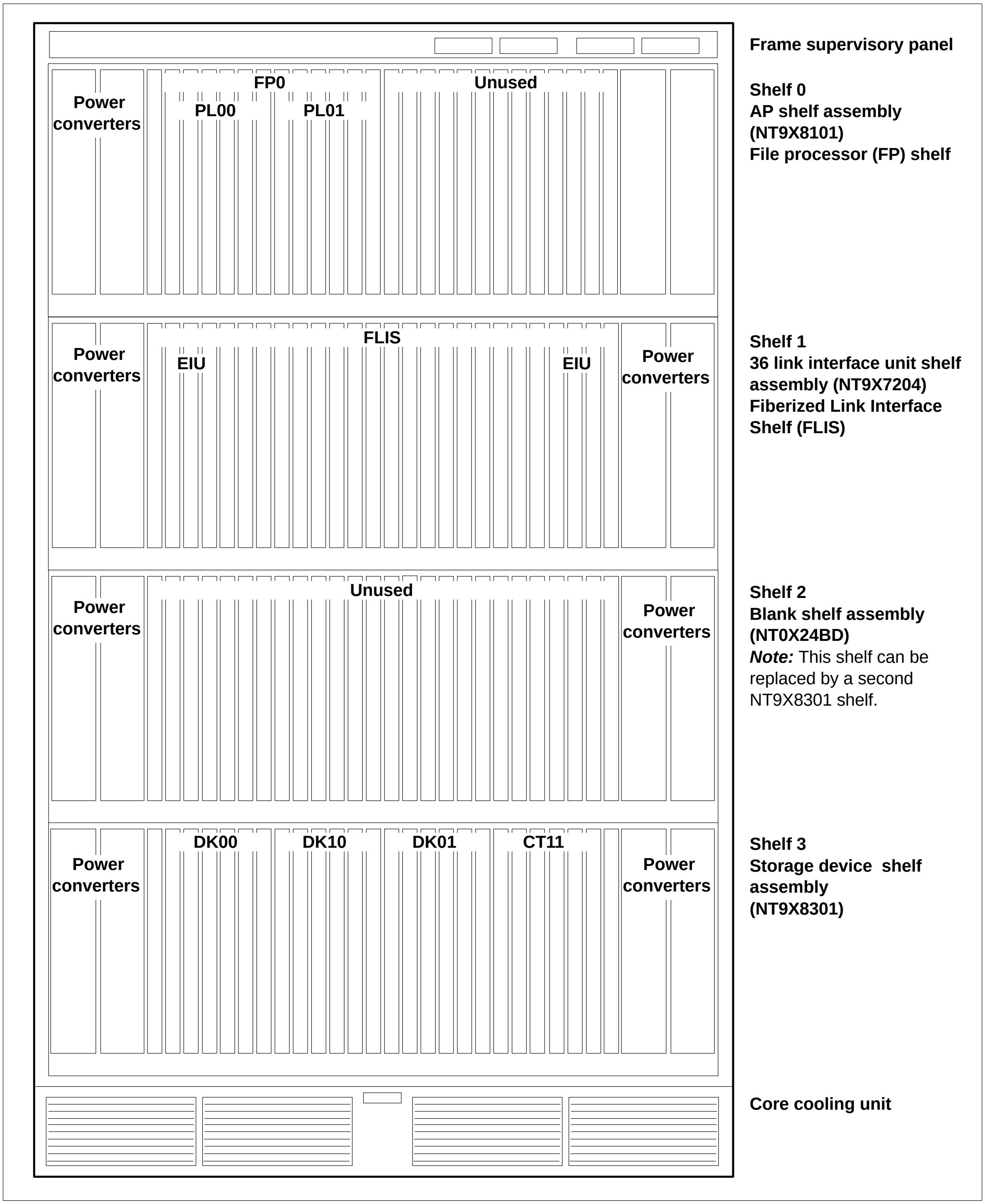
Table 2-2 shows the required FP hardware components, the respective PEC, the physical slot location, and the quantity required for operation. Figure 2-5 illustrates Billing Server configuration in the APC. The slot assignments shown in the table are examples only; actual slot assignments are engineered to meet specific site needs.

Table 2-2
APC hardware requirements

APC hardware	PEC	Shelf number	Quantity
Frame supervisory panel assembly kit (48 V)	NT9X03AA	not applicable	1
AP/FP shelf assembly (see note)	NT9X8101	0	1
36 LIU link interface unit shelf assembly	NT9X7204	1	1
Blank shelf assembly (This shelf may be replaced by NT9X8301 if required.)	NT0X24BD	2	1
Storage device shelf assembly (see note)	NT9X8301	3	1
Cooling Unit (48 V)	NT9X95CU	not applicable	1

Note: When ordering new shelves, specify the cabinet type, shelf type, and PEC.

Figure 2-5
APC configuration



File processor shelf

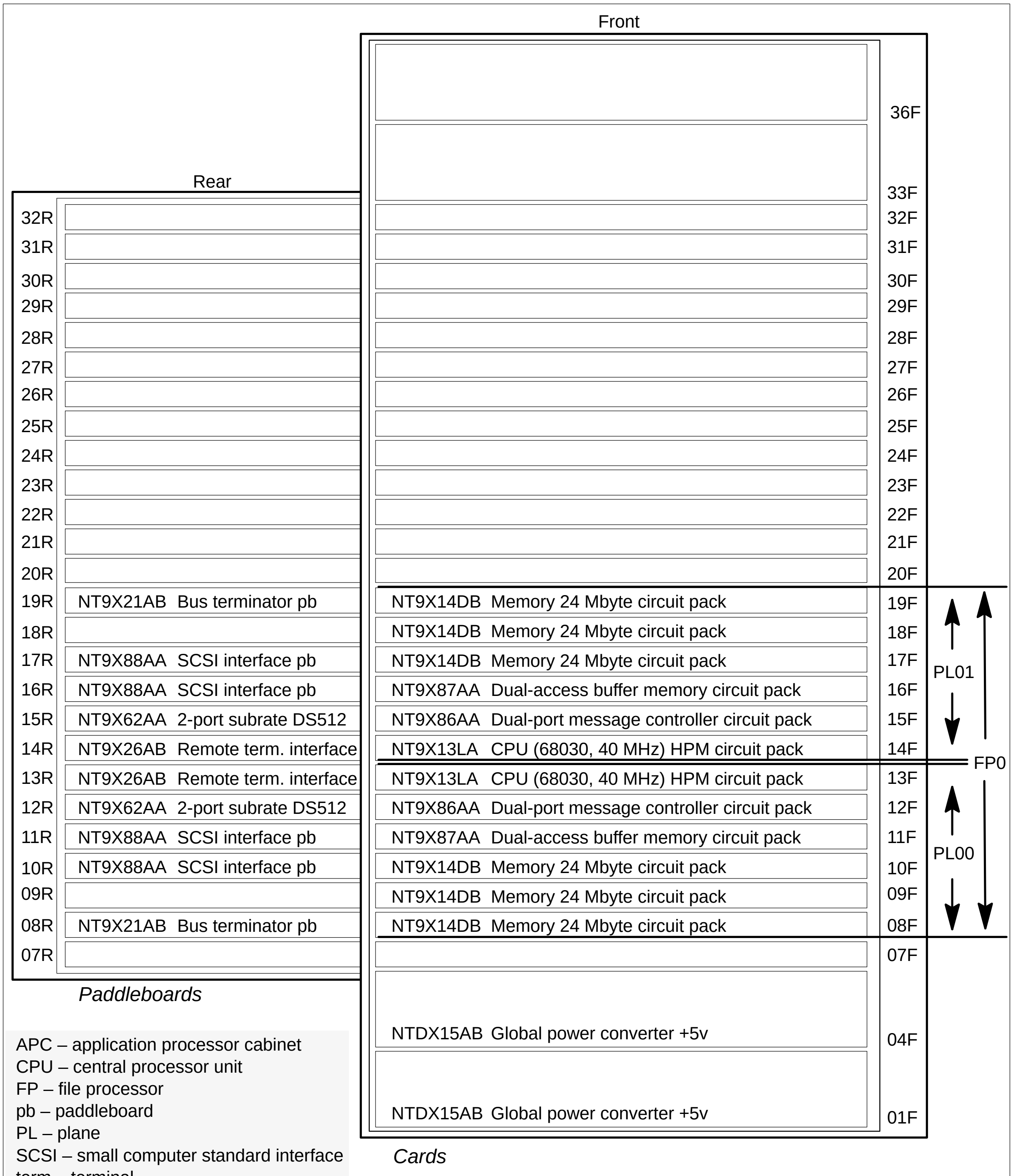
File processor (FP) hardware uses Small Computer Standard Interface (SCSI) buses, and high-reliability, high-performance SCSI technology. The FP duplicates connectivity to the devices and provides dual SCSI controllers for high performance shadow operations.

Table 2-3 shows the required FP hardware components, the respective PEC, the physical slot location, and the quantity required for operation. FP circuit pack layout is shown in Figure 2-6.

Table 2-3
FP hardware requirements for PL00 and PL01

FP hardware	PEC	Physical slots	Quantity
Global power converter +5v	NTDX15AB	1–3 and 4–6	2
Memory 24 Mbyte circuit pack	NT9X14DB	08F, 09F, 10F, 17F, 18F, and 19F	6
Dual-access buffer memory circuit pack	NT9X87AA	11F and 16F	2
Dual-port message controller circuit pack	NT9X86AA	12F and 15F	2
CPU (68030, 40 MHz) HPM circuit pack	NT9X13LA	13F and 14F	2
Bus terminator paddleboard	NT9X21AB	08R and 19R	2
SCSI Bus 1:			
SCSI interface paddleboard	NT9X88AA	10R and 17R	2
SCSI Bus 0:			
SCSI interface paddleboard	NT9X88AA	11R and 16R	2
MS links			
2-port subrate DS512 paddleboard	NT9X62AA	12R and 15R	2
Remote terminal interface paddleboard	NT9X26AB	13R and 14R	2
Quad fiber cable	NT0X97AF	not applicable	2

Figure 2-6
FP circuit pack layout



APC – application processor cabinet
CPU – central processor unit
FP – file processor
pb – paddleboard
PL – plane
SCSI – small computer standard interface
term – terminal

Fiberized Link Interface Shelf

The Fiberized Link Interface Shelf (FLIS) uses the EIUs to transfer billing data to the downstream computer, when requested by the downstream computer. FLIS is also known as a Single Shelf Link Peripheral Processor (SSLPP).

The EIUs (located on the FLIS) transfer stored data to the downstream processor. An EIU consists of three circuit packs. Data transfer does not occur until the Billing Server receives a request from the downstream processor.

Billing Server and the downstream processor are connected through an Ethernet local area network (LAN) that supports 10baseT (10 Mbit/s baseband). The LAN uses a carrier sense multiple access with collision detection (CSMA/CD) method for arbitrating access to the communication channel.

The FTAM protocol provides robust billing transfer. Each EIU supports up to four FTAM logical connections from the downstream processor. Each FLIS supports a maximum of two EIUs for Billing Server.

FLIS hardware

Table 2-4 shows the required FLIS hardware components, the respective PEC, the physical slot location, and the quantity required for operation. The FLIS circuit pack layout is shown in Figure 2-7.

Table 2-4
FLIS hardware requirements

FLIS hardware	PEC	Physical slots	Quantity
Power converters			
• –5 V power converter	NT9X31AA	1–3 and 36–38	2
• +5 V power converter	NT9X30AA	4–6 and 33–35	2
LIS fbus controller circuit pack	NT9X96AA	7F and 32F	2
Intra-Fbus B termination paddleboard	NTEX20BA	08R	1
Intra-Fbus A termination paddleboard	NTEX20AA	31R	1

Note 1: The slot assignments shown in the table are examples only; actual slot assignments are engineered to suport specific site needs.
Note 2: All empty slots contain filler packs.

—continued—

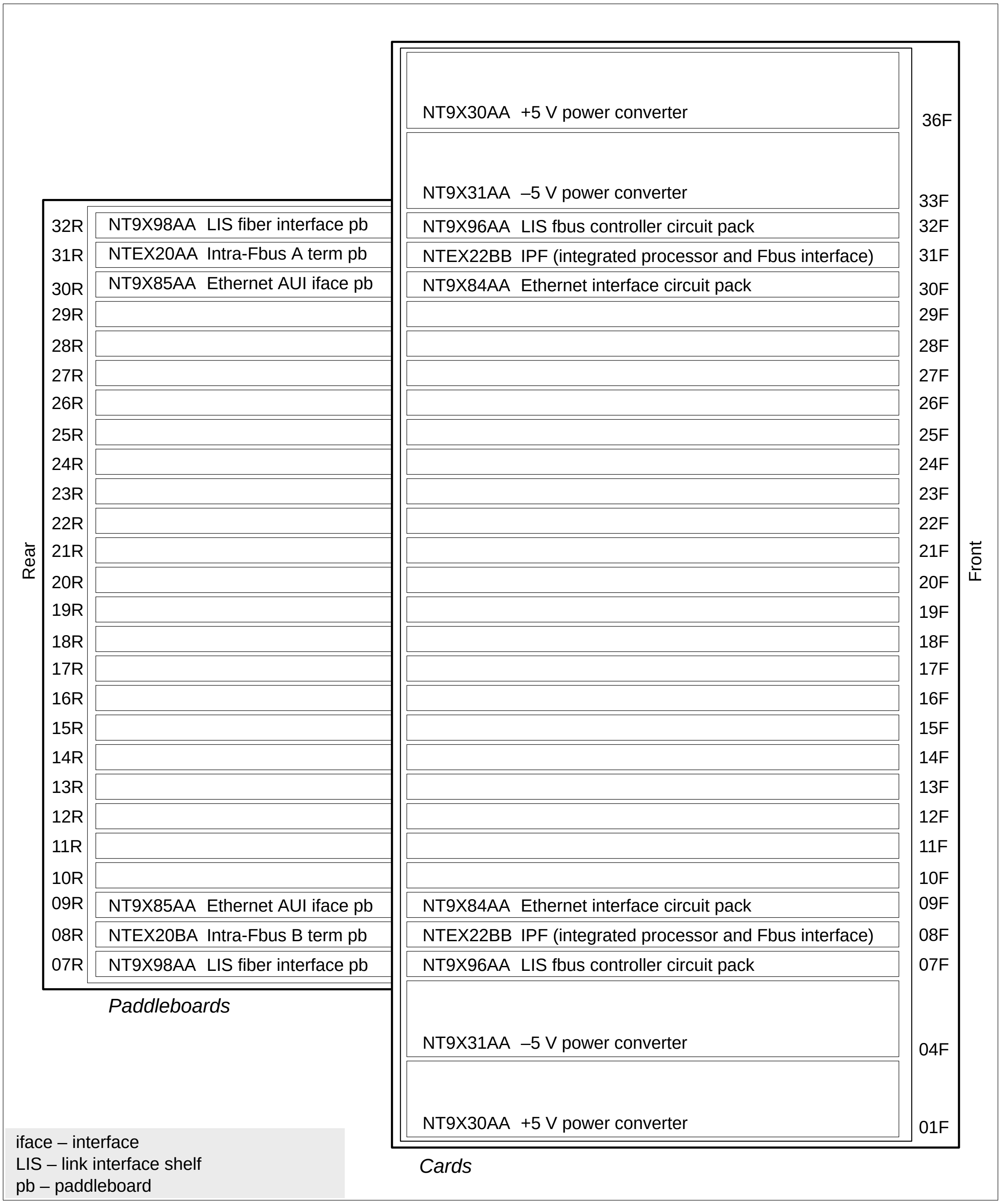
Table 2-4
FLIS hardware requirements (continued)

FLIS hardware	PEC	Physical slots	Quantity
To MS			
LIS fiber interface paddleboard	NT9X98AA	7R and 32R	2
EIUs			
• IPF (integrated processor and Fbus interface)	NTEX22BB	8F and 31F	2
• Ethernet interface circuit pack	NT9X84AA	9F and 30F	2
• To downstream processor			
— Ethernet AUI interface paddleboard	NT9X85AA	9R and 30R	2
MS–FLIS dual fiber cable	NT0X97AE	not applicable	2
Inter-EIU synchronization cable	NT9X98	not applicable	1

Note 1: The slot assignments shown in the table are examples only; actual slot assignments are engineered to suport specific site needs.
Note 2: All empty slots contain filler packs.

—end—

Figure 2-7
FLIS circuit pack layout



Storage device shelf

The storage device shelves support up to eight storage devices in a combination of the following:

- two 2.1 Gbyte formatted SCSI disks (a minimum of two disks is required, with up to eight disks supported)
- 1.3 Gbyte digital audio tape (DAT) drive which may replace one of the eight SCSI disks (only one DAT is supported)

Table 2-5 shows the required storage device shelf hardware components, the respective PEC, the physical slot location, and the quantity required for operation.

Table 2-5
Storage device shelf hardware requirements

Storage device shelf hardware	PEC	Physical slots	Quantity
+5/+12 V power converter	NT9X91AB	1–3, 4–6, 33–35, 36–38	4
Storage device assembly (2.1 Gbyte disk)	NT9X90AB	8F, 14F, 20F	3
Storage device assembly (1.3 Gbyte DAT tape)	NT9X90BA	26F	1
SCSI device interface paddleboard	NT9X89BA	8R, 9R, 14R, 15R, 20R, 21R, 26R, 27R	8
SCSI cables	NT9X04xx where, xx is cable length	not applicable	determined on a per-site basis

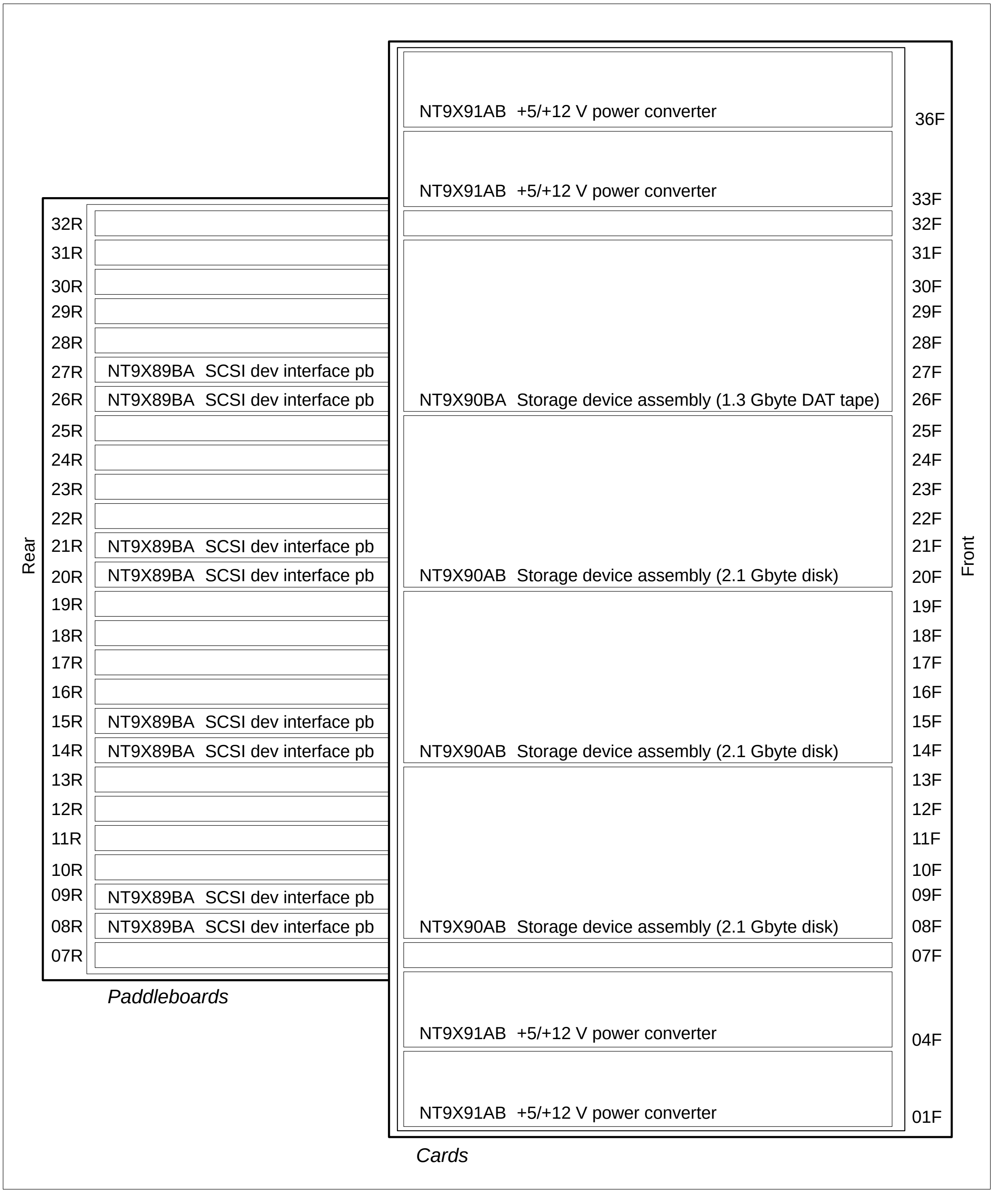
SCSI device naming conventions

SCSI devices are assigned names according to the type of device, the device number, and the SCSI bus number (for example, DK00). DK represents disk drives; CT, cassette tapes, in this case DAT. The first number in the name represents the SCSI bus number and the second number is the device number. Shadow sets may contain between two and five disk devices. The circuit pack layout is shown in Figure 2-8.

Note 1: Shadow devices should reside on different SCSI buses to ensure data and disk availability in case of a disk failure.

Note 2: Shadowing is not available for DAT.

Figure 2-8
Storage device shelf circuit pack layout



SCSI bus configuration (cabling)

The FP is connected to one or more SCSI devices by a daisy chain SCSI bus arrangement. In this arrangement, a cable is run from the SCSI port on the paddleboard at the rear of the FP to the paddleboard of the first SCSI device in the chain. See Figure 2-9.

Subsequent SCSI device paddleboards are chained together by provisionable cabling, backplane connections, or both. The last SCSI device in the chain is cabled back to the FP, terminating on the original paddleboard.

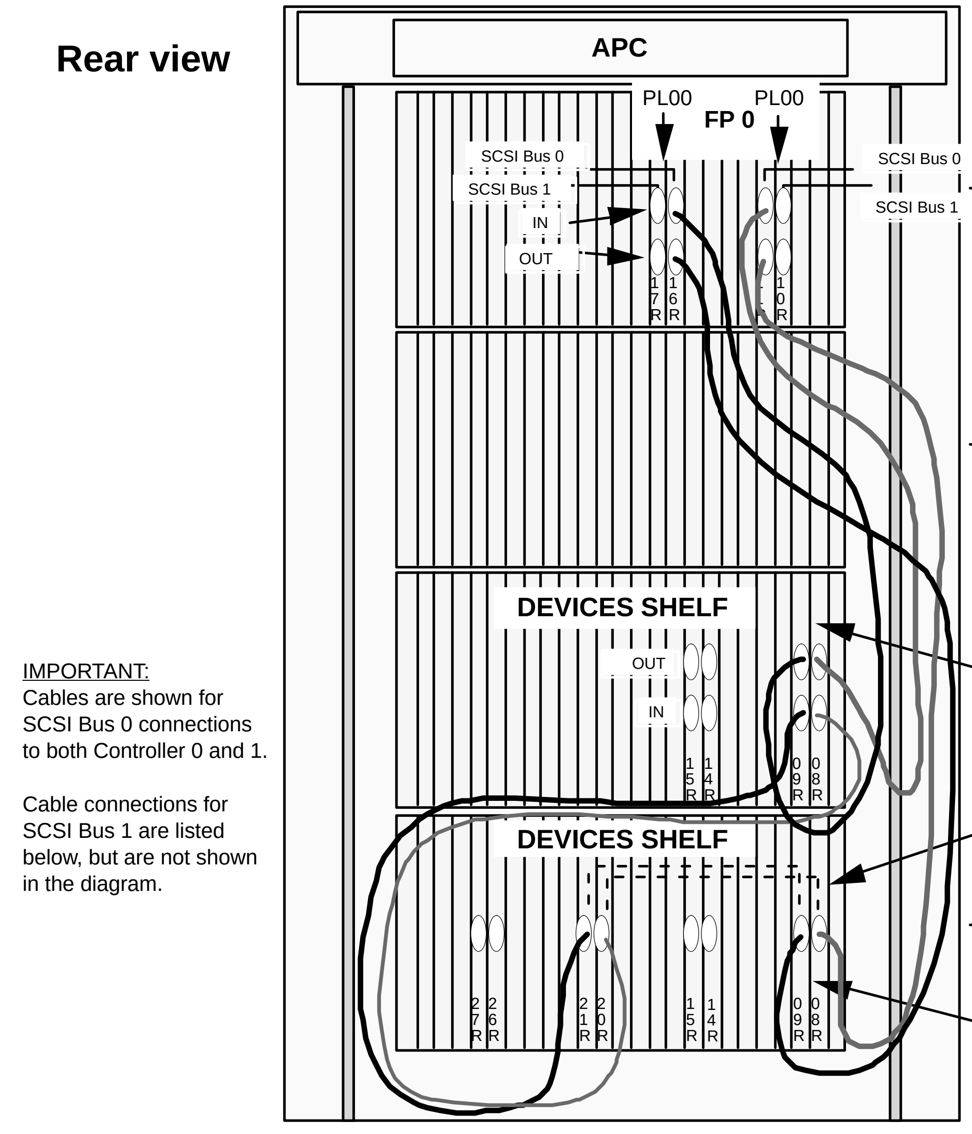
NT9X89AA

The NT9X89AA paddleboard is used for a filled device shelf. A device shelf is full when it contains four storage devices. The devices use the backplane as the SCSI bus to route data.

NT9X89BA

The NT9X89BA paddleboard is used for a partially filled device shelf. A partially filled device shelf has less than four storage devices. These devices must be physically connected by cable.

Figure 2-9
SCSI cabling



IMPORTANT:
Cables are shown for SCSI Bus 0 connections to both Controller 0 and 1.

Cable connections for SCSI Bus 1 are listed below, but are not shown in the diagram.

Example of SCSI cabling for multi-devices using both NT9X89AA and NT9X89BA paddleboards:

Controller 0 SCSI Bus 0 :
Shelf 3, Slot 08R = SDIPNO 0
Shelf 3, Slot 20R = SDIPNO 1
Shelf 2, Slot 08R = SDIPNO 2

Controller 1 SCSI Bus 0 :
Shelf 3, Slot 09R = SDIPNO 0
Shelf 3, Slot 21R = SDIPNO 1
Shelf 2, Slot 09R = SDIPNO 2

Controller 0 SCSI Bus 1 :
Shelf 3, Slot 14R = SDIPNO 0
Shelf 3, Slot 26R = SDIPNO 1
Shelf 2, Slot 14R = SDIPNO 2

Controller 1 SCSI Bus 1 :
Shelf 3, Slot 15R = SDIPNO 0
Shelf 3, Slot 27R = SDIPNO 1
Shelf 2, Slot 15R = SDIPNO 2

Billing Server software

Billing Server uses the following software components:

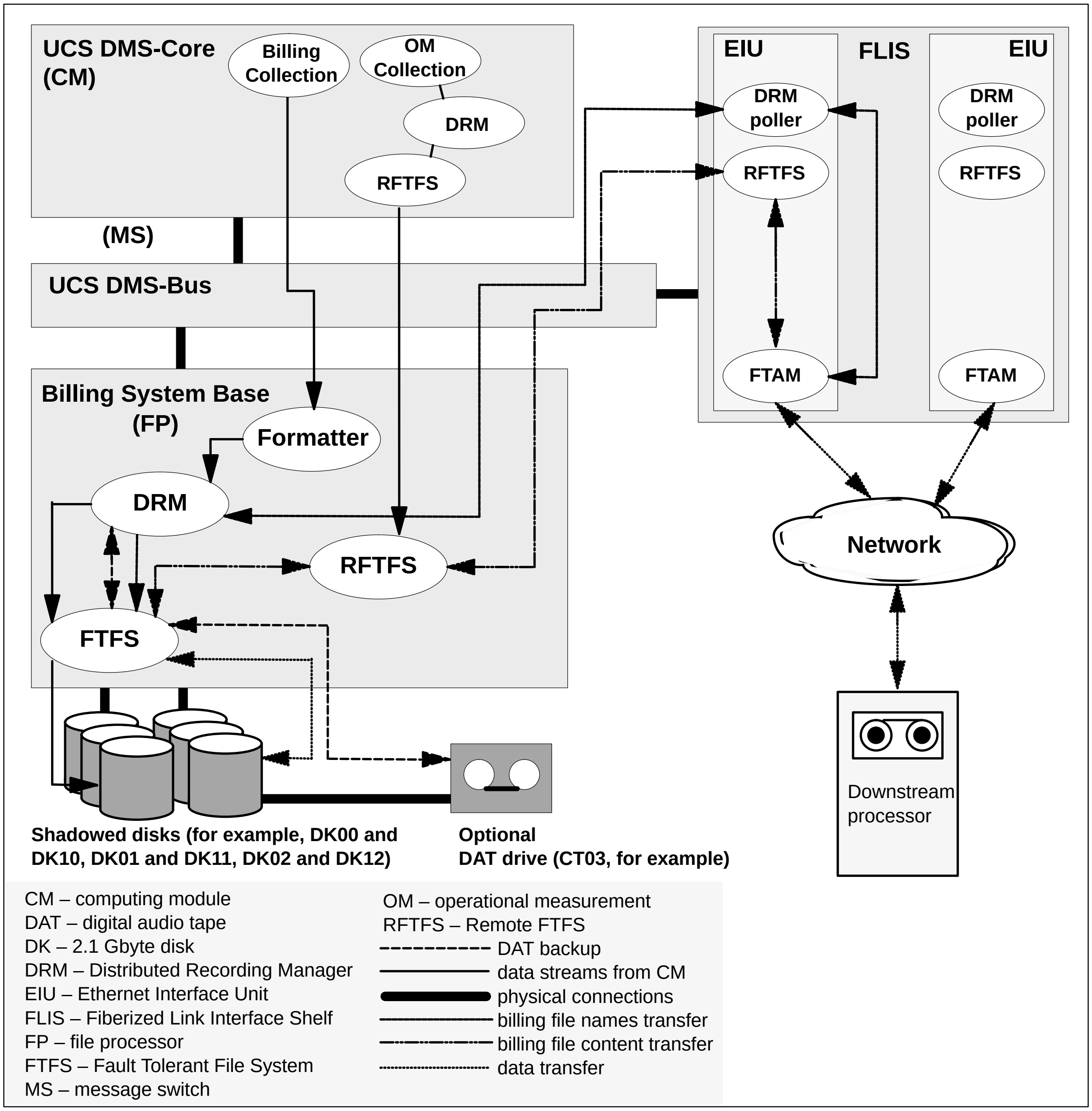
- Supernode billing system (SBS) base:
  - — collector (located on the DMS-Core)
  - — stream connection manager (SCM)
  - — formatter/storage agent (FSA)
- Distributed Recording Manager (DRM)
- Fault Tolerant File System (FTFS)
- teleprocessing through File Transfer, Access, and Management (FTAM)
- user interface

Billing Server software provides the following:

- unformatted billing transfer from the computing module to the file processor (FP)
- billing formatters infra-structure on the FP (For more information, refer to the *UCS DMS-250 Billing Records Application Guide*.)
- passage of call detail records (CDR), operator services records (OSR), and operational measurements (OM) to DRM, finally to be stored on the FP SCSI disks.

Figure 3-1 shows the interaction between the Billing Server software components.

Figure 3-1
Billing Server software components



SuperNode billing system base

The SBS provides the infrastructure to collect, format, and store billing data.

Billing collector

The collector is an interface that receives unformatted billing data as calls are completed. The collector resides on the computing module (CM).

All billing data has a call data type that determines the billing stream (datafilled in table SBSMAP). Billing Server supports the user-defined streams for billing and can handle up to sixteen streams.

These user-defined streams are used to route billing data and operational measurements to the billing server. The billing data is routed to the applicable preformatter, which adds the appropriate tags for transferring the unformatted billing data to the formatter on the FP.

Billing Server can be used to store OM data by interfacing with SuperNode Operational Measurements (S/OM). S/OM has its own commands and moves OM storage from the CM to an FP, in this case Billing Server. Refer to the *UCS DMS-250 SuperNode Operational Measurements (S/OM) Reference Manual* for further details.

OMs provide information about the performance and activities of a UCS DMS-250 switch and its peripheral components. The OM information is gathered through periodic scans of equipment components and activities.

Stream collection manager

The SCM supervises connections between collectors and FSAs on all nodes and ensures that the configuration data is distributed to other nodes. SCM decides which FSA is connected to the collector based on the following:

- priority assigned by DRM
- which FSAs are registered on SCM
- which FSAs have disk space available for the stream

The SCM resides on the DMS-Core.

Formatter/storage agent

The FSA (located on the FP) receives raw billing data from the collector through the SCM connection. The raw billing data is formatted according to the provisioned flexible billing format and placed into blocks. The blocked data is then passed to DRM to be written to disk. The number of records on each block varies depending on how the flexible billing framework is provisioned. Refer to the *UCS DMS-250 Billing Records Application Guide* for further details.

Distributed Recording Manager

The Distributed Recording Manager (DRM) manages the volumes used for the storage of CDRs, OSRs, and OMs on FTFS disks. These management functions include writing CDRs, OSRs, and OMs to the file system, keeping track of files generated, monitoring file and volume states, providing a mechanism to backup files to core input/output controller (IOC) 9-track tape or digital audio tape (DAT) on Billing Server, and space monitoring. For more information, refer to Chapter 5, “Billing Server SCSI device management.”

Fault Tolerant File System

The Fault Tolerant File System (FTFS) is used for storage and retrieval of billing data and OMs on the file processor (FP). It provides for disk shadowing and copying of files to core IOC 9-track tape. The FTFS interfaces to disk and tape.

The Remote FTFS (RFTFS) resides within the upper layer of the FTFS. It presents a consistent, seamless file system interface to an application, regardless of the application’s node location. It supports disk shadowing and a high degree of fault tolerance. RFTFS accesses files on a given node from other nodes in the switch. Refer to Chapter 5, “Billing Server SCSI device management” for more information.

Disk shadowing

Billing Server allows you to create exact copies of data onto multiple disks by providing a disk shadowing utility. You create shadow sets, consisting of a master disk and between 1 and 4 duplicate disks. These shadow sets are synchronous; each disk contains an exact copy of the data on the master disk. Disk shadowing is not required, but is recommended.

DAT

Billing Server provides base utilities to copy files to DAT and also supports copying DAT files to disk. This dual transfer capability allows you to archive data from the switch and to transport data to a downstream processor.

DAT can be formatted to the existing Nortel tape format using ASCII.

Core IOC 9-track tapes

Billing Server provides base utilities to copy files to a core IOC 9-track tape.

Teleprocessing

Billing Server uses the Open Systems Interconnection (OSI) set of protocols for transferring billing and OMs from Billing Server to the downstream processor. OSI was designed to connect unlike networks.

OSI uses File Transfer, Access, and Management (FTAM) through an Ethernet interface to provide access to files on Billing Server from a foreign system such as the downstream processor. Refer to Chapter 6, “Teleprocessing,” for more information.

User interface

User interface provides a means of manipulating Billing Server through a MAP terminal by using various utilities.

- The FCDRSRCH menu searches the CDRs and OSRs written to disk by DRM and is accessed through the following command: **>MAPCI;MTC;APPL;OAMAP;FCDRSRCH**. Refer to Chapter 8, “FCDRSRCH menu,” and Appendix C, “FCDRSRCH commands,” for more information.
- The DISKADM utility handles disk device administration. Refer to Chapter 5, “Billing Server SCSI device management,” for more information.
- The DISKUT utility maintains and manipulates the SCSI devices. Refer to Chapter 5, “Billing Server SCSI device management,” for more information.
- The DRM menu allows access to control and review functions for DRM. Use the following command to access the DRM menu: **>MAPCI;MTC;APPL;OAMAP;DRM**. Refer to Chapter 5, “Billing Server device management,” for more information.
- Table SHADOW and the SHADOWUT utility are used by the FTFS to set up disk shadowing. Refer to Chapter 5, “Billing Server SCSI device management,” for more information.
- The SBS menu is used to activate and deactivate Billing Server and is accessed through the following command: **>MAPCI;MTC;APPL;OAMAP;SBS**. Refer to Chapter 7, “SBS menu,” for more information.
- The table editor is used to datafill tables required for Billing Server operation. Table 3-1 maps the types of datafill requirements to the location of the procedure.

Table 3-1
Datafill map

Datafill type	Procedure location	Table name
Billing Server hardware datafill procedures	Chapter 4, “Billing Server hardware datafill”	APINV, APCDINV, FPDEVINV, FPDIPINV, LIUINV, MSCDINV, MSFWLOAD, MSILINV, MSINV, PMLOADS, SUSHELF
DRM-related datafill procedures	Chapter 5, “Billing Server SCSI device management”	DRMPOOL and DRMAPPL
FTAM-related datafill procedures	Chapter 6 , “Teleprocessing”	OSIPARMS and OSIROUTE
General table information	Appendix B, “Billing Server data schema”	all tables
SBS-related datafill procedures	Chapter 7, “SBS menu”	SBSFMT and SBSMAP
Shadowing datafill	Chapter 5, “Billing Server SCSI device management”	SHADOW

Billing Server hardware datafill

A UCS DMS-250 switch must contain datafill for every piece of hardware used by Billing Server. Table 4-1 shows the datafill procedures in alphabetical order.

Table 4-1
Billing Server hardware datafill procedures

Procedure	Table name	Page
Add Billing Server to DMS system, establish MS-FP fiber links	APINV	4–2
Add DMS-Core, Billing Server-specific circuit packs to the MS	MSCDINV	4–3
Add FLIS to DMS system	SUSHELF	4–5
Define the EIUs	LIUINV	4–6
Define sequence of devices along the SCSI bus	FPDIPINV	4–8
Define storage devices	FPDEVINV	4–9
Ensure the MS is datafilled	MSINV	4–10
Establish the inter-MS links	MSILINV	4–11
Verify firmware load in datafilled	MSFWLOAD	4–12
Verify FP circuit packs and FP and device shelf power converter datafill	APCDINV	4–13
Verify software load is datafilled	PMLOADS	4–15

Add Billing Server to DMS system, establish MS-FP fiber links

Note: PMLOADS must be datafilled first in table APINV.

At the CI prompt

1 From the MAPCI level, enter table APINV by typing:

>TABLE APINV
and pressing the Enter key. ↵

2 Add Billing Server to the DMS system using the following format:

>ADD smntype smnno function loadname selfload floor row frame shelf shelfpec quadrant linkrate links $
where

smntype	is the sync-matched node type (FP).
smnno	is the sync-matched node instance number.
function	is the sync-matched node function.
loadname	is the default load file name (same as file in table PMLOADS).
selfload	indicates the self-loading capability of the system (Y or N).
floor	is the floor position of the cabinet.
row	is the row position of the cabinet.
frame	is the frame position of the cabinet.
shelf	is the shelf number of the FP.
shelfpec	is the shelf PEC.
quadrant	is the quadrant range where the FP is located.
linkrate	is the is the bandwidth on the FP-MS fiber link.
links	is a multiple-entry field comprised of 5 subfields: TCARDNO, TLINKNO, DNODENO, DCARDNO, and DLINKNO. This field defines the MS-FP fiber links.

TCARDNO	is the physical slot location of the FP paddleboard.
TLINKNO	is the terminating link number (0 or 1).
DNODENO	is the MS node number (0 or 1).
DCARDNO	is the physical slot location of the MS paddleboard.
DLINKNO	is the link number (0 or 1)

Sample entry: **>ADD fp 0 sbs fcc02bq n 1 a 6 0 nt9x81aa 0 r256 12 0 0 15 0 15 0 1 15 0 12 1 1 16 0 15 1 0 16 0 $**

3 Verify the datafill by typing:

>POS FP0
and pressing the Enter key. ↵

Billing Server is added to the DMS system and the MS-FP fiber links are established.

Add DMS-Core, Billing Server-specific circuit packs to the MS

Note: Table MSINV must be datafilled before table MSCDINV.

At the CI prompt

1 From the MAPCI level, enter table MSCDINV by typing:

>TABLE MSCDINV
and pressing the Enter key. ↵

2 Add the MS-FLIS link circuit packs to the MS using the following format:

>ADD mscdkey slotinfo
where

mscdkey is a multiple-entry field comprised of 3 subfields: MSNUM, SHELFNUM, and CARDNUM.
MSNUM is the MS node number (0 or 1).
SHELFNUM is the shelf number where the circuit pack resides.
CARDNUM is the logical slot location of the circuit pack (1–26).

slotinfo consists of multiple-entry field comprised of 4 subfields: CARDTYPE, FRONTPEC, BACKPEC, and NUMLINKS

CARDTYPE	is the type of card (DS512).
FRONTPEC	is the PEC for the front circuit pack (NT9X17AD).
BACKPEC	is the PEC for the corresponding paddleboard (NT9X62BA).
NUMLINKS	is the number of links on the paddleboard (1).

Sample entry: **>ADD 0 0 12 ds512 nt9x17ad nt9x62ba 1**

3 Add the first inter-MS link circuit packs to the MS using the following format:

>ADD mscdkey slotinfo
and pressing the Enter key. ↵

where

mscdkey is a multiple-entry field comprised of 3 subfields: MSNUM, SHELFNUM, and CARDNUM.
MSNUM is the MS node number (0 or 1).
SHELFNUM is the shelf number where the circuit pack resides.
CARDNUM is the logical slot location of the circuit pack (1–26).

slotinfo consists of multiple-entry field comprised of 4 subfields: CARDTYPE, PROTOCOL, FRONTPEC, BACKPEC, and NUMLINKS

CARDTYPE	is the type of card (CHAIN).
PROTOCOL	is the chain protocol (DS512).
FRONTPEC	is the PEC for the front circuit pack (NT9X17DA).
BACKPEC	is the PEC for the corresponding paddleboard (NT9X20BB).
BODYPEC	contains up to seven groups of BODYPECS which are not required for Billing Server. Enter $.

Add DMS-Core, Billing Server-specific circuit packs to the MS (end)

Sample entry: **>ADD 0 0 18 chain ds512 nt9x17da nt9x20bb $**

4 Repeat step 3 to add the second inter-MS link circuit pack.

5 Add the first MS-FP link circuit packs to the MS using the following format:

>ADD mscdkey slotinfo
and pressing the Enter key. ↵

where

mscdkey	is a multiple-entry field comprised of 3 subfields: MSNUM, SHELFNUM, and CARDNUM. MSNUM is the MS node number (0 or 1). SHELFNUM is the shelf number where the circuit pack resides. CARDNUM is the logical slot location of the circuit pack (1–26).
slotinfo	consists of multiple-entry field comprised of 6 subfields: CARDTYPE, PROTOCOL, FRONTPEC, BACKPEC, NUMLINKS, and BODYPEC.

CARDTYPE	is the type of card (CHAIN).
PROTOCOL	is the chain protocol (DS512).
FRONTPEC	is the PEC for the front circuit pack (NT9X17DA).
BACKPEC	is the PEC for the corresponding paddleboard (NT9X62BA).
NUMLINKS	is the number of links on the paddleboard (2).
BODYPEC	contains up to seven groups of BODYPECS which are not required for Billing Server. Enter $.

Sample entry: **>ADD 0 0 20 chain ds512 nt9x17da nt9x62ba 2 $**

6 Repeat step 5 to add the second MS-FP link circuit pack.

7 Verify the datafill by typing:

>LIST ALL
and pressing the Enter key. ↵

DMS-Core, Billing Server-specific circuit packs are added to the MS.

Add FLIS to DMS system

Note: Table MSCDINV must be datafilled before table SUSHELF.

At the CI prompt

1 From the MAPCI level, enter table SUSHELF by typing:

>TABLE SUSHELF
and pressing the Enter key. ↵

2 Add FLIS to the DMS system using the following format:

>ADD shelfkey floor row framepos frametyp framenum shelfpos shelfpec cardinfo
where

shelfkey is a multiple-entry field comprised of five subfields: CONTROL, CTRLNUM, CARDNUM, PORTNUM, and LIUSHELF.

CONTROL	is the controlling entity (MS)
CTRLNUM	is the control number (enter NIL for MS).
CARDNUM	is the interface card number on the MS (12)
PORTNUM	is the port number (0).
LIUSHELF	is the APC shelf position (1).

floor is the floor position of the cabinet.
row is the row position of the cabinet.
framepos is the frame position of the cabinet.
frametyp is the frame type (EMC).
framenum is the number of the frame (0).
shelfpos is the base mounting position of the shelf (26).
shelfpec is the PEC for the shelf (NT9X7204).
cardinfo is a multiple-entry field comprised of three subfields:SLOT, FRONTPEC, and BACKPEC.

Note: Both sets of circuit packs are filled at this time.

SLOT	is the physical slot location of the circuit pack.
FRONTPEC	is the PEC for the front circuit pack (NT9X96AA).
BACKPEC	is the PEC for the corresponding paddleboard (NT9X98AA).

Sample entry: **>MS NIL 12 0 1 0 A 1 emc 0 26 nt9x7204 7 nt9x96aa nt9x98aa 30 nil ntex20aa $ 32 nt9x96aa nt9x98aa 8 nil ntex20aa $**

3 Verify the datafill by typing:

>POS MS NIL 12 0 1
and pressing the Enter key. ↵

FLIS is added the DMS system.

Define the EIUs

Note: There are no prerequisite datafill requirements for table APINV.

At the CI prompt

1 From the MAPCI level, enter table LIUINV by typing:

>TABLE LIUINV
and pressing the Enter key. ↵

2 Define the first EIU using the following format:

>ADD liuname location load procinfo cardinfo
where

liuname is a multiple-entry field comprised of two subfields: LIUTYPE and LIUNO.

- LIUTYPE — is the link interface unit type (EIU).
- LIUNO — is the link interface unit number assigned to the EIU.

location is a multiple-entry field comprised of three subfields: CTRL, SHELFNUM, and LIUSLOT.

- CTRL — is a multiple-entry field comprised of two subfields: CONTROL and LIMNUM.
  CONTROL is the controlling host (LIM).
  LIMNUM is the host number of which the EIU resides (0).
- SHELFNUM — is the shelf number on the APC where the EIU is located (1)
- LIUSLOT — is the physical slot number where the EIU is located. (8)

load is the software load name found in table PMLOADS.

procinfo is a multiple-entry field comprised of subfield PROCPEC.

- PROCPEC — is the processor card used in the EIU (NTEX22BB).

cardinfo is a multiple-entry field comprised of subfield APPLPEC.

- APPLPEC — is the application PEC (NT9X84AA). Subfields PBPEC, HEARTBEAT, and MAC_ADDRESS are required for NT9X84AA.
  PBPEC is corresponding paddleboard (NT9X85AA)
  HEARTBEAT is YES if the EIU expects a heartbeat indication signal from the media access unit.
  MAC_ADDRESS is a 12-character hex string representing the Media Access Control address. This field is EIU-specific and should not be duplicated with in a network.

Sample entry: **>EIU 102 MS 12 0 0 12 OSU02A0 NTEX22BB NT9X84AA NT9X85AA YES 0000075F00101**

3 Define the second EIU.

Define the EIUs (end)

NT9X85AA YES 0000075F00101

4 Verify the datafill by typing:

>LIST ALL
and pressing the Enter key. ↵

The EIUs are defined.

Define sequence of devices along the SCSI bus

Note: Tables PMLOADS, APINV, and APCDINV must be datafilled before table FPDIPINV.

At the CI prompt

1 From the MAPCI level, enter table FPDIPINV by typing:

>TABLE FPDIPINV
and pressing the Enter key. ↵

2 Add the first device sequence using the following format:

>ADD fpno scsibus sdipno shelf shelfpec floor row frame dipids
where

fpno	is the FP instance number (0).
scsibus	is the SCSI bus number on the FP (0 or 1).
sdipno	is the SCSI device paddleboard pair on the SCSI bus (0–5).
shelf	is the shelf number containing the SCSI device paddleboards.
shelfpec	is the PEC of the shelf containing the SCSI devices.
floor	is the floor position of the APC.
row	is the row position of the APC.
frame	is the frame position in the row containing the APC.
dipids	is a multiple–entry field comprised of three fields: CTRLNO, SLOTNO, and DIPPEC. This field can have up to five DIPIDS.

CTRLNO	is the SCSI bus controller number the which the device is attached.
SLOTNO	is the device slot location.
DIPPEC	is is the device PEC.

Sample entry: **>ADD 0 0 0 3 nt9x83aa 1 a 6 0 8 nt9x89aa 1 9 nt9x89aa $**

3 Add the remaining device sequences:

Sample entry: **>ADD 0 0 1 3 nt9x83aa 1 a 6 0 20 nt9x89aa 1 21 nt9x89aa $**
>ADD 0 1 0 3 nt9x83aa 1 a 6 0 14 nt9x89aa 1 15 nt9x89aa $
>ADD 0 1 0 3 nt9x83aa 1 a 6 0 26 nt9x89aa 1 27 nt9x89aa $

4 Verify datafill by typing:

>LIST ALL
and pressing the Enter key. ↵

Device sequences are defined.

Define storage devices

Note: Tables APINV, APCDINV, and FPDIPINV must be datafilled before table FPDEVINV.

At the CI prompt

1 From the MAPCI level, enter table FPDEVINV by typing:

>TABLE FPDEVINV
and pressing the Enter key. ↵

2 Define the storage devices using the following format:

>ADD fpno scsibus devno devtype devpec scsiid quadno shelf shelfpec floor row frame
where

fpno	is the instance number of the FP (0).
scsibus	is the SCSI bus number (0 or 1).
devno	is the device instance number (0–5).
devtype	is the type of device (DK or CT).
devpec	is the device drive PEC.
scsiid	is the SCSI bus identifier.
quadno	is the quadrant number (0–3).
shelf	is the device shelf number (0–3).
shelfpec	is the PEC for the device shelf.
floor	is the floor position of the cabinet.
row	is the row position of the cabinet.
frame	is the frame position of the cabinet.

Sample entry: **>ADD 0 1 5 dk nt9x90aa 3 0 0 nt9x83aa 0 r 1**

3 Repeat step 2 for each storage device.

4 Verify the datafill by typing:

>LIST ALL
and pressing the Enter key. ↵

The storage devices are defined.

Ensure the MS is datafilled

Note: There are no prerequisite datafill requirements for table APINV.

At the CI prompt

1 From the MAPCI level, enter table MSINV by typing:

>TABLE MSINV
and pressing the Enter key. ↵

2 Ensure the MS is datafilled by typing:

>LIST ALL
and pressing the Enter key. ↵

Establish the inter-MS links

ATTENTION

Inter-MS links are required for Billing Server to function.

Note 1: All data presented in this procedure is sample datafill only; the user should be familiar with data required at their site.

Note 2: Table MSCDINV must be datafilled before table MSILINV.

At the CI prompt

1 From the MAPCI level, enter table MSILINV by typing:

>TABLE MSILINV
and pressing the Enter key. ↵

2 Add the first inter-MS tuples using the following format:

>ADD imsl shelfno cardno info
where

imsl	is the inter-message switch link. There are two inter-MS links; Link 0 must be added before Link 1.	
shelfno	is the MS shelf number. Both links connect MS0 to MS1.	
cardno	is the logical slot number of the interlink circuit pack. (The circuit pack in slot 18 on MS0 connects to the MS1-slot 18 circuit pack.)	
info	is comprised of two subfields: PROTOCOL and PORTNO.	
	PROTOCOL	is the messaging protocol (DS512 for Billing Server).
	PORTNO	is always 0 (zero) for DS512 protocol.

Sample entry: **>ADD 0 0 18 DS512 0**

3 Add the second inter-MS link tuple:

Sample entry: **>ADD 1 0 19 DS512 0**

4 Verify the inter-MS links by typing:

>LIST ALL
and pressing the Enter key. ↵

Response:

```
IMSL SHELFNO    CARDNO     INFO
– – – – – – – – – – – – – – – – – – – – – –
   0       0        18  DS512 0
   1       0        19  DS512 0
```

Inter-MS links are datafilled.

Verify firmware load is datafilled

Note 1: The NT9X17DA cards on the MS require firmware.

Note 2: There are no prerequisite datafill requirements for table APINV.

At the CI prompt

1 From the MAPCI level, enter table MSFWLOAD by typing:

>TABLE MSFWLOAD
and pressing the Enter key. ↵

2 Verify firmware load is datafilled by typing:

>LIST ALL
and pressing the Enter key. ↵

Verify FP circuit packs and FP/device shelf power converter datafill

Note: Table APINV must be datafilled before table APCDINV.

At the CI prompt

1 From the MAPCI level, enter table APCDINV by typing:

>TABLE APCDINV
and pressing the Enter key. ↵

2 Check the default data supplied by the system when the FP was added to table APINV by typing:

>LIST ALL
Example of a MAP response:

```
SMNTYPE SMNNO SHELF SLOT FRONTCRD FRONTPEC BACKCRD BACKPEC
----------------------------------------------------------
     FP     0     0    1    POWER NT9X15AB     NIL NIL
     FP     0     0    4    POWER NT9X15AB     NIL NIL
     FP     0     0    8   MEMORY NT9X14DB    TERM NT9X21AB
     FP     0     0    9   MEMORY NT9X14DB     NIL NIL
     FP     0     0   10   MEMORY NT9X14DB    PORT NT9X88AA
     FP     0     0   11   MEMORY NT9X87AA    PORT NT9X88AA
     FP     0     0   12     PORT NT9X86AA    PORT NT9X62AA
     FP     0     0   13      CPU NT9X13LA     TIF NT9X26AB
     FP     0     0   14      CPU NT9X13LA     TIF NT9X26AB
     FP     0     0   15     PORT NT9X86AA    PORT NT9X62AA
     FP     0     0   16   MEMORY NT9X87AA    PORT NT9X88AA
     FP     0     0   17   MEMORY NT9X14DB    PORT NT9X88AA
     FP     0     0   18   MEMORY NT9X14DB     NIL NIL
     FP     0     0   19   MEMORY NT9X14DB    TERM NT9X21AB
     FP     0     0   33    POWER NT9X15AA     NIL NIL
     FP     0     0   36    POWER NT9X15AA     NIL NIL
     FP     0     2    1    POWER NT9X15AA     NIL NIL
     FP     0     2    4    POWER NT9X15AA     NIL NIL
     FP     0     2   33    POWER NT9X15AA     NIL NIL
     FP     0     2   36    POWER NT9X15AA     NIL NIL
     FP     0     3    1    POWER NT9X15AA     NIL NIL
     FP     0     3    4    POWER NT9X15AA     NIL NIL
     FP     0     3   33    POWER NT9X15AA     NIL NIL
     FP     0     3   36    POWER NT9X15AA     NIL NIL
```

Verify FP circuit packs and FP/device shelf power converter datafill
(end)

the following format:

>ADD smntype smnno shelf slot frontcrd frontpec backcrd backpec
or
>CHANGE smntype smnno shelf slot frontcrd frontpec backcrd backpec
where

smntype	is the sync–matched node type (FP).
smnno	is the sync–matched node instance number (0).
shelf	is the shelf number containing the FP (0–3).
slot	is the physical slot location of the circuit pack.
frontcrd	is the classification of the card (CPU, MEMORY, PORT, POWER, or NIL).
frontpec	is the PEC of the FRONTCRD.
backcrd	is the classification of the card (PORT, TERM, TIF, or NIL).
backpec	is the PEC of the BACKCRD.

Sample entry: **>ADD fp 0 3 1 power nt9x90aa nil nil**

Sample entry: **>POS fp 0 0 10**
>CHANGE memory nt9x14db port nt9x88aa

4 Verify datafill by typing:

>LIST ALL
and pressing the Enter key. ↵

Datafill is verified and corrected.

Verify software load is datafilled

Note: There are no prerequisite datafill requirements for table PMLOADS.

At the CI prompt

1 From the MAPCI level, enter table PMLOADS by typing:

>TABLE PMLOADS
and pressing the Enter key. ↵

2 Verify software load is datafilled by typing:

>LIST ALL
and pressing the Enter key. ↵

Billing Server SCSI device management

Billing Server uses distributed recording manager (DRM) and fault tolerant file system (FTFS) to manage the small computer standard interface (SCSI) devices.

DRM

DRM functions as a seamless interface to application file systems that are distributed across multiple nodes in the SuperNode multicomputing base. DRM provides the following management functions for Billing Server SCSI devices:

- volume allocation, management, and error recovery
- file naming, access, management, and error recovery
- directory generation and block programming
- access polling application from a downstream processor using the FTAM protocol
- ability to backup files to core input/output controller (IOC) 9-track tape or digital audio tape (DAT)

Billing Server is registered with DRM during software initialization. The registration process automatically datafills tables DRMAPPL and DRMPOOL. You can then modify the fields pertaining to DRM operations in table DRMAPPL. You must first create file volumes using the DISKADM CV command before FTFS can accept data to write to disk.

Note: To change table DRMPOOL, use the DRM menu to mount or demount volumes.

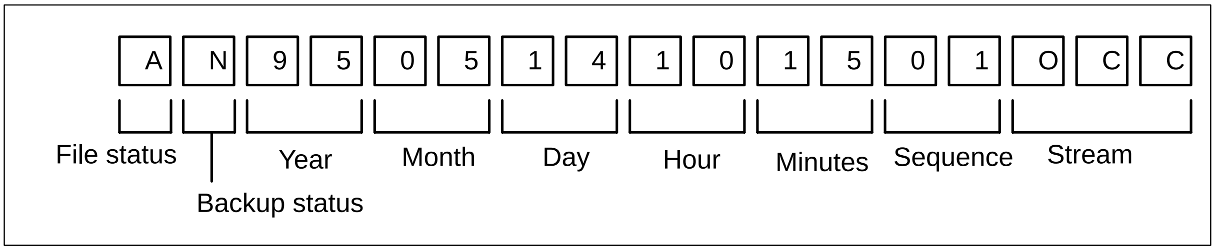
File handling

DRM automatically generates file names as follows (Figure 5-1):

- The first character of the filename is the file status.
  - A – Active file (file currently being written to by DRM)
  - U – Unprocessed file (file not processed by downstream processor)

  - — P – Processed file (proper acknowledgement of file reception is received from downstream processor)
  - — R – Removed file (removed from list of files that a DRM audit can delete)
- The second character of the filename is the backup status.
  - — B – file has been backed up to tape
  - — N – file has not been backed up to tape
- The third through eighth characters of the filename represent the date.
  - — YY – Year (00–99)
  - — MM – Month (00–12)
  - — DD – Day (00–31)
- The ninth through twelfth characters of the filename represent the time.
  - — hh – Hour (00–23)
  - — mm – Minute (00–59)
- The thirteenth and fourteenth characters of the filename represent a unique, two-digit sequence number assigned by DRM.
- Up to four remaining characters contain the first 1–4 characters of the stream name.

Figure 5-1
DRM file naming conventions



There are seven types of Billing Server files:

- AN – Active
- UN – Unprocessed, not backed up
- UB – Unprocessed, backed up
- PN – Processed, not backed up
- PB – Processed, backed up
- RN – Removed, not backed up

- RB – Removed, backed up

Note: Processed File means that the downstream processor has requested and acknowledged receipt of the file.

File rotation

File rotation refers to the process of opening a new file and closing the previously active file. An active file is closed when:

- a scheduled rotation occurs (set in table DRMAPPL)
- the file exceeds maximum size (set in table DRMAPPL)
- a DRM menu command (rotates the file)
- a device error occurs on the disk, causing a system rotation

Fault Tolerant File System

FTFS stores and retrieves data on the Billing Server SCSI device. FTFS also allows access to the disk shadowing feature. You can copy files to and restore them from core IOC 9-track tape. FTFS has the following characteristics:

- file security and access control
- reliability through consistent high availability in the presence of faults
- hierarchical file name directories
- supports access from foreign operating systems
- caching of frequently accessed data for enhanced data throughput and data access response time
- supports disk shadowing
- high degree of consistency in the presence of faults

Disk shadowing

Disk shadowing ensures backup disk availability in case a disk fails. A shadow set may contain two to five members which are synchronous (each disk contains an exact copy of the data on the master disk).

Disk shadowing offers high storage reliability, speeds responses, and increases data throughput by increasing the number of input and output requests the FTFS can handle.

Operational measurements

Billing Server uses the UCS DMS operational measurement (OM) collection system to collect and process operational measurements.

The OM system collects and reports operational measurements, which are integer quantities indicating the quality of a particular aspect of switch performance.

Table 5-1 provides a description, history, associated registers, and validation information for the OMs that are relevant to the Billing Server.

Table 5-1
Operational measurements

Group	Register	Information
DRM		**Description:** DRM is provided to count the number of blocks written to disk, the number of application records within those blocks, and the number of file rotations.
	RECORDS	**Description:** RECORDS is a count of the number of individual DRM application records written into a block. A block contains one or more records grouped together to be written to disk. When an application requests that DRM write a block to disk, it also passes DRM a count of the application records in that block. The number of records is not used by DRM except to add to the RECORDS register. **Associated registers:** BLOCKS **EXT registers:** RECORDS2 **Register validation:** If the application places ten records in a block and gives that block to DRM to write to disk, then RECORDS increases by ten and BLOCKS by 1.
Note: For additional information on these OMs, refer to the UCS DMS-250 *Operational Measurements Reference Manual*.		
—continued—		

Table 5-1
Operational measurements (continued)

Group	Register	Information
	BLOCKS	**Description:** BLOCKS is a count of the number of blocks DRM was given to write to disk. The block size is completely dictated by the DRM application. A block is a collection of one or more application records. **Associated registers:** RECORDS **Register validation:** If the application places ten records in a block and gives that block to DRM to write to disk, then RECORDS increases by ten and BLOCKS by 1.
	ROTATES	**Description:** ROTATES is a count of the number of new files opened. **Associated registers:** None. **Register validation:** None.
Note: For additional information on these OMs, refer to the UCS DMS-250 *Operational Measurements Reference Manual*.		
—end—		

Logs

DRM generates logs (Table 5-2) that fall into one of the following categories:

- functionality loss due to hardware faults
- maintenance state changes
- data communication failures
- disk utilization exceeding thresholds
- operations affecting switch configuration (datafill)

Table 5-2
Logs

Log	Explanation
DRM300 (Note 1)	An alarm condition has been created that can be cleared or the DRM has recovered from an alarm condition.
DRM600 (Note 1)	A ROTATE has been initiated and completed by DRM for an application.
DRM601 (Note 2)	A COPY has been successfully completed from the DRM level of the MAP terminal.
Note 1: For additional information on these logs, refer to the *AD3363* design document.	
Note 2: For additional information on this log, refer to the *AD4333* design document	

User interface

The following utilities are required to perform SCSI device management functions:

- DISKADM utility
- DISKUT utility
- DRM menu
- SHADOWUT utility

DISKADM

The DISKADM utility supports disk administration and functions only when the device is in the MBSY state. Table 5-3 lists the DISKADM commands available to Billing Server.

Table 5-3
DISKADM commands

Command	Syntax	Description
CREATEVOL	CV volume _name size volume_type [logical_block_size]	Creates a volume on the disk.
DELETEVOL	DDV volume_name	Deletes the volume on the disk
DISPLAYDISK	DD	displays information about the disk
DISPLAYVOLS	DV	Displays general information about all volumes.
FORMATDISK	FD disk_name	Formats the disk.
HELP	HELP	Accesses DISKADM help.
QUIT	QUIT	Exits the DISKADM utility.
REINITVOL	RV volume_name	Erases all files on a volume without destroying the volume.
SETCACHESIZE	SCS volume_name cache_type cache_size	May be needed to change block size.

DISKUT

The DISKUT utility is used to perform regular operations on the Billing Server SCSI devices. Table 5-4 lists the DISKUT commands available to Billing Server.

Table 5-4
DISKUT commands

Command	Syntax	Description
LISTFL	LF ::node/volname or for DAT LF ::remotevolname node	Displays information about files on the disk, including creation date, modification date, organizational type, file type, file size in bytes, file allocation in kbytes, and file name.
HELP	HELP	Accesses DISKUT help.
LISTVOLS	LV node	Lists all volumes on all INSV disk drives.

—continued—

Table 5-4
DISKUT commands (continued)

Command	Syntax	Description
QUIT	QUIT	Exits the DISKUT utility.
VOLINFO	VINFO	Displays information on the volumes.
	—end—	

DRM menu

The DRM is used to perform control and review functions. The DRM menu provides the following commands to the user:

- INFO – obtain information on the stream, node, registration, audit schedules, rotate schedules, volumes, trouble diagnostics, last error, mounted volumes and their files, or the active volume
- AUDIT – audit the recording measurements, space data, volumes, applications, and file directory
- RENAME – rename files
- MOUNT – mount volumes to DRM
- DEMOUNT – demount volumes from DRM
- RESET – reset stream errors and rotate counters
- DAT – manipulate DAT
  - — mount tape
  - — write to tape
  - — verify contents of tape
  - — restore contents from tape
  - — eject tape
- VIEW – view a hexadecimal dump of a file
- TCOPY – copy files to 9-track tape
- MONITOR – monitor the active file
- COPY – copy DIRP files on the computing module (CM) to Billing Server SCSI disk

To enter the DRM menu from a MAP terminal, type the following:

>MAPCI;MTC;APPL;OAMAP;DRM

Example of a MAP response:

```
CM     MS     IOD     NET     PM     CCS     TRKS     EXT     APPL

DRM                  OAMAP   EIO       SWMTC
0      QUIT
2
3      Info_
4      Audit_      DRM:       AFT:     MNP:        SBS:
5      Rotate_
6      Rename_
7      Mount_
8      Demount_    DRM:
9      Reset_
10     DAT_
11     View_
12     Tcopy_
13     Monitor_
14
15
16
17
18     Copy_



       TIME  12:00
```

Table 5-5 shows the DRM menu commands and their syntax.

Table 5-5
DRM menu commands

Command	Syntax	Description
QUIT	QUIT	Exits the DRM menu.
INFO	INFO stream type	Provides information on the status of Billing Server. The type variable may be one of the following, NODES, REG, SCHED, ROT, ROTVOL, VOL, VOLDIAG, VOLERR, VOLFIL, FIL, or ALL.
AUDIT	AUDIT stream type	Instructs DRM to check its systems. The type variable may be one of the following: ALL, REC, SPACE, VOL, APPL, or FDIR.
ROTATE	ROTATE stream	Instructs DRM to close current active file and open a new file.
RENAME	RENAME stream filetmpl newstate	Changes the first letter of the file name to another state. Active files cannot be changed.
MOUNT	MOUNT stream volname nodename	Adds a volume to DRM system.
DEMOUNT	DEMOUNT stream volname	Removes a volume from DRM.
RESET	RESET stream counter_type	Resets the stream error or rotate counters.
DAT	DAT stream subcommand	Controls DAT devices.
VIEW	VIEW stream filetmpl charcode blocknum	Shows a HEX dump of the file with ASCII or EBCDIC translation of characters.
TCOPY	TCOPY device stream filetmpl	Copies file to 9-track tape.
MONITOR	MONITOR stream	Monitors the active file.
COPY	COPY	Copies DIRP files to Billing Server.

Disk shadowing management

Table SHADOW is used to define shadow set configurations. SHADOWUT is used to define and operate the disk shadowing utility.

Table SHADOW

Table SHADOW describes all shadow sets defined on a DMS SuperNode system. Each tuple in the table contains the data for one shadow set. Set definitions, set deletions, and member changes are made in table SHADOW.

SHADOWUT

The SHADOWUT utility works with FTFS to define and operate the disk shadowing utility. Table 5-6 lists the SHADOWUT commands that are available to Billing Server. See table SHADOW for specifics on defining the datafill.

Table 5-6
SHADOWUT commands

Command	Syntax	Description
DISPLAYSET	DIS set_name	Displays information about the previously defined shadow sets or a specific shadow set. (This command does not require the disk to be in the manual busy state.)
DEFINESET	Add member	Displayed when ”q SHADOWUT” is entered at CI level. It can be done with table SHADOW
DELETESET	Delete member	Displayed when ”q SHADOWUT” is entered at CI level. It can be done with table SHADOW
STARTSHADOW	SS set_name	Starts shadowing on a set. Disks are put into sync if necessary.
STOP SHADOW	STS set_name	Stops shadowing on a set.
STARTMEMBER	SM set_name disk_name	Starts shadowing on a member.
STOP MEMBER	STM set_name disk_name	Stops shadowing on a member.
SETNODE	SETNODE nodename nodenum	Sets the node of reference. (This command does not require the disk to be in the manual busy state.)
QUIT	QUIT	Quits SHADOWUT.

Note: Disks must be in the manual busy state to perform SHADOWUT functions.

SCSI device management procedures

In several of the procedures, the MAP terminal prompts the user for one of the actions in Table 5-7.

Table 5-7
User actions

Terminal prompt	User action
`Press <CR> to continue, or enter QUIT`	Press the enter key to continue the process or type QUIT to discontinue the process.
`Select a file number`	Enter the number of the file used in the process preceded by a space.

Table 5-8 shows the procedures available to the user for maintaining the DRM application.

Table 5-8
SCSI device management procedures

Procedure	Utility – Command	Page
Back up files to DAT	DRM menu – DAT WRITE	5–14
Back up files to core IOC 9-track tape	DRM menu – TCOPY	5-16
Bring up file processor controllers and disks	PM menu – POST Devices menu – BSY and RTS	5–19
Configure DRM settings	table DRMAPPL	5–23
Create billing volumes	DISKADM – CV	5–25
Define shadow set	table SHADOW	5–31
Delete shadow set	table SHADOW	5–35
Delete billing volume	DISKADM – DDV	5–39
Demount DAT	DRM menu – DAT EJECT	5–44
Demount disk volume	DRM menu – DEMOUNT	5–46
Determine what disks are in a shadow set	SHADOWUT – DIS	5–48
Determine which volumes are on the FP	DISKUT – LV	5–50

—continued—

Table 5-8
SCSI device management procedures (continued)

Procedure	Utility – Command	Page
Format the disk	DISKADM – FD	5–51
Manually rotate a DRM-managed file	DRM menu – ROTATE	5–55
Monitor the state of the active DRM file	DRM menu – MONITOR	5–57
Mount DAT	DRM menu – DAT MOUNT	5–59
Mount disk volume	DRM menu – MOUNT	5–61
Obtain file information	DRM menu – INFO	5–64
Perform a DRM audit	DRM menu – AUDIT	5–74
Rename a DRM-managed file	DRM menu – RENAME	5–81
Reset volume counters	DRM menu – RESET	5–83
Restore files from DAT	DRM menu – DAT RESTORE	5–85
Start shadowing	SHADOWUT – SS	5–87
Stop shadowing	SHADOWUT – STS	5–92
Verify DAT contents	DRM menu – DAT VERIFY	5–93
View contents of a file	DRM menu – VIEW	5–95
View mounted volumes	table DRMPOOL	5–97
—end—		

Back up files to DAT

CAUTION
Lost data
Do not attempt to back up active files to DAT. Active files cannot be backed up and data will be lost!

At the CI prompt

1 Access the DRM level of the MAP terminal by typing:

>MAPCI;MTC;APPL;OAMAP;DRM
and pressing the Enter key. ↵

```
CM    MS    IOD    NET    PM    CCS    TRKS    EXT    APPL

DRM               OAMAP   EIO
0     QUIT
2
3     Info_
4     Audit_     DRM:      AFT:     MNP:       SBS:
5     Rotate_
6     Rename_
7     Mount_
8     Demount_   DRM:
9     Reset_
10    DAT_
11    View_
12    Tcopy_
13    Monitor_
14
15
16
17
18    Copy_



      TIME  12:00
```

2 Mount DAT by following the procedure entitled *Mount DAT*.

3 Back up each file (or write each file to DAT) by typing:

>DAT WRITE stream filetype
and pressing the Enter key. ↵

Back up files to DAT (end)

stream is the stream defined in table CRSMAP.
filetype is U for unprocessed, P for processed, R for removed.

Sample entry: **>DAT WRITE occ u**

Example of a MAP response:

```
Sending request to FP0 . . .
      Request sent.  (Please wait . . .)

RESPONSE READY.
```

Example of a MAP response:

```
These files match the supplied filename template:
  1.  UN9208170300000OCC     2.  UN9208200400000OCC

Select a :
```

Sample entry: **> 1**

Example of a MAP response:

```
Sending request . . .
> == Request sent == >     (Please wait . . .)

RESPONSE READY.

 Press <CR> to continue, or enter QUIT:
```

Sample entry: **>** ↵

Example of a MAP response:

```
Writing file
UN9208170300000OCC . . .
File written.
```

Back up files to core IOC 9-track tape

Back up files to core IOC 9-track tape

At the CI prompt

1 Access the DRM level of the MAP terminal by typing:

>MAPCI;MTC;APPL;OAMAP;DRM
and pressing the Enter key. ↵

Example of a MAP response:

```
CM     MS     IOD     NET     PM     CCS     TRKS     EXT     APPL

DRM                  OAMAP   EIO
0      QUIT
2
3      Info_
4      Audit_      DRM:      AFT:     MNP:       SBS:
5      Rotate_
6      Rename_
7      Mount_
8      Demount_    DRM:
9      Reset_
10     DAT_
11     View_
12     Tcopy_
13     Monitor_
14
15
16
17
18     Copy_



       TIME  12:00
```

2 Back up files to core IOC 9-track tape by typing:

>TCOPY 9track_dev applid filetmpl
and pressing the Enter key. ↵

where

9track_dev	is the 9-track tape drive number to which the file will be copied.
applid	is the ID of an application in table DRMAPPL.

Back up files to core IOC 9-track tape (continued)

filetmpl is the first character of a file name followed by any number of characters that sufficiently identify one or more files. It must begin with U, P, R, or $. A $ wildcard may be used to replace any number of characters within the file template.

Sample entry: **TCOPY 4 occ u**

Example of a MAP response:

```
Sending request to CM...
> == Request sent == >   (Please wait...)

RESPONSE READY.

   Press <CR> to continue, or enter QUIT:
```

Sample entry: **>**↵

Example of a MAP response:

```
These files match the supplied filename template:
   1.  UN920817030000OCC    2.  UN920818040000OCC

Select a
```

Sample entry: **>1**

Example of a MAP response:

```
Sending request...
> == Request sent == >     (Please wait...)

RESPONSE READY.

   Press <CR> to continue, or enter QUIT:
```

Sample entry: **>**↵

Example of a MAP response:

```
Mounting Tape Drive ’4’
The tape just mounted is already labeled as a DRM tape.
Do you wish to OVERWRITE it? (NO indicates APPEND.)
```

CAUTION
Lost data
Overwriting the file will erase it.

3 Indicate that the file should not be overwritten by typing:

>NO
and pressing the Enter key. ↵

Back up files to core IOC 9-track tape (end)

Example of a MAP response:

```
Writing file
UN920817030000OCC ...
File written.
```

Bring up file processor controllers and disks

At the CI prompt

1 Access the PM level of the MAP terminal by typing:

> MAPCI;MTC;PM
and pressing the Enter key. ↵

Example of a MAP response:

```
CM     MS     IOD     NET     PM     CCS     TRKS     EXT     APPL

PM                            SysB   ManB    OffL   CBsy ISTb InSv
0      QUIT       PM           0      0       9      0    1    10
2      Post
3
4
5
6
7
8
9
10
11     Disp_
12
13     Status
14     IPML
15
16
17
18



       TIME  12:00
```

2 Post the FP by typing:

> POST FP 0
and pressing the Enter key. ↵

Example of a MAP response:

```
CM     MS     IOD     NET     PM     CCS     TRKS     EXT     APPL

PM                            SysB  ManB   OffL  CBsy ISTb InSv
0      QUIT      PM            0     0      9     0    1    10
2      Post      FP            0     0      0     0    0     1
3      Plane
4      Devices  FP 0: SBS:  Plane       Devices
5               InSv
6      Tst
7      Bsy      POST:
8      RTS
9      Off
10     LoadPM
11     Disp_
12     Next
13
14     QueryPM_
15
16
17
18



       TIME  12:00
```

3 Access the DEVICES level of the MAP terminal by typing:

> DEVICES
and pressing the Enter key. ↵

Example of a MAP response:

```
CM     MS     IOD     NET     PM     CCS     TRKS     EXT     APPL

PM                            SysB  ManB   OffL  CBsy ISTb InSv
0      QUIT      PM            0     0      9     0    1    10
2      PostDEV_  FP            0     0      0     0    0     1
3      Plane
4             FP 0: SBS     Plane       Devices
5               InSv
6      Tst_
7      Bsy_            CTRL0      CTRL0          DEVICE
8      RTS_    DABM                            0 1 2 3 4 5
9      OffL_   SCSI 0          EN        DIS          - - -
10             SCSI 1          EN        DIS        - - - -
11
12            DEVICES:
13
14     QueryFP_
15     SwEn_
16
17
18



       TIME  12:00
```

4 Busy the FP controllers by typing:

> BSY CTRL 0
and pressing the Enter key. ↵

> BSY CTRL 1
and pressing the Enter key. ↵

Example of a MAP response:

```
FP 0 Busy CTRL 0:  Command request has been submitted.
FP 0 Busy CTRL 0:  Command passed.

FP 0 Busy CTRL 1:  Command request has been submitted.
FP 0 Busy CTRL 1:  Command passed.
```

5 Return the FP controllers to service by typing:

> RTS CTRL 0
and pressing the Enter key. ↵

> RTS CTRL 1
and pressing the Enter key. ↵

Bring up file processor controllers and disks (end)

```
FP 0 RTS CTRL 0:  Command request has been submitted.
FP 0 RTS CTRL 0:  Command passed.

FP 0 RTS CTRL 1:  Command request has been submitted.
FP 0 RTS CTRL 1:  Command passed.
```

Note: Unformatted disk drives will fail to RTS.

6 Busy the FP disks by typing:

> BSY DEV 0 ALL
and pressing the Enter key. ↵

> BSY DEV 1 ALL
and pressing the Enter key. ↵

Example of a MAP response:

```
FP 0 Busy DEV 0 ALL:  Command request has been submitted.
FP 0 Busy DEV 0 ALL:  Command passed.

FP 0 Busy DEV 1 ALL:  Command request has been submitted.
FP 0 Busy DEV 1 ALL:  Command passed.
```

7 Return the FP disks to service by typing:

> RTS DEV 0 ALL
and pressing the Enter key. ↵

> RTS DEV 1 ALL
and pressing the Enter key. ↵

Example of a MAP response:

```
FP 0 RTS DEV 0 ALL:  Command request has been submitted.
FP 0 RTS DEV 0 ALL:  Command passed.

FP 0 RTS DEV 1 ALL:  Command request has been submitted.
FP 0 RTS DEV 1 ALL:  Command passed.
```

Configure DRM settings

At the CI prompt

1 From the MAPCI level, enter table DRMAPPL by typing:

>TABLE DRMAPPL
and pressing the Enter key. ↵

Note: Table DRMAPPL does not allow additions, only changes to existing data.

2 Check the current datafill by typing:

>LIST ALL
and pressing the Enter key. ↵

3 Position on the tuple to change by typing:

>POS applname
where

applname is the application name.

4 Change the defaults as required by typing:

>CHANGE

5 Enter the changes required as prompted by the system.

where

poolname	is the pool name which must be identical to the POOLNAME entry in table DRMPOOL.
alarmmn	is the minor alarm threshold in megabytes (0–32767).
alarmmj	is the major alarm threshold in megabytes (0–32767).
alarmcr	is the critical alarm threshold in megabytes (0–32767).
retpd	is the retention period of a file in days (0–365).
filedate	dates the file when it is opened or closed (OPENED or CLOSED).
sheddays	sets the daily rotation schedule; enter Y or N for each day of the week, Monday through Sunday.
shedbase	sets the schedule rotation base; enter the hour of the first rotation (0–23).
shedincr	sets the scheduled rotation increment; enter the number of hours between the scheduled rotations (X1 for hourly, X2, X3, X4, X6, X8, X12, X24, or NOROTATE).
clostate	specifies the closed file's state (P, R, or U). This identifies state the file goes to when rotated (recommended default is U).
maxfsize	is the maximum size for a DRM file (00600 Mbytes). The maximum size must be supported by the downstream processor and 9-track as required by the user.
forcbkup	instructs DRM to erase processed files (P-files) that have expired, only if files have been backup to DAT (Y or N).

Configure DRM settings (end)

6 Verify the datafill by typing:

>POS applname

and pressing the Enter key. ↵

The user-defined DRM defaults are set.

Create billing volumes

ATTENTION

Identical billing volumes must be created on each disk belonging to shadow set.

Create billing volumes

At the CI prompt

1 Access the PM level of the MAP terminal by typing:

> MAPCI;MTC;PM
and pressing the Enter key. ↵

Example of a MAP response:

```
CM     MS     IOD     NET     PM     CCS     TRKS     EXT     APPL

PM                            SysB   ManB    OffL   CBsy  ISTb  InSv
0      QUIT       PM           0      0       9      0     1     10
2      Post
3
4
5
6
7
8
9
10
11     Disp_
12
13     Status
14     IPML
15
16
17
18



       TIME  12:00
```

2 Post the FP by typing:

> POST FP 0
and pressing the Enter key. ↵

Example of a MAP response:

```
CM     MS     IOD     NET     PM     CCS     TRKS     EXT     APPL

PM                            SysB  ManB   OffL  CBsy ISTb InSv
0      QUIT       PM           0     0      9     0    1    10
2      Post       FP           0     0      0     0    0     1
3      Plane
4      Devices  FP 0: SBS:  Plane       Devices
5               InSv
6      Tst
7      Bsy      POST:
8      RTS
9      Off
10     LoadPM
11     Disp_
12     Next
13
14     QueryPM_
15
16
17
18



       TIME  12:00
```

3 Access the DEVICES level of the MAP terminal by typing:

> DEVICES
and pressing the Enter key. ↵

Example of a MAP response:

```
CM     MS     IOD     NET     PM     CCS     TRKS     EXT     APPL

PM                        SysB  ManB   OffL  CBsy ISTb InSv
0      QUIT      PM         0     0      9     0    1    10
2      PostDEV_  FP         0     0      0     0    0     1
3      Plane
4             FP 0: SBS    Plane      Devices
5               InSv
6      Tst_
7      Bsy_            CTRL0      CTRL0          DEVICE
8      RTS_    DABM                            0 1 2 3 4 5
9      OffL_   SCSI 0          EN        DIS         - - -
10             SCSI 1          EN        DIS       - - - -
11
12            DEVICES:
13
14     QueryFP_
15     SwEn_
16
17
18



       TIME  12:00
```

4 Busy the disk on which volumes are to be formatted/created by typing:

> BSY DEV scsi_bus# scsi_dev#
and pressing the Enter key. ↵

where

scsi_bus# is the SCSI bus number (0 or 1).
scsi_dev# is the SCSI device number.

Sample entry: **BSY DEV 0 0**

Example of a MAP response:

```
FP 0 Busy DEV 0 0:  Command request has been submitted.
FP 0 Busy DEV 0 0:  Command passed.
```

5 Perform disk administration by typing:

> DISKADM devname nodetype nodenum
and pressing the Enter key. ↵

where

devname	is the name of the device needing administration.
nodetype	is the node on which the device resides.
nodenum	is the node number.

Sample entry: **DISKADM dk00 fp 0**

Example of a MAP response:

```
Start up command sequence is in progress.
This may take a few minutes.
Administration of device DK00 on FP0 is now active.
DISKADM; FP0
```

6

If	Do
disk needs formatting	step 7
disk does not need formatting	step 9

7

CAUTION
Lost data
The following command causes all data on the disk to be lost.

Erase all volumes and files on the disk by typing:

> FD QUICK FORCE
and pressing the Enter key. ↵

Example of a MAP response:

```
           ***** WARNING *****
Formatting of DK00
Will destroy the contents of the disk.

The formatting will:
  allocate 3 spare or alternate sectors per track,
  allocate 16 spare or alternate tracks per disk,
  use the G defect list,
  assign QUICK as the name for the disk,
  perform quick format,
  include force option.

Do you want to continue?
Please confirm (”YES” or ”NO”):
```

8 Confirm by typing:

> YES
and pressing the Enter key. ↵

Example of a MAP response:

```
Initializing the system data structures on the disk.
Formatting and initialization of the disk is completed.
```

9 Create a billing volume for use with Billing Server by typing:

> CV volname volsize voltype [LBLOCK blksize]
and pressing the Enter key. ↵

where

volname	is the logical name of the volume (any 8 characters).
volsize	is the size of the volume (1–32767).
voltype	is the type of volume (FTFS)
LBLOCK	precedes the logical block size
blksize	is the size of the volume in Kbytes (1, 2, 4, 8, 16, 32, or 64; the recommended size is 16)

Note: If LBLOCK is not defined, the default logical block size is 1.

Sample entry: **CV occ 600 ftfs LBLOCK 16**

Example of a MAP response:

```
FTFS volume OCC will be created on DK00.
Volume size:                600 megabytes
first FID table extent size:        32754 entries
Volume Free Space Map size:     7936 segments

Do you want to continue?
Please confirm (”YES” or ”NO”):
```

Create billing volumes (end)

10 Confirm by typing:

> YES
and pressing the Enter key. ↵

Example of a MAP response:

```
Creation of the volume is completed.
```

11 Quit DISKADM by typing:

> QUIT
and pressing the Enter key. ↵

12 Bring the disk back into service by typing:

> RTS DEV scsi_bus# scsi_dev#
and pressing the Enter key. ↵

where

scsi_bus# is the SCSI bus number.
scsi_dev# is the SCSI device number.

Example of a MAP response:

```
FP 0 RTS DEV 0 0:  Command request has been submitted
FP 0 RTS DEV 0 0:  Command passed
```

Define shadow set

Define shadow set

At the CI prompt

1 Access the PM level of the MAP terminal by typing:

> MAPCI;MTC;PM
and pressing the Enter key. ↵

Example of a MAP response:

```
CM     MS     IOD     NET     PM     CCS     TRKS     EXT     APPL

PM                            SysB   ManB    OffL   CBsy  ISTb  InSv
0      QUIT       PM           0      0       9      0     1     10
2      Post_
3
4
5
6
7
8
9
10
11     Disp_
12
13     Status
14     IPML
15
16
17
18



       TIME  12:00
```

2 Post the FP by typing:

> POST FP 0
and pressing the Enter key. ↵

Example of a MAP response:

```
CM     MS     IOD     NET     PM     CCS     TRKS     EXT     APPL

PM                            SysB  ManB   OffL  CBsy ISTb InSv
0      QUIT      PM            0     0       9     0    1    10
2      Post      FP            0     0       0     0    0     1
3      Plane
4      Devices  FP 0: SBS:  Plane      Devices
5               InSv
6      Tst
7      Bsy      POST:
8      RTS
9      Off
10     LoadPM
11     Disp_
12     Next
13
14     QueryPM_
15
16
17
18



       TIME  12:00
```

3 Access the DEVICES level of the MAP terminal by typing:

> DEVICES
and pressing the Enter key. ↵

Example of a MAP response:

```
CM    MS    IOD    NET    PM    CCS    TRKS    EXT    APPL

PM                        SysB  ManB   OffL  CBsy ISTb InSv
0     QUIT      PM         0     0      9     0    1    10
2     PostDEV_  FP         0     0      0     0    0     1
3     Plane
4            FP 0: SBS    Plane      Devices
5              InSv
6     Tst_
7     Bsy_           CTRL0      CTRL0           DEVICE
8     RTS_    DABM                            0 1 2 3 4 5
9     OffL_   SCSI 0          EN        DIS         - - -
10            SCSI 1          EN        DIS       - - - -
11
12           DEVICES:
13
14    QueryFP_
15    SwEn_
16
17
18



      TIME  12:00
```

4 Busy the members of the shadow set by typing:

> BSY DEV scsi_bus# scsi_dev#
for the master and pressing the Enter key. ↵

> BSY DEV scsi_bus# scsi_dev#
for the slave and pressing the Enter key. ↵

where

scsi_bus# is the SCSI bus number (0 or 1).
scsi_dev# is the SCSI device number.

Sample entry: **BSY DEV 0 0**
BSY DEV 1 0

Example of a MAP response:

```
FP 0 Busy DEV 0 0:  Command request has been submitted.
FP 0 Busy DEV 0 0:  Command passed.

FP 0 Busy DEV 1 0:  Command request has been submitted.
FP 0 Busy DEV 1 0:  Command passed.
```

Define shadow set (end)

5 From the MAPCI level, enter table SHADOW by typing:

>TABLE SHADOW
and pressing the Enter key. ↵

6 Add the first shadow set by typing:

>ADD nodetype nodeno setname devtype
and pressing the Enter key. ↵

where

nodetype is the node on which the shadow set resides (FP).
nodeno is the number of the node identified in field NODETYPE (0).
setname is the shadow set's name.
devtype is the device type (SCSIDK). Field DEVTYPE has the following refinements (PERMSCSI and OMEMSCSI):
PERMSCSI is the shadow set's permanent disk (master disk) and is specified by a SCSI-bus number (0 or 1) and a device number (0–5)
OMEMSCSI contains the slave shadow sets. One to four disks may be defined here.

Sample entry: **>ADD fp 0 fp0ss0 0 0 0 1 $**

7 Verify the datafill by typing:

>LIST
and pressing the Enter key. ↵

The shadow set is defined.

Delete shadow set

At the CI prompt

1 Access the PM level of the MAP terminal by typing:

> MAPCI;MTC;PM
and pressing the Enter key. ↵

Example of a MAP response:

```
 CM     MS     IOD     NET     PM     CCS     TRKS     EXT     APPL
 .      .      .       .       .      .       .        .       .

PM                                 SysB  ManB  OffL  CBsy  ISTb  InSv
0    Quit          PM                 0     0     9     0     1    10
2    Post_
3
4
5
6
7
8
9
10
11   Disp_
12
13   Status
14   IPML
15
16
17
18

     TIME  12:00
```

2 Post the FP by typing:

> POST FP 0
and pressing the Enter key. ↵

Example of a MAP response:

```
    CM      MS       IOD      NET      PM       CCS      TRKS     EXT      APPL
     .       .        .        .        .        .        .        .        .

   FP                                     SysB   ManB   OffL   CBsy   ISTb   InSv
   0     Quit              PM                0      0      9      0      1     10
   2     Post              FP                0      0      0      0      0      1
   3     Plane
   4     Devices        FP 0:    SBS            Plane       Devices
   5                    InSv                      .            .
   6     Tst
   7     Bsy            POST:
   8     RTS
   9     OffL
   10    LoadPM
   11    Disp_
   12    Next
   13
   14    QueryPM_
   15
   16
   17
   18

         TIME    12:00
```

3 Access the DEVICES level of the MAP terminal by typing:

> DEVICES
and pressing the Enter key. ↵

Example of a MAP response:

```
 CM     MS     IOD     NET     PM     CCS     TRKS     EXT     APPL
  .      .      .       .       .      .       .        .       .

Devices                               SysB  ManB  OffL  CBsy  ISTb  InSv
0    Quit            PM                  0     0     9     0     1    10
2    PostDEV_        FP                  0     0     0     0     0     1
3
4                FP 0:    SBS          Plane     Devices
5                InSv                    .          .
6    Tst_
7    Bsy_                  CTRL0         CTRL0              DEVICE
8    RTS_        DABM        .             .              0 1 2 3 4 5
9    OffL_       SCSI 0      .    EN       .    DIS       . . . – – –
10               SCSI 1      .    EN       .    DIS       . . – – – –
11
12               DEVICES:
13
14   QueryFP_
15   SwEn_
16
17
18

     TIME   12:00
```

4 Busy the members of the shadow set by typing:

> BSY DEV scsi_bus# scsi_dev#
for the master and pressing the Enter key. ↵

> BSY DEV scsi_bus# scsi_dev#
for the slave and pressing the Enter key. ↵

where

scsi_bus# is the SCSI bus number (0 or 1).
scsi_dev# is the SCSI device number.

Sample entry: **BSY DEV 0 0**
BSY DEV 1 0

Example of a MAP response:

```
FP 0 Busy DEV 0 0:  Command request has been submitted.
FP 0 Busy DEV 0 0:  Command passed.

FP 0 Busy DEV 1 0:  Command request has been submitted.
FP 0 Busy DEV 1 0:  Command passed.
```

Delete shadow set (end)

5 From the MAPCI level, enter table SHADOW by typing:

>TABLE SHADOW
and pressing the Enter key. ↵

6 List all the shadow sets by typing:

>LIST ALL
and pressing the Enter key. ↵

7 Delete the required shadow set by typing:

>DELETE nodetype nodeno setname devtype
and pressing the Enter key. ↵

where

nodetype is the node on which the shadow set resides (FP).
nodeno is the number of the node identified in field NODETYPE (0).
setname is the shadow set's name.
devtype is the device type (SCSIDK). Field DEVTYPE has the following refinements (PERMSCSI and OMEMSCSI):
PERMSCSI is the shadow set's permanent disk (master disk) and is specified by a SCSI-bus number (0 or 1) and a device number (0–5)
OMEMSCSI contains the slave shadow sets. One to four disks may be defined here.

Sample entry: **>DELETE fp 0 fp0ss0 0 0 0 1 $**

8 Verify the deletion by typing:

>LIST ALL
and pressing the Enter key. ↵

The shadow set is deleted.

Delete billing volume

At the CI prompt

1 Access the PM level of the MAP terminal by typing:

> MAPCI;MTC;PM
and pressing the Enter key. ↵

Example of a MAP response:

```
CM     MS     IOD     NET     PM     CCS     TRKS     EXT     APPL

PM                            SysB   ManB    OffL   CBsy ISTb InSv
0      QUIT      PM            0      0       9      0    1    10
2      Post
3
4
5
6
7
8
9
10
11     Disp_
12
13     Status
14     IPML
15
16
17
18



       TIME  12:00
```

2 Post the FP by typing:

> POST FP 0
and pressing the Enter key. ↵

Example of a MAP response:

```
CM     MS     IOD     NET     PM     CCS     TRKS     EXT     APPL

PM                            SysB  ManB   OffL  CBsy ISTb InSv
0      QUIT      PM            0      0      9      0    1    10
2      Post      FP            0      0      0      0    0     1
3      Plane
4      Devices  FP 0: SBS:  Plane      Devices
5               InSv
6      Tst
7      Bsy      POST:
8      RTS
9      Off
10     LoadPM
11     Disp_
12     Next
13
14     QueryPM_
15
16
17
18



       TIME  12:00
```

3 Access the DEVICES level of the MAP terminal by typing:

> DEVICES
and pressing the Enter key. ↵

Example of a MAP response:

```
CM    MS    IOD    NET    PM    CCS    TRKS    EXT    APPL

PM                        SysB  ManB   OffL  CBsy ISTb InSv
0     QUIT      PM         0     0      9     0    1    10
2     PostDEV_  FP         0     0      0     0    0     1
3     Plane
4            FP 0: SBS    Plane      Devices
5              InSv
6     Tst_
7     Bsy_            CTRL0      CTRL0          DEVICE
8     RTS_    DABM                            0 1 2 3 4 5
9     OffL_   SCSI 0          EN        DIS         - - -
10            SCSI 1          EN        DIS       - - - -
11
12           DEVICES:
13
14    QueryFP_
15    SwEn_
16
17
18



      TIME  12:00
```

4 Busy the disk on which volumes are to be deleted by typing:

> BSY DEV scsi_bus# scsi_dev#
and pressing the Enter key. ↵

where

scsi_bus# is the SCSI bus number (0 or 1).
scsi_dev# is the SCSI device number.

Sample entry: **BSY DEV 0 0**

Example of a MAP response:

```
FP 0 Busy DEV 0 0:  Command request has been submitted.
FP 0 Busy DEV 0 0:  Command passed.
```

5 Perform disk administration by typing:

> DISKADM devname nodetype nodenum
and pressing the Enter key. ↵

where

devname is the name of the device needing administration.
nodetype is the node on which the device resides.
nodenum is the node number.

Sample entry: **DISKADM dk00 fp 0**

Example of a MAP response:

```
Start up command sequence is in progress.
This may take a few minutes.
Administration of device DK00 on FP0 is now active.
DISKADM; FP0
```

CAUTION
Lost data
Verify that no data exists on the volume to be deleted. Confirming deletion causes all data on the disk to be lost. DEMOUNT volume takes it out of table DRMPOOL.

6 Delete the billing volume by typing:

> DDV volname
and pressing the Enter key. ↵

where

volname is the logical name of the volume (any 8 characters).

Sample entry: **DDV occ**

Example of a MAP response:

```
          ***** WARNING *****
Deleting volume OCC on DK00 will DESTROY the contents of
the volume.

It may CORRUPT TRMS databases on OTHER volumes.

Do you want to continue?
Please confirm (”YES” or ”NO”):
```

If	Do
Yes	step 7
No	step 8

Delete billing volume (end)

7 Confirm by typing:

> YES
and pressing the Enter key. ↵

Example of a MAP response:

```
Deletion of the volume is completed.
```

8 Abort the delete command by typing:

> No
and pressing the Enter key. ↵

Example of a MAP response:

```
DELETEVOL command is aborted
```

9 Quit DISKADM by typing:

> QUIT
and pressing the Enter key. ↵

10 Bring the disk back into service by typing:

> RTS DEV scsi_bus# scsi_dev#
and pressing the Enter key. ↵

where

scsi_bus# is the SCSI bus number.
scsi_dev# is the SCSI device number.

Example of a MAP response:

```
FP 0 RTS DEV 0 0:  Command request has been submitted
FP 0 RTS DEV 0 0:  Command passed
```

Demount DAT

At the CI prompt

1 Access the DRM level of the MAP terminal by typing:

>MAPCI;MTC;APPL;OAMAP;DRM
and pressing the Enter key. ↵

Example of a MAP response:

```
CM     MS     IOD     NET     PM     CCS     TRKS     EXT     APPL

DRM                 OAMAP   EIO
0     QUIT
2
3     Info_
4     Audit_     DRM:     AFT:     MNP:       SBS:
5     Rotate_
6     Rename_
7     Mount_
8     Demount_   DRM:
9     Reset_
10    DAT_
11    View_
12    Tcopy_
13    Monitor_
14
15
16
17
18    Copy_



      TIME  12:00
```

2 Demount DAT by typing:

>DAT EJECT
and pressing the Enter key. ↵

Demount DAT (end)

```
Sending request...
> == Request sent == >    (Please wait...)

RESPONSE READY.

   Press <CR> to continue, or enter QUIT:
```

Sample entry: **>**↵

Example of a MAP response:

```
  OPERATION SUCCESSFUL.
Tape CT02 is ejected.
```

Demount a disk volume

At the CI prompt

1 Access the DRM level of the MAP terminal by typing:

>MAPCI;MTC;APPL;OAMAP;DRM
and pressing the Enter key. ↵

Example of a MAP response:

```
CM     MS     IOD     NET     PM     CCS     TRKS     EXT     APPL

DRM                OAMAP   EIO
0      QUIT
2
3      Info_
4      Audit_      DRM:      AFT:     MNP:       SBS:
5      Rotate_
6      Rename_
7      Mount_
8      Demount_    DRM:
9      Reset_
10     DAT_
11     View_
12     Tcopy_
13     Monitor_
14
15
16
17
18     Copy_



       TIME  12:00
```

2 Demount the disk volume by typing:

>DEMOUNT stream volname
and pressing the Enter key. ↵

where

stream	is the stream datafilled in CRSMAP.
volname	is the volume name prefixed by shadow set name for shadow sets and DK for non-shadowed disks.

Sample entry: **DEMOUNT occ fp0ss0occ5**

Demount a disk volume (end)

```
Sending request...
> == Request sent == >    (Please wait...)

RESPONSE READY.

   Press <CR> to continue, or enter QUIT:
```

Sample entry: **>**↵

Example of a MAP response:

```
OPERATION SUCCESSFUL.
Volume FP0SS0OCC5 has been removed from OCCPOOL.
```

Determine what disks are in a shadow set

At the CI prompt

1 Enter the shadow utility by typing:

>SHADOWUT nodetype nodenum
and pressing the Enter key. ↵

where

nodetype is the node on which the device resides.
nodenum is the node number.

Sample entry: **SHADOWUT fp 0**

Example of a MAP response:

```
FP0 is now node of reference
Disk shadowing utility is now active
SHADOWUT; FP0
```

2 View a list of all shadow sets by typing:

>DIS
and pressing the Enter key. ↵

Example of a MAP response:

```
Defined Shadow Sets:
  1.  SS1   (InSv) 1 members
```

3 View information regarding a specific shadow set by typing:

>DIS setname
and pressing the Enter key. ↵

where

setname is the user-defined name of the shadow set.

Sample entry: **DIS ss1**

Determine what disks are in a shadow set (end)

```
Information about the shadow set #1:

Node name:                    FP0
Shadow set name:              SS1
Set definition state:         RUNNING
Set operational state:        IN SERVICE
Synchronization status:       SYNCHRONIZED
Multi-Writes:                 SERIAL
Capacity (blocks):            1244655
Transfer Length:              Optimal
Interval:                     0
=====================================================
Information about member disks:

   Name     State       SyncState   Reads  Writes
Perm DK01   INSV        Yes         246      644
     DK11   INSV        Yes         250      644
```

Note: The information shown under “Information about member disks” indicates that shadow set SS1 contains the disks DK01 and DK11.

Determine which volumes are on the file processor (end)

At the CI prompt

1 Access DISKUT by typing:

>DISKUT
and pressing the Enter key. ↵

2 List the volumes on FP0 by typing:

>LV FP0
and pressing the Enter key. ↵

Example of a MAP response:

```
Volumes found on the node FP0:
------------------------------

NAME    TYPE    TOTAL   FREE    TOTAL   OPEN    ITO
                BLOCKS  BLOCKS  FILES   FILES   FILES
-------------------------------------------------------
SS0OCC0 FTFS    1023968 777151  10      0       N/A
SS0OCC1 FTFS    1023968 786175   7      0       N/A
SS0OCC2 FTFS    1023968 834175   8      0       N/A
SS0OCC3 FTFS     989152 780895   5      0       N/A
SS0OCC4 FTFS      14304   3167   1      0       N/A
SS0FTAM FTFS      20448   9183   3      1       N/A
SS1OCC0 FTFS    1023968 828543   5      0       N/A
SS1OCC1 FTFS    1023968 828799  29      3       N/A
SS1OCC2 FTFS    1023968 750783   9      0       N/A
SS1OCC3 FTFS    1023968 838303  10      0       N/A

LARGEST
FREE SEGMENT
------------
697663
736895
722271
  3167
  9183
729983
747935
701503
706847

Total number of volumes found on node FP0: 10
```

3 Exit DISKUT by typing:

>QUIT
and pressing the Enter key. ↵

Format the disk

At the CI prompt

1 Access the PM level of the MAP terminal by typing:

> MAPCI;MTC;PM
and pressing the Enter key. ↵

Example of a MAP response:

```
CM     MS     IOD     NET     PM     CCS     TRKS     EXT     APPL

PM                            SysB   ManB    OffL   CBsy ISTb InSv
0      QUIT       PM           0      0       9      0     1    10
2      Post
3
4
5
6
7
8
9
10
11     Disp_
12
13     Status
14     IPML
15
16
17
18



       TIME  12:00
```

2 Post the FP by typing:

> POST FP 0
and pressing the Enter key. ↵

Example of a MAP response:

```
CM     MS     IOD     NET     PM     CCS     TRKS     EXT     APPL

PM                            SysB  ManB   OffL  CBsy ISTb InSv
0      QUIT      PM            0     0      9     0    1    10
2      Post      FP            0     0      0     0    0     1
3      Plane
4      Devices  FP 0: SBS:  Plane      Devices
5               InSv
6      Tst
7      Bsy      POST:
8      RTS
9      Off
10     LoadPM
11     Disp_
12     Next
13
14     QueryPM_
15
16
17
18



       TIME  12:00
```

3 Access the DEVICES level of the MAP terminal by typing:

> DEVICES
and pressing the Enter key. ↵

Example of a MAP response:

```
CM     MS     IOD     NET     PM     CCS     TRKS     EXT     APPL

PM                            SysB  ManB   OffL  CBsy ISTb InSv
0      QUIT      PM            0     0      9     0    1    10
2      PostDEV_  FP            0     0      0     0    0     1
3      Plane
4             FP 0: SBS     Plane      Devices
5               InSv
6      Tst_
7      Bsy_            CTRL0      CTRL0          DEVICE
8      RTS_    DABM                            0 1 2 3 4 5
9      OffL_   SCSI 0          EN        DIS         - - -
10             SCSI 1          EN        DIS       - - - -
11
12            DEVICES:
13
14     QueryFP_
15     SwEn_
16
17
18



       TIME  12:00
```

4 Busy the disk on which volumes are to be formatted/created by typing:

> BSY DEV scsi_bus# scsi_dev#
and pressing the Enter key. ↵

where

scsi_bus# is the SCSI bus number (0 or 1).
scsi_dev# is the SCSI device number.

Sample entry: **BSY DEV 0 0**

Example of a MAP response:

```
FP 0 Busy DEV 0 0:  Command request has been submitted.
FP 0 Busy DEV 0 0:  Command passed.
```

5 Perform disk administration by typing:

> DISKADM devname nodetype nodenum
and pressing the Enter key. ↵

Format the disk (end)

devname is the name of the device needing administration.
nodetype is the node on which the device resides.
nodenum is the node number.

Sample entry: **DISKADM dk00 fp 0**

Example of a MAP response:

```
Start up command sequence is in progress.
This may take a few minutes.
Administration of device DK00 on FP0 is now active.
DISKADM; FP0
```

6 Erase all volumes and files on the disk by typing:

> FD QUICK FORCE
and pressing the Enter key. ↵

Example of a MAP response:

```
           ***** WARNING *****
Formatting of DK00
Will destroy the contents of the disk.

The formatting will:
  allocate 3 spare or alternate sectors per track,
  allocate 16 spare or alternate tracks per disk,
  use the G defect list,
  assign QUICK as the name for the disk,
  perform quick format,
  include force option.

Do you want to continue?
Please confirm (”YES” or ”NO”):
```

7 Confirm by typing:

> YES
and pressing the Enter key. ↵

Example of a MAP response:

```
Initializing the system data structures on the disk.
Formatting and initialization of the disk is completed.
```

8 See “Create billing volumes” procedure in this section to allocate the disk.

Manually rotate a DRM-managed file

At the CI prompt

1 Access the DRM level of the MAP terminal by typing:

>MAPCI;MTC;APPL;OAMAP;DRM
and pressing the Enter key. ↵

Example of a MAP response:

```
CM     MS     IOD     NET     PM     CCS     TRKS     EXT     APPL

DRM                 OAMAP   EIO
0      QUIT
2
3      Info_
4      Audit_      DRM:       AFT:     MNP:        SBS:
5      Rotate_
6      Rename_
7      Mount_
8      Demount_    DRM:
9      Reset_
10     DAT_
11     View_
12     Tcopy_
13     Monitor_
14
15
16
17
18     Copy_



       TIME  12:00
```

2 Rotate to a new file by typing:

>ROTATE stream
and pressing the Enter key. ↵

where

stream is the stream datafilled in table CRSMAP.

Manually rotate a DRM-managed file (end)

```
Downloading parameters...
Done.
Sending request...
> == Request sent == >   (Please wait...)

Waiting for OCC response...
(30 sec timeout...)
Done.
```

Monitor the state of the active DRM file

At the CI prompt

1 Access the DRM level of the MAP terminal by typing:

>MAPCI;MTC;APPL;OAMAP;DRM
and pressing the Enter key. ↵

Example of a MAP response:

```
CM     MS     IOD     NET     PM     CCS     TRKS     EXT     APPL

DRM                  OAMAP   EIO
0     QUIT
2
3     Info_
4     Audit_      DRM:      AFT:      MNP:       SBS:
5     Rotate_
6     Rename_
7     Mount_
8     Demount_    DRM:
9     Reset_
10    DAT_
11    View_
12    Tcopy_
13    Monitor_
14
15
16
17
18    Copy_



      TIME  12:00
```

2 Monitor the state of the active DRM file by typing:

>MONITOR stream
and pressing the Enter key. ↵

where

stream is the name of the stream defined in table CRSMAP.

Sample entry: **MONITOR occ**

Monitor the state of the active DRM file (end)

```
CM    MS    IOD    NET    PM    CCS    TRKS    EXT    APPL

DRM                 OAMAP   EIO
0 QUIT
2
3 Info_
4 Audit_      DRM:      AFT:     MNP:       SBS:
5 Rotate_
6 Rename_
7 Mount_      OCC:   ACTN FP0      VOL   :FP0/FP0SS0OCC
8 Demount_        DRM: BLOCKS  BLKSZ T RECORDS FLSZ-KB ER V#
9 Reset_ AN9510080300000CC  1   2048  F     2        2  0 0
10 DAT_       MONITOR
11 View_
12 Tcopy_
13 Monitor_
14
15
16
17
18 Copy_



      TIME  12:00
```

Mount DAT

At the CI prompt

1 Access the DRM level of the MAP terminal by typing:

>MAPCI;MTC;APPL;OAMAP;DRM
and pressing the Enter key. ↵

Example of a MAP response:

```
CM     MS     IOD     NET     PM     CCS     TRKS     EXT     APPL

DRM                 OAMAP   EIO
0      QUIT
2
3      Info_
4      Audit_      DRM:      AFT:     MNP:        SBS:
5      Rotate_
6      Rename_
7      Mount_
8      Demount_    DRM:
9      Reset_
10     DAT_
11     View_
12     Tcopy_
13     Monitor_
14
15
16
17
18     Copy_



       TIME  12:00
```

2 Mount DAT by typing:

DAT MOUNT dat_drive nodename tapeformat [INIT]
and pressing the Enter key. ↵

where

dat_drive	is the name of the DAT tape drive to be mounted.
nodename	is the node where the DAT drive resides.
tapeformat	is the format the DAT will use (NT).

Mount DAT (end)

tape and writes a new DRM volume identifier onto the tape. A subsequent WRITE command will begin at the beginning of the tape. INIT must be specified when mounting a DAT tape for the first time.

Sample entry: **>DAT MOUNT ct11 fp0 INIT**

Example of a MAP response:

```
Sending request . . .
> == Request sent == >     (Please wait . . .)

RESPONSE READY.

 Press <CR> to continue, or enter QUIT:
```

Sample entry: **>**↵

Example of a MAP response:

```
  OPERATION SUCCESSFUL.
Tape CT11 is mounted.
```

Mount disk volume

At the CI prompt

1 Access the DRM level of the MAP terminal by typing:

>MAPCI;MTC;APPL;OAMAP;DRM
and pressing the Enter key. ↵

Example of a MAP response:

```
CM     MS     IOD     NET     PM     CCS     TRKS     EXT     APPL

DRM                 OAMAP   EIO
0      QUIT
2
3      Info_
4      Audit_      DRM:      AFT:     MNP:       SBS:
5      Rotate_
6      Rename_
7      Mount_
8      Demount_    DRM:
9      Reset_
10     DAT_
11     View_
12     Tcopy_
13     Monitor_
14
15
16
17
18     Copy_



       TIME  12:00
```

CAUTION
Lost data
Do not mount any volume ending in the letters “FTAM” into DRM.

2 Mount the disk volume by typing:

>MOUNT stream volname nodename [priority]
and pressing the Enter key. ↵

where

stream	is the name of the stream datafilled in table CRSMAP.
volname	is the name of the volume prefixed by shadow set name for shadow sets and DK for non-shadowed disks.
nodename	is the name of the node that contains, or will contain, the specified volume.
priority	is an optional parameter that assigns a usage priority to the volume. 1 is the highest priority level and 18 is the lowest.

Sample entry: **MOUNT occ ss1occ5 fp0**

Example of a MAP response:

```
Sending request...
> == Request sent == >   (Please wait...)

RESPONSE READY.

   Press <CR> to continue, or enter QUIT:
```

Sample entry: **>↵**

Example of a MAP response:

```
OPERATION SUCCESSFUL

Volume SS1OCC5 has been allocated to OCCPOOL.  The MOUNT
may take a few seconds to complete.
```

3 Discontinue the mounting process by typing:

If	Then
the operation successful response is received	the procedure is complete
an error message is received	continue to step 4

Note: Following is an example of an error message:

```
Volume SS1OCCX is not available.
Reason:  Device Unavailable

Perform Mount operation?  (Yes/No, or Quit)
```

Notice the X at the end of the volume name in the error message. This should be a number. The error response was received due to this spelling error.

Mount disk volume (end)

4 Respond to the error message by typing:

>No

and pressing the Enter key. ↵

Obtain file information

At the CI prompt

1 Access the DRM level of the MAP terminal by typing:

>MAPCI;MTC;APPL;OAMAP;DRM
and pressing the Enter key. ↵

Example of a MAP response:

```
CM     MS     IOD     NET     PM     CCS     TRKS     EXT     APPL

DRM                 OAMAP   EIO
0      QUIT
2
3      Info_
4      Audit_      DRM:       AFT:     MNP:        SBS:
5      Rotate_
6      Rename_
7      Mount_
8      Demount_    DRM:
9      Reset_
10     DAT_
11     View_
12     Tcopy_
13     Monitor_
14
15
16
17
18     Copy_



       TIME  12:00
```

2 The following syntax is used to view file information:

>INFO stream infotype [destnode]
where

stream is the name of the stream datafilled in table CRSMAP.
infotype is the type of information to obtain and may be one of the following:

NODES	lists all DRM nodes used by the stream.
REG	lists registration information for each DRM node where the stream is registered.
SCHED	shows the audit schedules.

ROT	shows the rotate schedule.
VOL	shows volume information.
VOLDIAG	shows trouble diagnostics information for all mounted volumes.
VOLERR	shows last error detected on each mounted volume.
VOLFIL	shows mounted volumes and their files.
FIL	requests information of files matching the supplied parameters.

destnode is one of the nodes listed by the NODES subcommand. This is an optional parameter to route the command to a specific node destination.

3 To obtain file information, choose one of the following and proceed, using the indicated command

To see this type of information,	Use this command	Sample entry
DRM nodes	INFO stream NODES	INFO occ NODES
application registration	INFO stream REG [destnode]	INFO occ REG
application audit schedules and next scheduled rotate time	INFO stream SCHED [destnode]	INFO occ SCHED fp0
application rotate information	INFO stream ROT [destnode]	INFO occ ROT fp0
application’s mounted volumes, including status information	INFO stream VOL [volnode volume ACTV ALL] [destnode]	INFO occ VOL fp0
application’s mounted volumes, including trouble diagnostics	INFO stream VOLDIAG [volnode volume ACTV ALL] [destnode]	INFO occ VOLDIAG ACTV
application’s mounted volumes with a brief, textual description of the last error detected on each volume	INFO stream VOLERR [volnode volume ACTV ALL] [destnode]	INFO occ VOLERR ACTV
application’s mounted volumes, along with its files	INFO stream VOLFIL [volnode volume ACTV ALL] [filetmpl fstates ALL] [destnode]	INFO occ VOLFIL 0 u fp0
application’s files and file information	INFO stream FIL [volnode volume ACTV ALL] [filetmpl fstates ALL] [destnode]	INFO occ FIL fp0 cm

—end—

4 View DRM node information by typing:

>INFO stream NODES
and pressing the Enter key. ↵

where

stream	is the name of the stream datafilled in table CRSMAP.
NODES	lists all DRM nodes used by the stream.

Sample entry: **INFO occ NODES**

Example of a MAP response:

```
OCC usage of DRM-nodes:     92/01/26 14:42:57

---------------------------------------------------
Nodes:  Status:     Fr MB:      Volume References:
------  -------     ------      ------------------
CM      Idle        0
FP0     ACTIVE      1152        0,1,2,3,4,5
                    ----
   Total Fr MB=     1152
```

DRM node information is obtained and procedure is complete.

5 View registration information by typing:

>INFO stream REG [destnode]
and pressing the Enter key. ↵

where

stream	is the name of the stream datafilled in table CRSMAP.
REG	is an optional subcommand that lists registration information for each DRM node where the stream is registered.
destnode	is one of the nodes listed by the NODES subcommand. This is an optional parameter to route the command to a specific node destination.

Sample entry: **INFO occ REG**

Example of a MAP response:

```
OCC Registration Info:      92/01/26 14:42:57
---------------------------------------------------
Node    Last Reg            Reg     Block       Allow
Name    (date-time)         Num     Size(B)/T   A->P
----    -----------         ---     ---------   -------
CM      92/01/24  09:32     1       2048/F      NO
FP0     92/01/24  10:16     1       2048/F      NO
```

Registration information is obtained and procedure is complete.

6 View schedule information by typing:

>INFO stream SCHED [destnode]
and pressing the Enter key. ↵

where

stream	is the name of the stream datafilled in table CRSMAP.
SCHED	shows the audit schedules.
destnode	is one of the nodes listed by the NODES subcommand. This is an optional parameter to route the command to a specific node destination.

Sample entry: **INFO occ SCHED fp0**

Example of a MAP response:

```
OCC Schedule; –>FP0 (ACTN): 92/01/26 14:32:57
-------------------------------------------------------
AUDIT TYPE           AUDIT HISTORY              INTERVAL
               BEGIN/END           BEGIN/END
----------------------------------------------------------
SPACE       19:29:03/19:29:03  19:39:03/19:39:03  00:10:00
VOLUME      19:28:42/19:59:03  19:59:03/19:59:58  00:30:00
APPLICATION 18:54:41/18:54:41  19:54:41/19:54:41  01:00:00
DRMFDIR     16:26:21/16:36:32  03:00:00/03:10:00  10:23:28

NEXT SCHEDULED ROTATE:  94/09/16 03:00:00
```

Schedule information is obtained and procedure is complete.

7 View rotate information by typing:

>INFO stream ROT [destnode]
and pressing the Enter key. ↵

where

stream	is the name of the stream datafilled in table CRSMAP.
ROT	shows the rotate schedule.
destnode	is one of the nodes listed by the NODES subcommand. This is an optional parameter to route the command to a specific node destination.

Sample entry: **INFO occ ROT fp0**

Example of a MAP response:

```
OCC Rotate info; –>FP0 (ACTN):92/01/26 14:32:57
---------------------------------------------------
Next scheduled ROTATE..........92/01/26 15:00:00
Volume chosen for next file....    0
Last ROTATE time...............92/01/26 14:00:00
   reason................    Schedule
Close file state (CLOSTATE)...     U

OCC Ordered Rotate Volume List; –>FP0 (ACTN): 92/01/26 14:32:57
          ROTATES: (since 92/01/26 09:32:00)
V#  STATUS NODE PR  SCH  MAN  FSZ  SPA  ERR  APP SHD
--  ----------- --  ---  ---  ---  ---  ---  --- ---
1   OK     FP0  1    12    0    5    2    0    1   0
0   OK     FP0  1    13    0    4    3    0    0   0
3   OK     FP0  2     0    0    0    0    0    0   0
```

Rotate information is obtained and procedure is complete.

8 View mounted volume information by typing:

>INFO stream VOL [volnode volume ACTV ALL] [destnode]
and pressing the Enter key. ↵

where

stream is the name of the stream datafilled in table CRSMAP.

VOL shows the volume information and must precede one of the following:

- **volnode** is a node where volumes reside. This parameter is optional, but must always follow the subcommand parameter when used. It is used to refer to all volumes in the application's pool that reside on the node.
- **volume** is either a volume name or volume number from table DRMPOOL. This parameter is optional, but must always follow the subcommand parameter.
- **ACTV** may replace any volume parameter to refer to the volume containing the application's active file. This parameter is optional, but must always follow the subcommand parameter when used.
- **ALL** requests registration, audit, volume, file and rotate information.

destnode is one of the nodes listed by the NODES subcommand. This is an optional parameter to route the command to a specific node destination.

Sample entry: **INFO occ VOL fp0**

Example of a MAP response:

```
OCC  FP0 volumes; –>FP0 (ACTN):                  95/10/17 20:08:51
_________________________________________________________________
V#  Path                              PR  STATE   SZ(M) FR(M)  FILES
--  ------------------------------ --  ------  ----- -----  -----
 0  :FP0/FP0SS0OCC                     1  OK        150    65    519
```

Mounted volume information is obtained and procedure is complete.

9 View mounted volume trouble diagnostics information by typing:

>INFO stream VOLDIAG [volnode volume ACTV ALL] [destnode]
and pressing the Enter key. ↵

where

stream is the name of the stream datafilled in table CRSMAP.

VOLDIAG shows trouble diagnostics information for all mounted volumes and must precede one of the following:

- **volnode** is a node where volumes reside. This parameter is optional, but must always follow the subcommand parameter when used. It is used to refer to all volumes in the application's pool that reside on the node.
- **volume** is either a volume name or volume number from table DRMPOOL. This parameter is optional, but must always follow the subcommand parameter.
- **ACTV** may replace any volume parameter to refer to the volume containing the application's active file. This parameter is optional, but must always follow the subcommand parameter when used.
- **ALL** requests registration, audit, volume, file and rotate information.

destnode is one of the nodes listed by the NODES subcommand. This is an optional parameter to route the command to a specific node destination.

Sample entry: **INFO occ VOLDIAG ACTV**

Example of a MAP response:

```
OCC FP0SS0OCC Diagnostics; –>FP0 (ACTN):     92/11/21 02:31:05
---------------------------|–Counts from 92/11/21/ 01:09:53–|
   -------SPACE------------|---VOLUME ERR----  -----FILE ERR----|
V# FR% A% U% P% R% N% STATE SP AC FL IO FS IF  CR OP WR CL RN DL
-- --- -- -- -- -- -- ----- -- -- -- -- -- --  -- -- -- -- -- --
 0  95  0  2  0  0  3 OK     0  6  0  0  0  4   0  0  0  0  0  0

OCC Error text for FP0SS0OCC; –>FP0 (ACTN): 92/11/21 02:31:05 10
----------------------------------------------------------------
V# Last error detected, followed by current status (–>):
-- -------------------------------------------------------------
 0  Non-existent device
          Occurred 92/11/21/ 01:15:15  on Access
           –> OK   92/11/21  01:19:56  by Audit
```

Mounted volume trouble diagnostic information is obtained and procedure is complete.

10 View mounted volume error information by typing:

>INFO stream VOLERR [volnode volume ACTV ALL] [destnode]
and pressing the Enter key. ↵

where

stream is the name of the stream datafilled in table CRSMAP.

VOLERR shows the last error detected on each mounted volume and must precede one of the following:

- **volnode** is a node where volumes reside. This parameter is optional, but must always follow the subcommand parameter when used. It is used to refer to all volumes in the application's pool that reside on the node.
- **volume** is either a volume name or volume number from table DRMPOOL. This parameter is optional, but must always follow the subcommand parameter.
- **ACTV** may replace any volume parameter to refer to the volume containing the application's active file. This parameter is optional, but must always follow the subcommand parameter when used.
- **ALL** requests registration, audit, volume, file and rotate information.

destnode is one of the nodes listed by the NODES subcommand. This is an optional parameter to route the command to a specific node destination.

Sample entry: **INFO occ VOLERR ACTV**

Example of a MAP response:

```
OCC Error text for ALL Vols; ->FP0:           92/11/21  02:31:05 10
-------------------------------------------------------------------
V# Last error detected, followed by current status (->):
-- ----------------------------------------------------------------
 0 Non-existent device
   Occurred   92/11/20 17:12:25 on Access
   -> OK 92/11/20 17:17:56 by Audit
 2 Non-existent device
   Occurred   92/11/20 17:13:02 on Access
   ->SHUTDN   92/11/21 01:19:56 by ManBsy
 1 Input/Output Error
   Occurred   92/11/21 01:15:15 on Write-AN921121010000OM
   ->DOWN 92/11/21/01:15:15 by File Op
```

Mounted volume error information is obtained and procedure is complete.

11 View mounted volume file information by typing:

>INFO stream VOLFIL [volnode volume ACTV ALL] [filetmpl fstates ALL] [destnode]
and pressing the Enter key. ↵

where

stream is the name of the stream datafilled in table CRSMAP.

VOLFIL shows the mounted volumes and their files and must precede one of the following:

- **volnode** is a node where volumes reside. This parameter is optional, but must always follow the subcommand parameter when used. It is used to refer to all volumes in the application's pool that reside on the node.
- **volume** is either a volume name or volume number from table DRMPOOL. This parameter is optional, but must always follow the subcommand parameter.
- **ACTV** may replace any volume parameter to refer to the volume containing the application's active file. This parameter is optional, but must always follow the subcommand parameter when used.
- **ALL** requests registration, audit, volume, file and rotate information.

filetmpl is the first character of the file name to be queried, followed by as many characters in the name as desired to identify a single file, or group of files. It refers to all files owned by the application whose names match this file name template. A $ may be used as a wildcard character to substitute any number of characters within the file template.

fstates specifies the files to be queried by providing a list of file states. It may contain any combination of the states A, U, P, and R.

ALL specifies all volumes.

destnode is one of the nodes listed by the NODES subcommand. This is an optional parameter to route the command to a specific node destination.

Sample entry: **INFO occ VOLFIL 0 u fp0**

Example of a MAP response:

```
OCC Vol :FP0/DK00OCC0 info; –>FP0       92/01/26 14:32:57
    Requested files: U   Counts: U = 2
----------------------------------------------------------------
V# PR SZ(M) FR(M) FR% A% U% P% R% N%       STATE  ER  FILES
-- -- ----- ----- --- -- -- -- -- --       -----  --  -----
 0  1  600   552  92  1   3  4  0  0         OK    0      3

FILE NAME           BLOCKS BLKSZ T RECORDS  FLSZ-KB  ER  V# P R
-------------------- ------ ----- - ------  -------  --  -- - -
UN9010171400050CC     2509  2048 F  40144    5018    0   0  A CL
UN9010171200030CC     8935  2048 F  142960   17870   0   0  A CL
```

Mounted volume file information is obtained and procedure is complete.

12 View file information by typing:

>INFO stream FIL [volnode volume ACTV ALL] [filetmpl fstates ALL] [destnode]
and pressing the Enter key. ↵

where

stream is the ID of an application in table DRMAPPL. The default requests to display all application IDs known to DRM.

VOLFIL shows the mounted volumes and their files and must precede one of the following:

- **volnode** is a node where volumes reside. This parameter is optional, but must always follow the subcommand parameter when used. It is used to refer to all volumes in the application's pool that reside on the node.
- **volume** is either a volume name or volume number from table DRMPOOL. This parameter is optional, but must always follow the subcommand parameter.
- **ACTV** may replace any volume parameter to refer to the volume containing the application's active file. This parameter is optional, but must always follow the subcommand parameter when used.
- **ALL** requests registration, audit, volume, file and rotate information.

filetmpl is the first character of the file name to be queried, followed by as many characters in the name as desired to identify a single file, or group of files. It refers to all files owned by the application whose names match this file name template. A $ may be used as a wildcard character to substitute any number of characters within the file template.

fstates specifies the files to be queried by providing a list of file states. It may contain any combination of the states A, U, P, and R.

ALL specifies all volumes.

destnode is one of the nodes listed by the NODES subcommand. This is an optional parameter to route the command to a specific node destination.

Sample entry: **INFO occ FIL fp0 cm**

Obtain file information (end)

```
OCC Files on FP0 volumes; –>CM (ACTN):  92/01/26 14:32:57
----------------------------------------------------------------
    Requested files: A,U,P,R  Counts: A=0, U=2, P=1, R=2
----------------------------------------------------------------
FILE NAME           BLOCKS  BLKSZ  T  RECORDS  FLSZ-KB ER   V#  P R
--------------------------  -----  -  -------  ------- --   --  - -
UN9010171400050CC   2509    2048   F    40144  5018     0   0   A CL
UN9010171200030CC   8935    2048   F   142960  17870    0   0   A CL
PN9201161900000CC   3778    2048   F    60448  7556     0   1   U PO
RN9201110100000CC   9834    2048   F   157344  19668    0   1   P RN
RN9201110200000CC   10987   2048   F   175792  21974    0   0   P RN
```

File information is obtained and procedure is complete.

Perform a DRM audit

ATTENTION
This process is CPU intensive. Use sparingly.

At the CI prompt

1 Access the DRM level of the MAP terminal by typing:

>MAPCI;MTC;APPL;OAMAP;DRM
and pressing the Enter key. ↵

Example of a MAP response:

```
CM     MS     IOD     NET     PM     CCS     TRKS     EXT     APPL

DRM                 OAMAP   EIO
0      QUIT
2
3      Info_
4      Audit_      DRM:      AFT:     MNP:       SBS:
5      Rotate_
6      Rename_
7      Mount_
8      Demount_    DRM:
9      Reset_
10     DAT_
11     View_
12     Tcopy_
13     Monitor_
14
15
16
17
18     Copy_



       TIME  12:00
```

2

If this audit is required,	Use this command	Sample entry
all audits	AUDIT stream ALL	AUDIT occ ALL
recording audit	AUDIT applid REC [ACTN] [nodename]	AUDIT occ REC fp0
space audit	AUDIT applid SPACE [ACTN] [nodename]	AUDIT occ SPACE fp0
volume audit	AUDIT applid VOL volume [ACTN] [nodename]	AUDIT occ VOL 0 fp0
application audit	AUDIT applid APPL [ACTN] [nodename]	AUDIT occ APPL fp0
DRMFDIR audit	AUDIT applid FDIR [ACTN] [nodename]	AUDIT occ FDIR fp0
	—end—	

3 Initiate all DRM audits by typing:

>AUDIT stream ALL
and pressing the Enter key. ↵

where

stream is the name of the stream defined in table CRSMAP.

Sample entry: **AUDIT occ ALL**

Example of a MAP response:

```
> == Request sent == >      (Please wait...)
RESPONSE READY
Press <CR> to continue, or enter QUIT:
```

Sample entry: **>**↵

Example of a MAP response:

```
AUDIT OCC APPL Completed on FP0
-----------| Counts from 93/02/22 14:50:01---|-----
        ---ERROR CONDITIONS---     REQUESTS PEND
STATUS  TBL SPA REQ FER VER SWE    RTR ROT AUD
------  --- --- --- --- --- ---    --- --- ---
ACTIVE   0   0   1   0   0   0       0   0   1

Press <CR> to continue, or enter QUIT:
```

Sample entry: **>**↵

Example of a MAP response:

```
AUDIT OCC REC Completed on FP0
---------------------------------------------------
REC     Records over last capture interval..... 0
AUDIT   Avg records per hour.................. 0
        Avg volume space used per minute...0 KB/min
        Avg volume space used per hour.....0 KB/hr
        Last capture interval.......00:03:20
        Begin–end of last capture
        interval...15:44:05–15:47:26
        Next scheduled capture time..15:54:06
        Last manual capture time...00/00/00  00:00:00

   Press <CR> to continue, or enter QUIT:
```

Sample entry: **>**↵

Example of a MAP response:

```
AUDIT OCC VOL ALL Completed on FP0
--------------------| Counts from 93/02/22 14:50:01
-----SPACE-----      --VOLUME ERR--- ---FILE ERR---
V FR A U P R N STATE SP AC FL IO FS IF CR OP WR CL RN DL
- -- - - - - - ----- -- -- -- -- -- -- -- -- -- -- -- --
0 97 0 0 0 0 3   OK   0  0  0  0  0  0  0  0  0  0  0  0
1 97 0 2 0 0 1   OK   0  0  0  0  0  0  0  0  0  0  0  0

Press <CR> to continue, or enter QUIT:
```

Sample entry: **>**↵

Example of a MAP response:

```
AUDIT OCC SPACE Completed on FO0
---------------------------------------------------
InSv    Space recovered during audit..  0 MB    0%
POOL    Volume chosen for next file...  1
SPACE   Total space...................  600 MB
    Free space....................      582 MB  97%
    Active space..(A files).......      0 MB    0%
    Unprocessed space..(U files)..      7 MB    1%
    Processed space..(P files)....      0 MB    0%
    Removed space..(R files)......      0 MB    0%
    Non–usable space..(Non–DRM)...      11 MB   2%
    Available recording space.....      582 MB  97%
       (free + P)

  Press <CR> to continue, or enter QUIT:
```

Sample entry: **>**↵

Example of a MAP response:

```
AUDIT OCC FDIR Completed on FP0
---------------------------------------------------
FILES        Number of files................ 6
             Active files................... 1
             Unprocessed files.............. 5
             Processed files................ 0
             Removed files.................. 0
```

All DRM audits are performed and the procedure is complete.

4 Initiate a recording audit by typing:

>AUDIT applid REC [ACTN] [nodename]
and pressing the Enter key. ↵

where

applid is the ID (key) of an application in table DRMAPPL.
ACTN is an optional parameter indicating the active node is to be audited.
nodename is an optional parameter indicating which node is to be audited.

Sample entry: **AUDIT occ REC fp0**

Example of a MAP response:

```
Sending request to FP0...
Request sent.  (Please wait...)

RESPONSE READY

   Press <CR> to continue, or enter QUIT:
```

Sample entry: **>**↵

Example of a MAP response:

```
AUDIT OCC REC Completed on FP0
---------------------------------------------------
REC      Records over last capture interval..... 0
AUDIT    Avg records per hour................... 0
         Avg volume space used per minute...0 KB/min
         Avg volume space used per hour.....0 KB/hr
         Last capture interval.......00:03:20
         Begin–end of last capture
         interval...15:44:05–15:47:26
         Next scheduled capture time..15:54:06
         Last manual capture time...00/00/00  00:00:00
```

The recording audit is performed and the procedure is complete.

5 Initiate a space audit by typing:

>AUDIT applid SPACE [ACTN] [nodename]
and pressing the Enter key. ↵

where

applid	is the ID (key) of an application in table DRMAPPL.
ACTN	is an optional parameter indicating the active node is to be audited.
nodename	is an optional parameter indicating which node is to be audited.

Sample entry: **AUDIT occ SPACE fp0**

Example of a MAP response:

```
Sending request to FP0...
Request sent.  (Please wait...)

RESPONSE READY

   Press <CR> to continue, or enter QUIT:
```

Sample entry: **>**↵

Example of a MAP response:

```
AUDIT OCC SPACE Completed on FO0
---------------------------------------------------
InSv    Space recovered during audit..0 MB      0%
POOL    Volume chosen for next file...1
SPACE   Total space...................600 MB
        Free space....................582 MB    97%
        Active space..(A files).......0 MB      0%
        Unprocessed space..(U files)..7 MB      1%
        Processed space..(P files)....0 MB      0%
        Removed space..(R files)......0 MB      0%
        Non–usable space..(Non–DRM)...11 MB     2%
        Available recording space.....582 MB    97%
            (free + P)
```

The space audit is performed and the procedure is complete.

6 Initiate a volume audit by typing:

>AUDIT applid VOL volume [ACTN] [nodename]
and pressing the Enter key. ↵

where

applid	is the ID (key) of an application in table DRMAPPL.
ACTN	is an optional parameter indicating the active node is to be audited.
nodename	is an optional parameter indicating which node is to be audited.

Sample entry: **AUDIT occ VOL 0 fp0**

Example of a MAP response:

```
Sending request to FP0...
Request sent.  (Please wait...)

RESPONSE READY

   Press <CR> to continue, or enter QUIT:
```

Sample entry: **>↵**

Example of a MAP response:

```
AUDIT OCC VOL FP0SS0OCC Completed on FP0
---------------------------------------------------
V  SZ(M)    FR% A% U% P% R% N%  STATE   ER  FILES
-  -----    --- -- -- -- -- --  -----   --  -----
0  200      97   0  0  0  0  3    OK     0     4
```

The volume audit is performed and the procedure is complete.

7 Initiate an application audit by typing:

>AUDIT applid APPL [ACTN] [nodename]
and pressing the Enter key. ↵

where

applid	is the ID (key) of an application in table DRMAPPL.
ACTN	is an optional parameter indicating the active node is to be audited.
nodename	is an optional parameter indicating which node is to be audited.

Sample entry: **AUDIT occ APPL fp0**

Example of a MAP response:

```
Sending request to FP0...
Request sent.  (Please wait...)

RESPONSE READY

   Press <CR> to continue, or enter QUIT:
```

Sample entry: **>↵**

Perform a DRM audit (end)

```
AUDIT OCC APPL Completed on FP0
-----------| Counts from 93/02/22 14:50:01---|-----
        ---ERROR CONDITIONS---     REQUESTS PEND
STATUS  TBL SPA REQ FER VER SWE    RTR ROT AUD
------  --- --- --- --- --- ---    --- --- ---
ACTIVE   0   0   1   0   0   0       0   0   1
```

The application audit is performed and the procedure is complete.

8 Initiate an FDIR audit by typing:

>AUDIT applid FDIR [ACTN] [nodename]
and pressing the Enter key. ↵

where

applid	is the ID (key) of an application in table DRMAPPL.
ACTN	is an optional parameter indicating the active node is to be audited.
nodename	is an optional parameter indicating which node is to be audited.

Sample entry: **AUDIT occ FDIR fp0**

Example of a MAP response:

```
Sending request to FP0...
Request sent.  (Please wait...)

RESPONSE READY

   Press <CR> to continue, or enter QUIT:
```

Sample entry: **>**↵

Example of a MAP response:

```
AUDIT OCC FDIR Completed on FP0
---------------------------------------------------
FILES       Number of files................ 6
            Active files................... 1
            Unprocessed files.............. 5
            Processed files................ 0
            Removed files.................. 0
```

The FDIR audit is performed and the procedure is complete.

Rename a DRM-managed file

At the CI prompt

1 Access the DRM level of the MAP terminal by typing:

>MAPCI;MTC;APPL;OAMAP;DRM
and pressing the Enter key. ↵

Example of a MAP response:

```
CM     MS     IOD     NET     PM     CCS     TRKS     EXT     APPL

DRM                 OAMAP   EIO
0      QUIT
2
3      Info_
4      Audit_      DRM:      AFT:     MNP:       SBS:
5      Rotate_
6      Rename_
7      Mount_
8      Demount_    DRM:
9      Reset_
10     DAT_
11     View_
12     Tcopy_
13     Monitor_
14
15
16
17
18     Copy_



       TIME  12:00
```

2 Rename the file to be removed by typing:

>RENAME stream currenttype newtype
and pressing the Enter key. ↵

where

stream is the name of the stream datafilled in table CRSMAP.
currenttype is the file's status (U – unprocessed, P – processed, or R – removed).
newtype is the status the file is renamed to (U – unprocessed, P – processed, or R – removed).

Rename a DRM-managed file (end)

Example of a MAP response:

```
> == Request sent == >  (Please wait...)
RESPONSE READY.

   Press <CR> to continue, or enter QUIT:
```

Sample entry: **>↵**

Example of a MAP response:

```
These files match the supplied filename template:
1. UN920803101900OCC    2.  UN920803123600OCC
3. UN920804030000OCC    4.  UN920804083200OCC

Select a
```

Sample entry: **> 4**

Example of a MAP response:

```
Sending request...
> == Request sent == >    (Please wait...)

RESPONSE READY.

   Press <CR> to continue, or enter QUIT:
```

Sample entry: **>↵**

Example of a MAP response:

```
   OPERATION SUCCESSFUL.
UN920804083200OCC has been renamed:
PN920804083200OCC
```

Reset volume counters

At the CI prompt

1 Access the DRM level of the MAP terminal by typing:

>MAPCI;MTC;APPL;OAMAP;DRM
and pressing the Enter key. ↵

Example of a MAP response:

```
CM     MS     IOD     NET     PM     CCS     TRKS     EXT     APPL

DRM                 OAMAP   EIO
0      QUIT
2
3      Info_
4      Audit_      DRM:      AFT:     MNP:       SBS:
5      Rotate_
6      Rename_
7      Mount_
8      Demount_    DRM:
9      Reset_
10     DAT_
11     View_
12     Tcopy_
13     Monitor_
14
15
16
17
18     Copy_



       TIME  12:00
```

2 Reset volume errors by typing:

>RESET stream counter
and pressing the Enter key. ↵

where

stream is the name of the stream defined in table CRSMAP.
counter is the type of counter (ERR – error or ROT – rotation).

Sample entry: **RESET occ err**

Reset volume counters (end)

```
OPERATION SUCCESSFUL.

RESET OCC  ERR completed on FP0
```

Restore files from DAT

At the CI prompt

1 Mount DAT by following the procedure entitled, *Mount DAT*.

Note: Each file must be restored from DAT separately.

2 Restore files from DAT by typing:

>DAT RESTORE filetmpl [newstate]
and pressing the Enter key. ↵

where

filetmpl is U for Unprocessed, P for Processed, or R for Removed.
newstate is an optional parameter that forces P-files to be restored as R-files. The only valid value is RE (for retained state). Only P-files can be changed to R-files.

Sample entry: **>DAT RESTORE u**

Example of a MAP response:

```
> == Request sent == >  (Please wait...)
RESPONSE READY.
```

Example of a MAP response:

```
These files match the supplied filename template:
1. UB9205310958000CC    2.  UB9208017030000OCC

   Press <CR> to continue, or enter QUIT:
```

Sample entry: **>**↵

Example of a MAP response:

```
Select a file name
```

Sample entry: **> 1**

Example of a MAP response:

```
Sending request...
> == Request sent == >    (Please wait...)

RESPONSE READY.

   Press <CR> to continue, or enter QUIT:
```

Sample entry: **>**↵

Example of a MAP response:

```
FILE UB9205310958000CC has been restored to the disk . . .

Restoration complete.
```

CAUTION
Lost data
If a response appears asking if the file should be overwritten, enter **NO**. Overwriting files can cause loss of data.

6 Demount DAT by following the procedure entitled *"Demount DAT"*.

Start shadowing

At the CI prompt

1 Access the PM level of the MAP terminal by typing:

> MAPCI;MTC;PM
and pressing the Enter key. ↵

Example of a MAP response:

```
CM     MS     IOD     NET     PM     CCS     TRKS     EXT     APPL

PM                            SysB   ManB    OffL   CBsy ISTb InSv
0      QUIT       PM           0      0       9      0    1    10
2      Post_
3
4
5
6
7
8
9
10
11     Disp_
12
13     Status
14     IPML
15
16
17
18



       TIME  12:00
```

2 Post the FP by typing:

> POST FP 0
and pressing the Enter key. ↵

Example of a MAP response:

```
CM     MS     IOD     NET     PM     CCS     TRKS     EXT     APPL

PM                            SysB  ManB   OffL  CBsy ISTb InSv
0      QUIT      PM            0     0       9     0    1    10
2      Post      FP            0     0       0     0    0     1
3      Plane
4      Devices  FP 0: SBS:  Plane       Devices
5               InSv
6      Tst
7      Bsy      POST:
8      RTS
9      Off
10     LoadPM
11     Disp_
12     Next
13
14     QueryPM_
15
16
17
18



       TIME  12:00
```

3 Access the DEVICES level of the MAP terminal by typing:

> DEVICES

and pressing the Enter key. ↵

Example of a MAP response:

```
CM     MS     IOD     NET     PM     CCS     TRKS     EXT     APPL

PM                          SysB  ManB   OffL  CBsy ISTb InSv
0      QUIT       PM         0     0      9     0    1    10
2      PostDEV_  FP          0     0      0     0    0     1
3      Plane
4             FP 0: SBS     Plane       Devices
5               InSv
6      Tst_
7      Bsy_            CTRL0      CTRL0          DEVICE
8      RTS_    DABM                            0 1 2 3 4 5
9      OffL_   SCSI 0          EN        DIS         – – –
10             SCSI 1          EN        DIS       – – – –
11
12            DEVICES:
13
14     QueryFP_
15     SwEn_
16
17
18



       TIME  12:00
```

4 Busy the members of the shadow set by typing:

> BSY DEV scsi_bus# scsi_dev#
for the master and pressing the Enter key. ↵

> BSY DEV scsi_bus# scsi_dev#
for the slave and pressing the Enter key. ↵

where

scsi_bus# is the SCSI bus number (0 or 1).
scsi_dev# is the SCSI device number.

Sample entry: **BSY DEV 0 0**
BSY DEV 1 0

Example of a MAP response:

```
FP 0 Busy DEV 0 0:  Command request has been submitted.
FP 0 Busy DEV 0 0:  Command passed.

FP 0 Busy DEV 1 0:  Command request has been submitted.
FP 0 Busy DEV 1 0:  Command passed.
```

5 Invoke the shadowing utility by typing:

> SHADOWUT nodetype nodenum
and pressing the Enter key. ↵

where

nodetype	is the node on which the device resides.
nodenum	is the node number.

Sample entry: **SHADOWUT fp 0**

Example of a MAP response:

```
FP0 is now node of reference
Disk shadowing utility is now active
SHADOWUT; FP0
```

6 Start shadowing on the disks by typing:

> SS setname [INSVSYNC]
and pressing the Enter key. ↵

where

setname	is the user-defined name of the shadow set.
INSVSYNC	is an optional parameter which brings devices to in-service state when SS command is executed.

Sample entry: **SS ss1 INSVSYNC**

Note: The disks will return to service (RTS) when finished.

Example of a MAP response:

```
The shadow set will be started with the following
parameter settings:

Node name        :   FP0
Shadow set name :   SS1
New Master       :
Transfer length :   Optimal
Interval         :   0
Synchronization :   InsvSync
Force            :   NO

Only members that are in a Manual Busy state can be
started.  Do you want to continue?
Please confirm (”YES” or ”NO”):
```

7 Confirm by typing:

> YES
and pressing the Enter key. ↵

Start shadowing (end)

```
Ok, Shadow Set start initiated.
1–45 minutes to complete.
```

8 Display the shadow set information by typing:

> DIS setname
and pressing the Enter key. ↵

where

setname is the user-defined name of the shadow set.

Sample entry: **DIS ss1**

Example of a MAP response:

```
Information about the shadow set #0:

Node name:               FP0
Shadow set name:         SS1
Set definition state:    RUNNING
Set operational state:   IN SERVICE
Synchronization status:  Not SYNCHRONIZED
Multi–Writes:            Serial
Capacity (blocks):       1244655
Transfer Length:         Optimal
Interval:                0
================================================
Information about member disks:
Name        State   SyncState   Reads       Writes
Perm DK00   INSV    Yes         31          0
     DK10   INSV    Fsync 0%    0           0
```

9 Exit SHADOWUT by typing:

> QUIT
and pressing the Enter key. ↵

Stop shadowing

At the CI prompt

1 Invoke the shadow utility by typing:

> SHADOWUT nodetype nodenum
and pressing the Enter key. ↵

where

nodetype is the node on which the device resides.
nodenum is the node number.

Sample entry: **SHADOWUT fp 0**

Example of a MAP response:

```
FP0 is now node of reference
Disk shadowing utility is now active
SHADOWUT; FP0
```

2 Stop shadowing on the disks by typing:

> STS setname
and pressing the Enter key. ↵

where

setname is the user-defined name of the shadow set.

Sample entry: **STS fp0ss0**

Example of a MAP response:

```
**********************************************************
***  WARNING:                                          ***
***   File Processing will no longer be available on   ***
***    the shadow set: FP0SS0   on FP0                 ***
**********************************************************
Do you wish to proceed?
Please confirm (”YES”, ”Y”, ”NO”, or ”N”):
```

3 Enter Y to stop shadowing:

Example of a MAP response:

```
Ok, Shadow Set Stop initiated.
1–10 minutes to complete.
Please wait for Stop Shadow Completion Log.
```

Verify DAT contents

At the CI prompt

1 Mount DAT by following the procedure entitled *Mount DAT*.

2 Access the DRM level of the MAP terminal by typing:

>MAPCI;MTC;APPL;OAMAP;DRM
and pressing the Enter key. ↵

Example of a MAP response:

```
CM     MS     IOD     NET     PM     CCS     TRKS     EXT     APPL

DRM                 OAMAP   EIO
0      QUIT
2
3      Info_
4      Audit_      DRM:      AFT:     MNP:       SBS:
5      Rotate_
6      Rename_
7      Mount_
8      Demount_    DRM:
9      Reset_
10     DAT_
11     View_
12     Tcopy_
13     Monitor_
14
15
16
17
18     Copy_



       TIME  12:00
```

3 Verify the contents of DAT by typing:

>DAT VERIFY
and pressing the Enter key. ↵

Verify DAT contents (end)

```
Sending request . . .
> == Request sent ==  (Please wait...)

RESPONSE READY.

   Press <CR> to continue, or enter QUIT:
```

Sample entry: ↵

Example of a MAP response:

```
Tape contains:
  UB920531095800OCC     92/06/10    12416
  PB920812082600OCC     92/08/12    48
  PB920812084500OCC     92/08/12    8
```

4 Demount DAT by following the procedure entitled *Demount DAT*.

View contents of a file

At the MAP terminal

1 Access the DRM level of the MAP terminal by typing:

>MAPCI;MTC;APPL;OAMAP;DRM
and pressing the Enter key. ↵

Example of a MAP response:

```
CM     MS     IOD     NET     PM     CCS     TRKS     EXT     APPL

DRM                 OAMAP   EIO
0      QUIT
2
3      Info_
4      Audit_      DRM:      AFT:     MNP:       SBS:
5      Rotate_
6      Rename_
7      Mount_
8      Demount_    DRM:
9      Reset_
10     DAT_
11     View_
12     Tcopy_
13     Monitor_
14
15
16
17
18     Copy_



       TIME  12:00
```

2 View the contents of a file by typing:

>VIEW stream filetmpl charcode blocknum
and pressing the Enter key. ↵

where

stream is the name of the stream datafilled in table CRSMAP.

filetmpl is the first character of a file name followed by any number of characters that sufficiently identify one or more files. It must begin with U, P, R, or $. A $ may be used to wildcard any number of characters within the file template.

View contents of a file (end)

(ASCII) or EB (EBCDIC).

blocknum is the location within the file to start viewing.

Sample entry: **VIEW occ u as 1**

Example of a MAP response:

```
Sending request to FP0...
Request sent.  (Please wait...)

RESPONSE READY.

Press <CR> to continue, or enter QUIT:

These files match the supplied filename template:
1. UN951007154100OCC       2. UN951007172000OCC
3. UN951007215600OCC       4. UN951007220001OCC

Select a :
```

Sample entry: **> 1**

Example of a MAP response:

```
Sending request to FP0...
Request sent.  (Please wait...)

RESPONSE READY.

Press <CR> to continue, or enter QUIT:
```

Sample entry: **>** ↵

Example of a MAP response:

```
0000 AAAA AAAA AAAA AAAA AAAA AAAA AAAA AAAA ,,,,,,,,,,,,,,,,
0010 AAAA AAAA AAAA AAAA AAAA AAAA AAAA AAAA ,,,,,,,,,,,,,,,,
0020 AAAA AAAA AAAA AAAA AAAA AAAA AAAA AAAA ,,,,,,,,,,,,,,,,
0030 AAAA AAAA AAAA AAAA AAAA AAAA AAAA AAAA ,,,,,,,,,,,,,,,,
0040 AAAA AAAA AAAA AAAA AAAA AAAA AAAA AAAA ,,,,,,,,,,,,,,,,
0050 AAAA AAAA AAAA AAAA AAAA AAAA AAAA AAAA ,,,,,,,,,,,,,,,,
0060 AAAA AAAA AAAA AAAA AAAA AAAA AAAA AAAA ,,,,,,,,,,,,,,,,
0070 AAAA AAAA AAAA AAAA AAAA AAAA AAAA AAAA ,,,,,,,,,,,,,,,,

Press <CR> to continue, or enter QUIT:
```

3 Follow the instructions on the screen to see more data or to quit the process.

View mounted volumes

At the MAP terminal

1 From the MAPCI level, enter table DRMAPPL by typing:

>TABLE DRMPOOL
and pressing the Enter key. ↵

Note: Table DRMPOOL does not allow any modifications.

2 View the mounted volumes by typing:

>LIST ALL
and pressing the Enter key. ↵

Example of a MAP response:

```
TOP
POOLNAME        VOLUME0         VOLUME1          VOLUME2
        VOLUME3         VOLUME4          VOLUME5          VOLUME6
        VOLUME7         VOLUME8          VOLUME9          VOLUME10
        VOLUME11        VOLUME12         VOLUME13         VOLUME14
        VOLUME15        VOLUME16         VOLUME17
---------------------------------------------------------
OMPOOL:   FP0/FP0SS0OM,1        $                $
        $               $                $                $
        $               $                $                $
        $               $                $                $
        $               $                $

 OCCPOOL :FP0/FP0SS0OCC,1       $                $
        $               $                $                $
        $               $                $                $
        $               $                $                $
        $               $                $

BOTTOM
```

Teleprocessing

Billing Server uses the Open Systems Interconnection (OSI) protocol to transfer billing data (call detail records [CDR], operator services records [OSR], and event records) and operational measurements (OM) files from a UCS DMS-250 switch to a downstream processor. The OSI stack includes seven layers of protocols, as shown in Figure 6-1.

Figure 6-1
OSI stack

Layer 7	Application
Layer 6	Presentation
Layer 5	Session
Layer 4	Transport
Layer 3	Network
Layer 2	Data
Layer 1	Physical

The three lower layers (physical, data, and network) represent the hardware and firmware commonly referred to as the lower layer protocols. The three upper layers (session, presentation, and application) represent the software commonly referred to as the upper layer protocols. The transport layer is the interface between the upper layer protocols and the lower layer protocols.

- The application layer contains service elements to support application processes.
- The presentation layer provides data translation services.
- The session layer is responsible for managing data exchange between application programs.

- The transport layer is primarily responsible for the end-to-end data integrity and shielding user applications from network concept differences.
- The network layer specifies the network interface and handles the switching and routing functions within a network.
- The data layer is responsible for the transfer of data across a communication link.
- The physical layer is responsible for activating, maintaining, and deactivating physical circuits.

Ethernet Interface Unit

The Ethernet Interface Units (EIU) contain all seven layers of the OSI application transport service. The EIU provides an Ethernet interface based on IEEE 802.3 standards and supports a 10 Mbit/s baseband bus type of LAN. The LAN uses a Carrier Sense Multiple Access with Collision Detection (CSMA/CD) method for allowing access to the communication channel. Figure 6-2 shows the LAN OSI stack used by Billing Server and the ISO protocol requirements.

Figure 6-2
LAN OSI stack

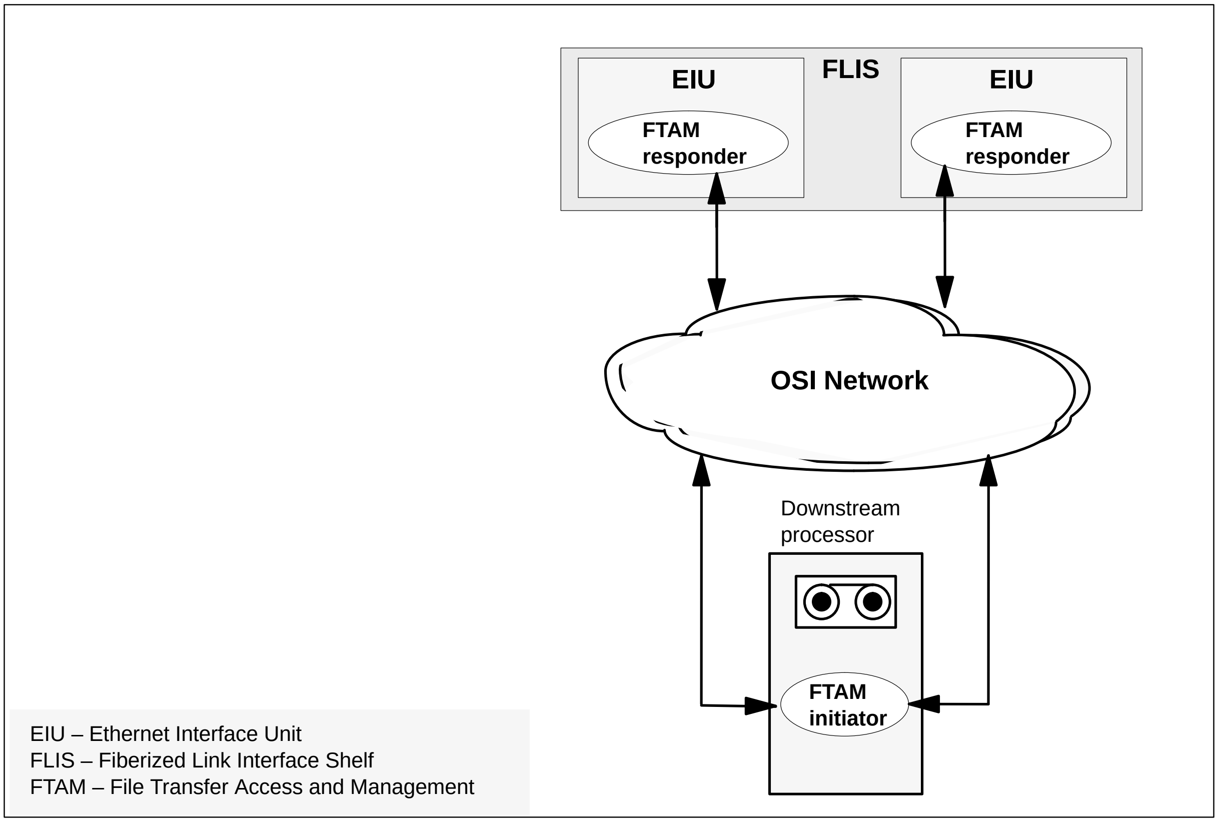
Layer 7	ISO 8571 FTAM	ISO 8649 ACSE
Layer 6	ISO 8823 Connection Oriented Presentation Protocol	ISO 8824 ASN.1
Layer 5	ISO 8327 Connection Oriented Session Protocol	
Layer 4	ISO 8073 Connection Oriented Transport Protocol Class 4	ISO 8073 AD2 Class 4 Service Over Connectionless Network Service
Layer 3	ISO 9542 ES-IS Routing Protocol	ISO 8473 CLNP
Layer 2	ISO 802.2 LLC/MAC	ISO 802.3 Ethernet Protocol
Layer 1	ISO 802.3 Ethernet Protocol	

ACSE – Association Control Service Element
ASN – Abstract Syntax Notation
CLNP – Connectionless Network Service
ES-IS – End System-Intermediate System
FTAM – File Transfer, Access, and Management
IEEE – Institute of Electrical and Electronic Engineers
ISO – International Standards Organization
LLC – Logical Link Control
MAC – Media Access Control

This LAN OSI stack also resides at the downstream processor with the same protocols.

Billing Server has two EIUs. If either link fails, the downstream computer re-establishes the connection. If the downstream computer cannot re-establish the link, it initiates a session with the other EIU. Each link can transfer CDRs and OSRs, OMs, or a combination of both. Figure 6-3 shows the connections between the EIUs and the downstream processor.

Figure 6-3
EIU and downstream processor connectivity



File Transfer, Access, and Management

The ISO 8571 File Transfer, Access, and Management (FTAM) protocol supports data transfer between Billing Server and the downstream processor. FTAM enables the transfer of CDRs, OSRs, and OMs from the Billing Server through the EIU to the downstream computer. The downstream processor requests all data transfers.

FTAM requires that one disk volume be dedicated to FTAM on the file processor (FP) to hold association and error recovery information. See the “Create FTAM volumes” procedure in this chapter.

FTAM initiator

The FTAM initiator provides a command line interface for the customer to transfer billing records to a billing site. The FTAM initiator on the downstream processor makes all requests to transfer data residing on the Billing Server.

FTAM responder

The FTAM responder on the EIU receives and implements data requests from the downstream processor.

After the downstream processor initiates the data transfer, only the FTAM responder can gracefully end the transfer. DRM files can be read and transferred several times to several destinations without using multiple copies of the file. They can also be transferred simultaneously to multiple applications. These files are available for transfer until they are erased.

The active DRM file can also be transferred to the downstream processor in real-time. The active file transfer will continue until the file is rotated.

When a file is created by DRM, it is marked as an active file. When the file is rotated (manual or automatic), it becomes an unprocessed file. A file becomes a processed file on DRM when a request to delete the file is received from the FTAM initiator.

Table control

Tables OSIPARMS and OSIROUTE define the routing requirements between Billing Server and the downstream processor.

Table OSIPARMS

Table OSIPARMS identifies the OSI layer parameters. Because the default parameters are hard-coded into the Billing Server software, datafill for table OSIPARMS is required only to change the default values. However, even though this table is available, input to this table will *not* change any of the default values associated with the OSI protocol at this time.

Table OSIROUTE

Table OSIROUTE describes the OSI static routing information and is required for a network environment. Table OSIROUTE automatically fills during installation; changes to this table are required only when the network changes.

Operational measurements

Billing Server uses the UCS DMS operational measurement (OM) collection system to collect and process operational measurements.

The OM system collects and reports operational measurements that are integer quantities indicating the quality of a particular aspect of switch performance.

Table 6-1 provides a description, history, associated registers, and validation information for the OMs that are relevant to teleprocessing.

Table 6-1
Operational measurements

Group	Register	Information
EIUETHER		**Description:** The EIUETHER OM allows access to traffic information at the Media Access Control (MAC) protocol level. The information given by this OM group includes total counts of incoming and outgoing packets and bytes at the MAC interface in a long-word format. The counts are accumulated over the OM transfer period. This OM group can be used for traffic measurements of EIU.
	EIURXBYT	**Description:** EIURXBYT is the number of bytes received on the MAC interface, including frame check sequence bytes. This register is used for EIU traffic and performance measurements. **Associated registers:** EIURXBY2 **Register validation:** None.
	EIURXPKT	**Description:** EIURXPKT is the number of unicast packets delivered to higher layer protocols on an EIU. **Associated registers:** EIURXPK2 **Register validation:** None.
	EIURXBCA	**Description:** EIURXBCA is the number of broadcast packets delivered to higher layer protocols. **Associated registers:** EIURXBC2 **Register validation:** None.
	EIURXDIS	**Description:** EIURXDIS is the minimum number of received packets to be discarded, although no errors are detected in the packet. Possible reason: resource limitations in the EIU (no free buffers to hold a received packet). **Associated registers:** EIURXDI2 **Register validation:** None.
	EIURXERR	**Description:** EIURXERR is the minimum number of received packets containing errors. **Associated registers:** EIURXER2 **Register validation:** None.

Note: For additional information on these OMs, refer to the *UCS DMS-250 Operational Measurements Reference Manual*.

—continued—

Table 6-1
Operational measurements (continued)

Group	Register	Information
	EIURXUPP	**Description:** EIURXUPP is the number of packets discarded because of an unknown or unsupported protocol. **Associated registers:** EIURXUP2 **Register validation:** None.
	EIUTXBYT	**Description:** EIUTXBYT is the number of bytes transmitted out of the MAC interface, including frame check sequence bytes. **Associated registers:** EIUTXBY2 **Register validation:** None.
	EIUTXPKT	**Description:** EIUTXPKT is the number of packets that higher-layer protocols requested to be transmitted to a unicast MAC address. **Associated registers:** EIUTXPK2 **Register validation:** None.
	EIUTXBCA	**Description:** EIUTXBCA is the number of packets that higher-layer protocols requested to be transmitted to a broadcast MAC address. **Associated registers:** EIUTXBC2 **Register validation:** None.
	EIUTXDIS	**Description:** EIUTXDIS is the number of outbound packets chosen to be discarded even though no errors are detected to prevent their being transmitted. **Associated registers:** EIUTXDI2 **Register validation:** None.
	EIUTXERR	**Description:** EIUTXERR is the number of outbound packets that could not be transmitted because of errors. **Associated registers:** EIUTXER2 **Register validation:** None.
FTAM		**Description:** FTAM OMs allow access to FTAM application activities. The information given by this group of OMs includes the total number of files transferred and the number of bytes transferred successfully.

Note: For additional information on these OMs, refer to the *UCS DMS-250 Operational Measurements Reference Manual*.

—continued—

Table 6-1
Operational measurements (continued)

Group	Register	Information
	FASSOK	**Description:** FASSOK is the number of successful attempted FTAM associations. **Associated registers:** None. **Register validation:** None.
	FASSFL	**Description:** FASSFL is the number of failed attempted FTAM associations. **Associated registers:** None. **EXT registers:** FASSFL2 **Register validation:** None.
	FDIRINF	**Description:** FDIRINF is the number of requests for directory information. **Associated registers:** None. **EXT registers:** FDIRINF2 **Register validation:** None.
	FFILOK	**Description:** FFILOK is the number of successful file transfers. **Associated registers:** None. **EXT registers:** FFILOK2 **Register validation:** None.
	FKBOK	**Description:** FKBOK is the number of Kbytes transferred successfully. **Associated registers:** None. **EXT registers:** FKBOK2 **Register validation:** None.
	FRQDEL	**Description:** FRQDEL is the number of files requested for deletion. **Associated registers:** None. **EXT registers:** FRQDEL2 **Register validation:** None.

Note: For additional information on these OMs, refer to the *UCS DMS-250 Operational Measurements Reference Manual*.

—continued—

Table 6-1
Operational measurements (continued)

Group	Register	Information
	FRESTS	**Description:** FRESTS is the number of successful restart attempts. **Associated registers:** None. **EXT registers:** FRESTS2 **Register validation:** None.
	FRESTF	**Description:** FRESTF is the number of failed restart attempts. **Associated registers:** None. **EXT registers:** FRESTF2 **Register validation:** None.
	FRECAS (Note 2)	**Description:** FRECAS is the number of successful recovery attempts. **Associated registers:** None. **EXT registers:** FRECAS2 **Register validation:** None.
	FRECAF (Note 2)	**Description:** FRECAF is the number of failed recovery attempts. **Associated registers:** None. **EXT registers:** FRECAF2 **Register validation:** None.
Note: For additional information on these OMs, refer to the *UCS DMS-250 Operational Measurements Reference Manual*.		
—end—		

Logs

FTAM generates logs that fall into one of the following categories:

- functionality loss due to hardware faults
- maintenance state changes
- data communication failures
- disk utilization exceeding thresholds
- operations affecting switch configuration (datafill)

Table 6-2 shows the logs specific to Billing Server teleprocessing.

Table 6-2
Logs

Log	Explanation
FTAM301	An error has occurred while trying to access the application processor. The actual error description and application processor location are included in the log.
FTAM600	An FTAM general diagnostic has been sent or received.
FTAM601	An FTAM protocol diagnostic has been sent or received.
FTAM605	An FTAM access diagnostic has been sent or received.
FTAM700	An FTAM event has occurred. This is an informational log.
FTAM701	Reports normal or abnormal completion of a file transfer, as well as the file size transferred. It is generated at the beginning and end of a file transfer.
Note: For additional information on these logs, refer to the *AD3500* design document.	

User interface

In several of the procedures, the MAP terminal prompts the user for one of the actions in Table 6-3.

Table 6-3
User actions

Terminal prompt	User action
`Press <CR> to continue, or enter QUIT`	Press the enter key to continue the process or type QUIT to discontinue the process.
`Select a file number`	Enter the number of the file used in the process preceded by a space.

Table 6-4 shows the teleprocessing procedures available to the user.

Table 6-4
Teleprocessing procedures

Procedure	Utility – Command	page
Create FTAM volumes	DISKADM – CV	6–12
Delete FTAM volume	DISKADM – DDV	6–18
Determine FTAM billing transfer	QUERYXFER	6–23

Create File Transfer, Access, and Management volumes

At the CI prompt

1 Access the PM level of the MAP terminal by typing:

> MAPCI;MTC;PM
and pressing the Enter key. ↵

Example of a MAP response:

```
CM     MS     IOD     NET     PM     CCS     TRKS     EXT     APPL

PM                            SysB   ManB    OffL   CBsy ISTb InSv
0      QUIT      PM            0      0       9      0     1    10
2      Post
3
4
5
6
7
8
9
10
11     Disp_
12
13     Status
14     IPML
15
16
17
18


       TIME  12:00
```

2 Post the FP by typing:

> POST FP 0
and pressing the Enter key. ↵

Example of a MAP response:

```
CM    MS    IOD    NET    PM    CCS    TRKS    EXT    APPL

PM                        SysB  ManB   OffL  CBsy ISTb InSv
0     QUIT      PM         0     0      9     0    1    10
2     Post      FP         0     0      0     0    0     1
3     Plane
4     Devices  FP 0: SBS:  Plane       Devices
5              InSv
6     Tst
7     Bsy      POST:
8     RTS
9     Off
10    LoadPM
11    Disp_
12    Next
13
14    QueryPM_
15
16
17
18



      TIME  12:00
```

3 Access the DEVICES level of the MAP terminal by typing:

> DEVICES
and pressing the Enter key. ↵

Example of a MAP response:

```
CM     MS     IOD     NET     PM     CCS     TRKS     EXT     APPL

PM                             SysB  ManB   OffL  CBsy ISTb InSv
0      QUIT      PM             0     0      9     0    1    10
2      PostDEV_  FP             0     0      0     0    0     1
3      Plane
4             FP 0: SBS     Plane      Devices
5               InSv
6      Tst_
7      Bsy_            CTRL0      CTRL0          DEVICE
8      RTS_    DABM                            0 1 2 3 4 5
9      OffL_   SCSI 0          EN        DIS          - - -
10             SCSI 1          EN        DIS        - - - -
11
12            DEVICES:
13
14     QueryFP_
15     SwEn_
16
17
18



       TIME  12:00
```

4 Busy the disk on which volumes are to be formatted/created by typing:

> BSY DEV scsi_bus# scsi_dev#
and pressing the Enter key. ↵

where

scsi_bus# is the SCSI bus number.
scsi_dev# is the SCSI device number.

Sample entry: **BSY DEV 0 0**

Example of a MAP response:

```
FP 0 Busy DEV 0 0:  Command request has been submitted.
FP 0 Busy DEV 0 0:  Command passed.
```

5 Perform disk administration by typing:

> DISKADM devname nodetype nodenum
and pressing the Enter key. ↵

where

devname is the name of the device needing administration.
nodetype is the node on which the device resides.
nodenum is the node number.

Sample entry: **DISKADM dk00 fp 0**

Example of a MAP response:

```
Start up command sequence is in progress.
This may take a few minutes.
Administration of device DK00 on FP0 is now active.
DISKADM; FP0
```

6

If	Do
disk needs formatting	step 7
disk does not need formatting	step 9

7

CAUTION
Lost data
The following command causes all data on the disk to be lost.

Erase all volumes and files on the disk by typing:

> FD QUICK FORCE
and pressing the Enter key. ↵

Example of a MAP response:

```
           ***** WARNING *****
Formatting of DK00
Will destroy the contents of the disk.

The formatting will:
  allocate 3 spare or alternate sectors per track,
  allocate 16 spare or alternate tracks per disk,
  use the G defect list,
  assign QUICK as the name for the disk,
  perform quick format,
  include force option.

Do you want to continue?
Please confirm (”YES” or ”NO”):
```

8 Confirm by typing:

> YES
and pressing the Enter key. ↵

Example of a MAP response:

```
Initializing the system data structures on the disk.
Formatting and initialization of the disk is completed.
```

9 Create an FTAM volume for use with Billing Server by typing:

> CV volname volsize voltype [LBLOCK blksize]
and pressing the Enter key. ↵

where

volname	is the logical name of the volume (any 8 characters).
volsize	is the size of the volume (1–32767).
voltype	is the type of volume (FTFS)
LBLOCK	precedes the logical block size
blksize	is the size of the volume in Kbytes (1, 2, 4, 8, 16, 32, or 64)

Note: If LBLOCK is not defined, the default logical block size is 1.

Sample entry: **CV ftam 6 ftfs LBLOCK 16**

Example of a MAP response:

```
FTFS volume FTAM will be created on DK00.
Volume size:                6 megabytes
first FID table extent size:        32754 entries
Volume Free Space Map size:     7936 segments

Do you want to continue?
Please confirm (”YES” or ”NO”):
```

10 Confirm by typing:

> YES
and pressing the Enter key. ↵

Example of a MAP response:

```
Creation of the volume is completed.
```

11 Quit DISKADM by typing:

> QUIT
and pressing the Enter key. ↵

12 Bring the disk back into service by typing:

> RTS DEV scsi_bus# scsi_dev#
and pressing the Enter key. ↵

where

scsi_bus#	is the SCSI bus number.
scsi_dev#	is the SCSI device number.

Create File Transfer, Access, and Management volumes (end)

```
FP 0 RTS DEV 0 0:  Command request has been submitted
FP 0 RTS DEV o o:  Command passed
```

Delete FTAM volume

CAUTION
FTAM volume is required for normal file transfer
The FTAM volume is required by Billing Server for normal downstream file transfer.

At the CI prompt

1 Access the PM level of the MAP terminal by typing:

> MAPCI;MTC;PM
and pressing the Enter key. ↵

Example of a MAP response:

```
CM     MS     IOD     NET     PM     CCS     TRKS     EXT     APPL

PM                            SysB   ManB    OffL   CBsy ISTb InSv
0      QUIT       PM           0      0       9      0     1    10
2      Post
3
4
5
6
7
8
9
10
11     Disp_
12
13     Status
14     IPML
15
16
17
18



       TIME  12:00
```

2 Post the FP by typing:

> POST FP 0
and pressing the Enter key. ↵

Example of a MAP response:

```
CM     MS     IOD     NET     PM     CCS     TRKS     EXT     APPL

PM                            SysB  ManB   OffL  CBsy ISTb InSv
0      QUIT      PM            0     0      9     0    1    10
2      Post      FP            0     0      0     0    0     1
3      Plane
4      Devices  FP 0: SBS:   Plane       Devices
5               InSv
6      Tst
7      Bsy      POST:
8      RTS
9      Off
10     LoadPM
11     Disp_
12     Next
13
14     QueryPM_
15
16
17
18



       TIME  12:00
```

3 Access the DEVICES level of the MAP terminal by typing:

> DEVICES
and pressing the Enter key. ↵

Example of a MAP response:

```
CM     MS     IOD     NET     PM     CCS     TRKS     EXT     APPL

PM                            SysB  ManB   OffL  CBsy ISTb InSv
0      QUIT      PM            0     0      9     0    1    10
2      PostDEV_  FP            0     0      0     0    0     1
3      Plane
4             FP 0: SBS     Plane      Devices
5               InSv
6      Tst_
7      Bsy_            CTRL0      CTRL0          DEVICE
8      RTS_    DABM                            0 1 2 3 4 5
9      OffL_   SCSI 0          EN        DIS         – – –
10             SCSI 1          EN        DIS       – – – –
11
12            DEVICES:
13
14     QueryFP_
15     SwEn_
16
17
18



       TIME  12:00
```

4 Busy the disk on which volumes are to be formatted/created by typing:

> BSY DEV scsi_bus# scsi_dev#
and pressing the Enter key. ↵

where

scsi_bus# is the SCSI bus number.
scsi_dev# is the SCSI device number.

Sample entry: **BSY DEV 0 0**

Example of a MAP response:

```
FP 0 Busy DEV 0 0:  Command request has been submitted.
FP 0 Busy DEV 0 0:  Command passed.
```

5 Perform disk administration by typing:

> DISKADM devname nodetype nodenum
and pressing the Enter key. ↵

where

devname	is the name of the device needing administration.
nodetype	is the node on which the device resides.
nodenum	is the node number.

Sample entry: **DISKADM dk00 fp 0**

Example of a MAP response:

```
Start up command sequence is in progress.
This may take a few minutes.
Administration of device DK00 on FP0 is now active.
DISKADM; FP0
```

6

CAUTION
Lost data
Verify that no data exists on the volume to be deleted. Confirming deletion causes all data on the disk to be lost.

Delete the FTAM volume by typing:

> DDV volname
and pressing the Enter key. ↵

where

volname	is the name of the volume to be deleted.

Sample entry: **DDV ftam**

Example of a MAP response:

```
          ***** WARNING *****
Deleting volume FTAM on DK00 will DESTROY the contents of
the volume.

It may CORRUPT TRMS databases on OTHER volumes.

Do you want to continue?
Please confirm (”YES” or ”NO”):
```

If	Do
Yes	step 7
No	step 8

7 Confirm by typing:

> YES
and pressing the Enter key. ↵

Delete FTAM volume (end)

```
Deletion of the volume is completed.
```

8 Abort the delete command by typing:

> No
and pressing the Enter key. ↵

Response:

```
DELETEVOL command is aborted
```

9 Quit DISKADM by typing:

> QUIT
and pressing the Enter key. ↵

10 Bring the disk back into service by typing:

> RTS DEV scsi_bus# scsi_dev#
where

scsi_bus# is the SCSI bus number.
scsi_dev# is the SCSI device number.

Example of a MAP response:

```
FP 0 RTS DEV 0 0:  Command request has been submitted
FP 0 RTS DEV o o:  Command passed
```

Determine FTAM billing transfer

At the CI prompt

1 Query the billing transfer by typing:

>QUERYXFER

and pressing the Enter key. ↵

Example of a MAP response:

```
Querying remote nodes, Please wait.
Query Complete.  Processing Transfer Information.
There are no active FTAM sessions for EIU102
Node id: EIU208

                                    File       Xfer
File Name                           Xfer KB    State  Start Time
--------------------------------     -------   -----  --------------
:FP0/FP0SS1OCC/AN951013030000CC       132          read   10/13 09:47:21
Done.
```

Note: To verify a transfer is occurring, perform the QUERYXFER command three to four times. If a transfer is occurring, the number under the field File Xfer KB will increment. For example,

```
Querying remote nodes, Please wait.
Query Complete.  Processing Transfer Information.
There are no active FTAM sessions for EIU102
Node id: EIU208

                                    File       Xfer
File Name                           Xfer KB    State  Start Time
--------------------------------     -------   -----  --------------
:FP0/FP0SS1OCC/AN951013030000CC       126          read   10/13 09:47:21
Done.

Querying remote nodes, Please wait.
Query Complete.  Processing Transfer Information.
There are no active FTAM sessions for EIU102
Node id: EIU208

                                    File       Xfer
File Name                           Xfer KB    State  Start Time
--------------------------------     -------   -----  --------------
:FP0/FP0SS1OCC/AN951013030000CC       474          read   10/13 09:47:21
Done.
```

SBS menus

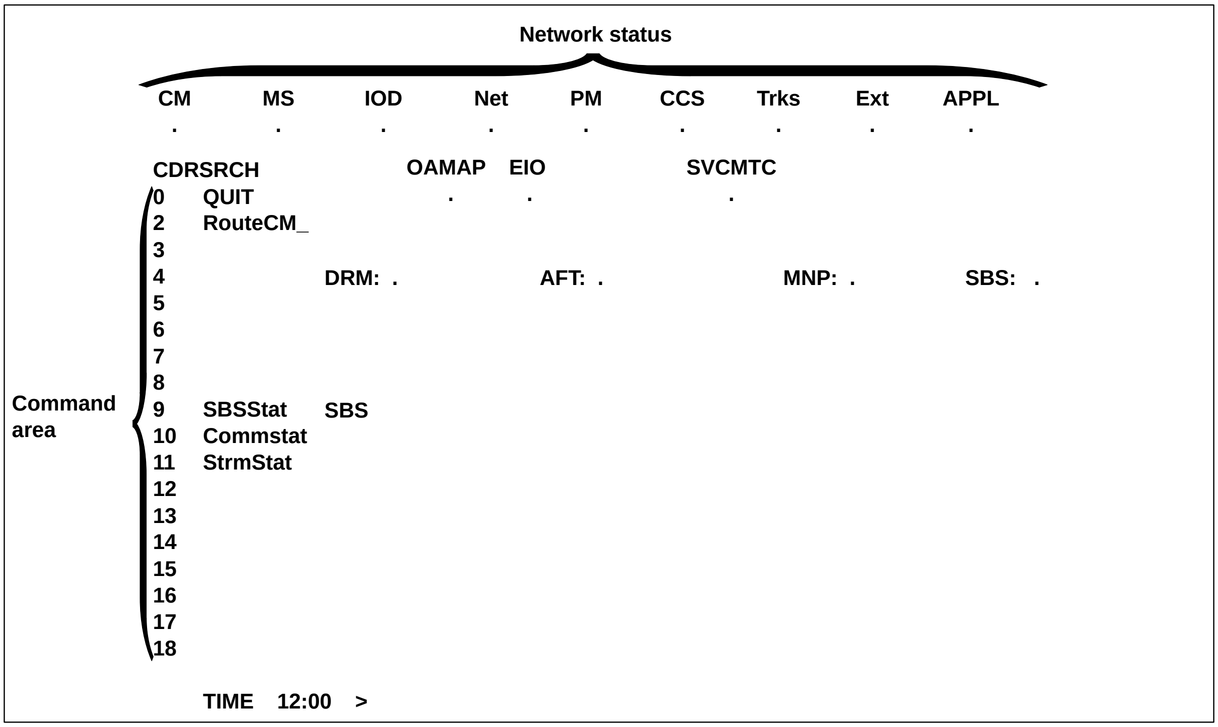
Billing Server uses the SuperNode Billing System (SBS) menus to activate, deactivate, and examine the current status of the billing system. There are four SBS menus:

- SBS
- SBS Status (SBSSTAT)
- Stream Status (STRMSTAT)
- Communication Status (COMMSTAT)

How to read a menu

UCS DMS-250 menus are divided into two areas: network status and command (Figure 7-1).

Figure 7-1
Menu areas



Network status

The network status area indicates the most recent and most severe alarm status of the UCS DMS-250 subsystems.

- CM – computing module
- MS – message switch
- IOD – input/output devices
- Net – network
- PM – peripheral module
- CCS – Common Channel Signaling
- Trks – trunks
- Ext – external alarms
- APPL – applications

The Billing Server's current status is located under the applications (APPL) subsystem. The following syntax is used to identify the subsystem status (located under the subsystem name):

- dot (.) — the subsystem is in service and has no alarms
- M — the subsystem is experiencing a minor alarm
- *M* — the subsystem is experiencing a major alarm
- *C* — the subsystem is experiencing a critical alarm

Command

The command area of the display lists commands available from this MAP level. The full command, abbreviated command (capitalized letters), or the associated number is used to execute the command. When an underscore precedes or follows a command, another command or parameter may be required.

SBS menu

To enter the SBS menu (Figure 7-2) from a MAP terminal, type the following:

MAPCI;MTC;APPL;OAMAP;SBS

Figure 7-2
SBS menu

```
      CM     MS      IOD      Net      PM      CCS      Trk      Ex      APPL
       .      .       .        .        .       .        s        t        .
  SBS                                                    .        .
  0   QUIT              OAMAP   EIO              SVMTC    SDMBIL
  2   RouteCM_            .      .                 .        .
  3
  4             DRM:        AFT:          MNP:   .        SBS:   SBS
  5             .           .                             OK
  6
  7
  8
  9   SBSStat      SBS
  10  Commstat
  11  StrmStat
  12
  13
  14
  15
  16
  17
  18

      TIME     12:00
        >
```

SBS commands

The SBS menu level provides the following five commands:

- QUIT — exits one or more MAP levels.
- RouteCM — designates the billing system (Billing Server or DIRP).
- SBSStat — accesses the SBS status area.
- Commstat — accesses the communications status area.
- StrmStat — accesses the stream status area.

SBSStat menu

The SBS Status (SBSStat) menu shows the current status and alarms for the SuperNode billing system. To enter the SBSStat menu (Figure 7-3) from a MAP terminal, type the following:

MAPCI;MTC;APPL;OAMAP;SBS;SBSStat

Figure 7-3
SBSStat menu

```
CM    MS    IOD    NET    PM    CCS    TRKS    EXT    APPL

SBSStat                OAMAP   EIO     SVMTC
0 QUIT
2 RouteCM_
3
4          DRM:      AFT:     MNP:       SBS: SBS OF
5
6
7 Prevpage
8 Nextpage
9       SBS Status:  In Service
10 CommStat ---Collector------Stream Group------FSA------
11StrmStat NodeStatAlarmComm NameStatAlarmNode StatAlarmComm
12 SortColl  CM                OCC               FP0
13 SortStrm
14 SortFSA
15
16
17
18    SBSSTAT:



      TIME  12:00
```

The SBSStat menu is divided into the following areas:

- network status
- command
- SBS status area
- collector status area
- stream group status area
- formatter/storage agent (FSA) status area

SBSStat commands

Nine commands are available at the SBSStat menu level:

- QUIT — exit one or more MAP levels.
- RouteCM — designates the billing system (Billing Server or DIRP).
- Prevpage — displays the previous page of streams when required.

- NextPage — displays the next page of streams when required.
- Commstat — accesses the communications status area.
- StrmStat — accesses the stream status area.
- SortColl — displays the data sorted by collector (CM), stream, and FSA.
- SortStrm — displays the data sorted by stream, collector, and FSA.
- SortFSA — displays the data sorted by FSA, stream, and collector.

SBS status area

The SBS status area indicates the most recent, most severe status of the SuperNode Billing Server. The status is derived from the Stream Group states. Possible SBS states and their associated alarms are listed below.

- SBS OK – All streams are in service.
- SBS IS – a stream (one or more) is in an in-service trouble (IStb) state
- SBS RT – a stream (one or more) is in a “node cannot rotate” (NoRot) state (minor alarm)
- SBS JN – a stream (one or more) is in a Join state (minor alarm)
- SBS SP – a stream (one or more) is in a Split state (major alarm)
- SBS DK – a stream (one or more) is in a NoDisk state (critical alarm)
- SBS CM – a stream (one or more) is in an Unknown state (critical alarm). A node communication problem exists.
- SBS CO – a node (one or more) is not communicating with the SCM and is in an Unknown Stream state (critical alarm)
- SBS AF – a stream (one or more) is in an “all full” state (critical alarm)
- SBS RB – the remote collector node has run out of billing blocks (critical alarm)
- SBS EX – the stream has run out of extension blocks and has switched back to the AMA DIRP billing system (critical alarm)
- SBS NF – the billing has switched to the core because there is no FSA available to send the billing. This means that the FP disks have become unavailable to write billing, or that the DRM is unable to do an active file rotate and there is no available FSA on any node for rotation (critical alarm).
- SBS NS – indicates that billing has switched to the core due to SIPC communication problems and is unable to communicate with the FP (critical alarm).

Collector status area

The collector status area shows the node name, status, alarms, and communications for the collector.

Node name

The node name shows the name of the originating billing data stream node (CM).

Status

The collector status area shows the state of a stream in the stream group. This data is reported by the collector node. If communication is disrupted between the SCM and the collector, the last known status remains until communication is re-established. Possible states and their associated alarms are listed below.

- Estab – collector is attempting to connect to the FSA for the first time (no alarm)
- EstabF – collector is attempting to connect to another FSA since it failed to connect to the first FSA (no alarm)
- EstabP – collector is pausing before attempting to connect to the FSA (no alarm). EstabP occurs after attempts to connect to all FSAs have been made.
- dot (.) – collector is inservice (no alarm). The FSA is handling the stream without any problems.
- Move – stream is changing to a different FSA (no alarm)
- MoveF – stream is changing to another FSA after failing to connect with the previous FSA
- MoveP – stream is changing to a different FSA after failing to connect to any FSA (no alarm)
- Failed – collector cannot establish a connection with the FSA (critical alarm)
- Unknow – no collector process status report has been received (critical alarm)
- Dead – collector process handling the stream died (critical alarm). The collector will attempt to recover itself.

Alarms

Alarms are represented by dot (.), M (minor), *M* (major), or *C* (critical).

Collector communications

The collector communications area indicates whether the communication lines between the SCM and the collector are working. Possible states are listed below.

- dot (.) – communication lines are open and working
- UNA – communication is unavailable. The collector status and alarm fields may not be current.
- EST – connection is being established
- RBB – connection is good, but the collector has run out of billing blocks

Stream group status area

The stream group status area shows the name, status, and alarms for the stream group.

Name

The name column holds the stream name. Each datafilled stream is listed. A stream name can be a maximum of four characters; one stream name appears on each line. Each stream has a stream group status and alarm status.

Stream group status

The stream group status area shows the most recent and most severe state of a stream. Possible states and their associated alarms are listed below.

- "." – stream is inservice (no alarm). All billing streams are being processed for this stream group.
- IStb – the collector for this stream can write billing data an FSA (no alarm). An FP may need volumes assigned to this stream. All billing streams are being processed for this stream group.
- Moving – a stream is being moved to a different FSA (no alarm).
- NoRot – the FSAs cannot perform a rotate (minor alarm). The stream group may run out of disk space soon. All billing streams are being processed for this stream group.
- Join – a stream name used by more than one collector has a stream (one or more) that is unable to talk to the FSA (minor alarm). The other streams of the group are still able to talk to the FSA.

The SCM has identified an FSA that can handle all of the single streams in the stream group and is in the process of connecting them. All billing streams are being processed somewhere. This is a minor alarm.

- Split – a stream name used by more than one collector has a stream (one or more) that is unable to talk to the same FSA as the other members of the stream group (major alarm).

  The SCM cannot find an FSA to handle the stream group. All billing streams are being processed somewhere.

- No Disk – the FSA has run out of disk space for this stream group (critical alarm). Therefore, no billing is being processed for this stream group.
- Contra – the FSA status and the collector status are contradictory (critical alarm).
- AlFull – every disk volume for this stream is in a critical alarm (major alarm for SBS). Possibly, all disks are full.
- SBS Ext BLK – the stream has run out of extension blocks and has switched back to the AMA DIRP billing system (critical alarm)
- NoFSA – the last status for the associated stream and the reason that the switchback has occurred. There is no FSA to write billing to for this stream. SBS is unable to write billing records to any active file for this stream, or on any FSA (critical alarm).
- NoSIPC – the last status for the associated stream and the reason that the switchback has occurred. This means that there is a SIPC communication problem between the collector and the FSA. SBS is unable to send billing to the FSA for the associated stream (critical alarm).

Alarms

Alarms are represented by dot (.), M (minor), *M* (major), or *C* (critical).

FSA status area

The FSA status area shows the name, status, alarms, and communications for the FSA.

FSA node

The FSA node indicates the name of the node where the billing data streams are merged and passed to the distributed recording manager. The FSA resides on the file processor (FP0).

FSA status

The FSA status area indicates the status of billing for a particular stream group. Possible states and their associated alarms are listed below.

- "." – billing stream is processing without problems (no alarms).
- DKCrit – free disk space is below the critical alarm threshold (critical alarm). The billing stream is still being processed.
- DKMin – free disk space is below the minor alarm threshold (minor alarm). The billing stream is still being processed.
- DKMaj – free disk space is below the major alarm threshold (major alarm). The billing stream is still being processed.
- NoColl – the FSA cannot talk to the collector (critical alarm). The billing stream is not being processed.
- DKBad – the SCSI disk is not available (critical alarm). The billing stream is not being processed.
- "——" – no FSA is currently assigned to the collector. The SCM believes there is no FSA to handle the collector's data stream.

Alarms

Alarms are represented by dot (.), M (minor), *M* (major), or *C* (critical).

FSA communications

The FSA communications area indicates whether the communication lines between the SCM and the FSA are working. Possible states are listed below.

- dot (.) – communication lines are open and working.
- UNA – communication is unavailable. The FSA status and alarm fields may not be current.
- EST – a connection with the SCM is being established.

StrmStat menu

Stream status information indicates whether the volumes for a particular stream on a particular node are full.

The stream status (STRMStat) menu shows, for each stream group, the status and alarms of each FSA node and whether the volumes for the stream are full.

The StrmStat display provides the following information for all FSA nodes and all streams:

- the status of each stream group
- which FSA nodes can handle which streams groups
- why FSA nodes cannot handle certain stream groups

- which FSA nodes are processing the stream

This information helps you diagnose problems, such as why a stream has no disk.

To enter the StrmStat menu (Figure 7-4) from a MAP terminal, type the following:

MAPCI;MTC;APPL;OAMAP;SBS;StrmStat

Figure 7-4
The STRMStat MAP display level

```
CM    MS    IOD    NET    PM    CCS    TRKS    EXT    APPL

StrmStat      OAMAP  EIO          SVMTC
0  QUIT
2  Route_CM
3
4          DRM:      AFT:        MNP:          SBS: SBS OF
5
6
7  PrevPage
8  NextPage
9  SBSStat  Stream   BS   Status   Alarm   FP0
10 CommStat    OCC    D
11    Disp_
12
13    Status
14    IPML
15
16
17
18



      TIME  12:00   >
```

The StrmStat display can be divided into the following areas:

- network status
- command
- stream group name
- billing system
- stream group status

- stream group alarms
- FSA node status

StrmStat commands

Six commands are available at the StrmStat menu level.

- QUIT — exits one or more MAP levels
- RouteCM — designates the billing system (Billing Server or DIRP)
- Prevpage — displays the previous page of streams when required
- NextPage — displays the next page of streams when required
- SBSStat — accesses the SBS status area
- StrmStat — accesses the stream status area

Stream group name

The stream group name area indicates the name of the stream group. Each stream group is listed. A stream group name can have a maximum of four characters; only one stream group name can appear on a line. All the fields in a row correspond to the stream group that is listed on that line.

Billing system

The billing system area indicates which billing system is currently assigned to the stream.

- S – SBS is handling the billing
- D – DIRP is handling the billing

Stream group status

The stream group status area shows the most recent and most severe state of a stream. Possible states and their associated alarms are listed below.

- "." – stream is inservice (no alarm). All billing streams are being processed for this stream group.
- IStb – the collector for this stream can write billing data an FSA (no alarm). An FP may need volumes assigned to this stream. All billing streams are being processed for this stream group.
- Moving – a stream is being moved to a different FSA (no alarm).
- NoRot – the FSAs cannot perform a rotate (minor alarm). The stream group may run out of disk space soon. All billing streams are being processed for this stream group.
- Join – a stream name used by more than one collector has a stream (one or more) that is unable to talk to the FSA (minor alarm). The other streams of the group are still able to talk to the FSA.

The SCM has identified an FSA that can handle all of the single streams in the stream group and is in the process of connecting them. All billing streams are being processed somewhere. This is a minor alarm.

- Split – a stream name used by more than one collector has a stream (one or more) that is unable to talk to the same FSA as the other members of the stream group (major alarm).

  The SCM cannot find an FSA to handle the stream group. All billing streams are being processed somewhere.

- No Disk – the FSA has run out of disk space for this stream group (critical alarm). Therefore, no billing is being processed for this stream group.
- Contra – the FSA status and the collector status are contradictory (critical alarm).
- AlFull – every disk volume for this stream is in a critical alarm (major alarm for SBS). Possibly, all disks are full.
- SBS Ext BLK – the stream has run out of extension blocks and has switched back to the AMA DIRP billing system (critical alarm)
- NoFSA – the last status for the associated stream and the reason that the switchback has occurred. There is no FSA to write billing to for this stream. SBS is unable to write billing records to any active file for this stream, or on any FSA (critical alarm).
- NoSIPC – the last status for the associated stream and the reason that the switchback has occurred. This means that there is a SIPC communication problem between the collector and the FSA. SBS is unable to send billing to the FSA for the associated stream (critical alarm).

Alarms

Alarms are represented by dot (.), M (minor), *M* (major), or *C* (critical).

FSA node status

The FSA status area indicates the status of billing for a particular stream group. Possible states and their associated alarms follow.

- “.” – billing stream is processing without problems (no alarms).
- DK – the volumes have no empty space for this stream to record on this node.
- DKCrit – free disk space is below the critical alarm threshold (critical alarm). The billing stream is still being processed.
- DKMin – free disk space is below the minor alarm threshold (minor alarm). The billing stream is still being processed.

- DKMaj – free disk space is below the major alarm threshold (major alarm). The billing stream is still being processed.
- NoColl – the FSA cannot talk to the collector (critical alarm). The billing stream is not being processed.
- DKBad – the SCSI disk is not available (critical alarm). The billing stream is not being processed.
- CM – the SIPC communication between the DMS-Core and the FSA on this node is not available. This status can override the “*” or “.” statuses.
- Nr – an attempt to register with DRM has not yet been successful. Normally, this is displayed only briefly.
- Wa – DRM registration was successful, but recording cannot yet be started because DRM has not found enough disk space to begin recording. If this situation persists, check the DRM MAP level to see how much disk space is available.
- “——” – no FSA is currently assigned to the collector. The SCM believes there is no FSA to handle the collector’s data stream.

CommStat menu

The communication status (COMMStat) menu displays the collectors and FSAs communicating with the SCM. This information is useful for diagnosing problems. Unlike the other three levels, COMMStat displays the status of collector nodes that do *not* have a stream assigned to them and the collector nodes that have a stream assigned.

To enter the CommStat menu (Figure 7-5) from a MAP terminal, type the following:

MAPCI;MTC;APPL;OAMAP;SBS;CommStat

Figure 7-5
The COMMStat MAP display level

```
CM    MS    IOD    NET    PM    CCS    TRKS    EXT    APPL

StrmStat      OAMAP  EIO           SVMTC
0  QUIT
2
3
4          DRM:      AFT:        MNP:          SBS: SBS OF
5
6
7
8
9  SBSStat   Collector Status:
10             CM
11  StrmStat
12
13
14             FSA Status:
15              FP0
16
17
18              COMMSTAT:



      TIME  12:00   >
```

The CommStat MAP display can be divided into the following areas:

- network status
- command
- collector status
- FSA status

CommStat commands

There are three commands available at the CommStat menu level.

- QUIT — is used to exit one or more MAP levels.
- SBSStat — is used to access the SBS status area.
- StrmStat — is used to access the stream status area.

Collector status

This area shows the collector’s current status. Possible state values are listed below.

- “.” – collector is in service.
- UNA – the SCM cannot communicate with the collector.
- RB – the collector has run out of billing blocks.

FSA status

This area indicates the current status of the FSA node. Possible state values are listed below

- “.” – FSA is in service.
- UNA – the SCM cannot communicate with the FSA.

Data streams

Before Billing Server can be activated, the data streams must be defined in tables SBSMAP and SBSFMT. Defining the streams in these tables allows data to be routed to Billing Server instead of DMS-Core based billing.

Note: DMS-Core based billing uses tables CRSFMT and CRSMAP.

Logs

The billing system generates logs that fall into one of the following categories:

- functionality loss due to hardware faults
- maintenance state changes
- data communication failures
- disk utilization exceeding thresholds
- operations affecting switch configuration (datafill)

Table 7-1 shows the logs specific to the billing system.

Table 7-1
Logs

Log	Explanation
AUD582	The SBS software on the Computing Module (CM) is trapped or has died while holding one of the SBS_EXT_BLOCK extension blocks.
SBS300	A switchback to another billing system has occurred due to some trouble the system has detected. The SCM generates this log when the stream is in SBSFMT and CRSFMT tables, and enters the failed state, and has no billing blocks. This log is always accompanied by an SBS500 log.
SBS301	SCM detects a communication failure with either the collector of the FSA.
SBS302	Stream cannot be handled while the SCM link is down.
SBS303	Billing stream has failed and there is no backup billing system to handle the stream.
SBS304	A supervisory process (not associated with a particular stream) has failed twice within 30 seconds. The process waits two minutes before attempting to recover.
SBS500	A SuperNode Billing System (SBS) has been activated or deactivated. Both user and system initiated changes are logged.
Note: For additional information on these logs, refer to *UCS DMS-250 Log Reports Reference Manual*.	

Table 7-2 shows the SBS procedures available to the user.

Table 7-2
SBS procedures

Procedure	Utility – Command	page
Activate Billing Server	SBS menu – RouteCM	7–17
Deactivate Billing Server	SBS menu – RouteCM	7–19
Set billing stream	tables SBSFMT and SBSMAP	7–21

Activate Billing Server

At the CI prompt

1 Access the SBS level of the MAP terminal by typing:

>MAPCI;MTC;APPL;OAMAP;SBS
and pressing the Enter key. ↵

Example of a MAP response:

```
CM     MS     IOD     NET     PM     CCS     TRKS     EXT     APPL

SBS
0  QUIT        OAMAP  EIO           SVMTC
2  RouteCM_
3
4          DRM:      AFT:        MNP:          SBS: SBS OF
5
6
7
8
9   SBSStat   SBS
10  CommStat          CM
11  StrmStat
12
13
14
15
16
17
18



      TIME  12:00   >
```

2 Activate the Billing Server by typing:

>ROUTECM SBS stream
and pressing the Enter key. ↵

where

stream is the stream assigned to handle billing.

Sample entry: **>ROUTECM sbs occ**

Activate Billing Server (end)

```
WARNING:  Activating the S/DMS Billing Server will send
billing to the FP:
billing will no longer be sent to the EIOC or CM.
Do you really want to activate the S/DMS Billing Server?
Please confirm (”YES” or ”NO”)
```

3 Confirm activation of the Billing Server by typing:

>Yes

and pressing the Enter key. ↵

Example of a MAP response:

```
CM     MS     IOD     NET     PM     CCS     TRKS     EXT     APPL

SBS
0  QUIT        OAMAP  EIO           SVMTC
2  RouteCM_
3
4          DRM:      AFT:        MNP:          SBS: SBS OK
5
6
7
8
9   SBSStat   SBS
10  CommStat          CM
11  StrmStat
12
13
14
15
16
17
18



      TIME  12:00   >
```

Deactivate Billing Server

At the CI prompt

1 Access the SBS level of the MAP terminal by typing:

>MAPCI;MTC;APPL;OAMAP;SBS
and pressing the Enter key. ↵

Example of a MAP response:

```
CM     MS     IOD     NET     PM     CCS     TRKS     EXT     APPL

SBS
0  QUIT        OAMAP  EIO           SVMTC
2  RouteCM_
3
4           DRM:      AFT:        MNP:           SBS: SBS OK
5
6
7
8
9   SBSStat   SBS
10  CommStat          CM
11  StrmStat
12
13
14
15
16
17
18



      TIME  12:00   >
```

2 Deactivate the Billing Server by typing:

>ROUTECM DIRP stream
and pressing the Enter key. ↵

where

stream is the stream assigned to handle billing.

Sample entry: **>ROUTECM DIRP occ**

Deactivate Billing Server (end)

```
WARNING:  If request is successful, billing data will no
longer be processed by the S/DMS Billing Server.  Billing
data will go to the Backup billing server.
Do you really want to deactivate the S/DMS Billing Server?
Please confirm (”YES” or ”NO”)
```

3 Confirm deactivation of the Billing Server by typing:

>Yes

and pressing the Enter key. ↵

Example of a MAP response:

```
CM     MS     IOD     NET     PM     CCS     TRKS     EXT     APPL

SBS
0  QUIT        OAMAP  EIO           SVMTC
2  RouteCM_
3
4          DRM:      AFT:        MNP:          SBS: SBS OF
5
6
7
8
9   SBSStat   SBS
10  CommStat          CM
11  StrmStat
12
13
14
15
16
17
18



      TIME  12:00   >
```

Set billing stream

At the MAP terminal

1 From the MAPCI level, enter table SBSFMT by typing:

>TABLE SBSFMT
and pressing the Enter key. ↵

2 Define the format characteristics of the billing stream using the following format:

>ADD key format
where

key	is the data stream (OCC).
format	is CDR250FMT.

Sample Entry:
>ADD occ cdr250fmt

3 Verify the datafill by typing:

>LIST
and pressing the Enter key. ↵

4 From the MAPCI level, enter table SBSMAP by typing:

>TABLE SBSMAP
and pressing the Enter key. ↵

5 Define the type of call recording data and the stream the data is routed to using the following format:

>ADD key stream
where

key	is the call data type.
stream	is the key used in table SBSFMT.

Sample Entry:
>ADD occ occ

5.1 Perform a restart reload.

5.2 Create a billing volume, as required.

5.3 Mount disk volume.

5.4 Verify changes by viewing table DRMPOOL.

6 The billing stream is set.

FCDRSRCH menu

The flexible call detail record search (FCDRSRCH) is used to read all billing records. In the UCS06 software release, FCDRSRCH replaces CDRSRCH (formerly used for records stored by the device independent recording package) and DCDRSRCH (formerly used for records stored by the Billing Server). Up to 20 FCDRSRCHs may be simultaneously performed on records stored by the Billing Server.

With FCDRSRCH, it is possible to issue menu commands to perform the following:

- select a billing file desired for searching
- define keys to compare against data in the call detail record (CDR) and operator services record (OSR) fields
- set a logical condition among the defined keys
- impose constraints on the scope of the search
- view the CDRs that are returned as a result of the search session

Initiating and using FCDRSRCH

To enter the FCDRSRCH menu from a MAP terminal, type the following:

>MAPCI;MTC;APPL;OAMAP;FCDRSRCH

Figure 8-1 illustrates the initial FCDRSRCH menu display before any commands have been issued.

Figure 8-1
FCDRSRCH menu display

```
CM    MS    IOD    NET    PM    CCS    TRKS    EXT    APPL

SBS
0  QUIT       OAMAP  EIO          SVMTC
2  SRCHfil_
3  CDRType_
4   LISTflds       DRM:      AFT:        MNP:      SBS: OK
5   SETKey
6   RSETkey
7   OPERator   FILE:
8   WINDow     BLOCK #:      #TO SRCH:         FORMAT:
9   BLOCk      OUTBUF POS:   #TO OUT           KEY OPER:
10  NUMSrch    WNDW START        WNDW STOP:         TYPE:
11  NUMOut
12  STATus
13  REINit
14  FORWard     FCDRSRCH:
15  BACKup
16  DISPlay
17
18  EXECsrch



      TIME  12:00
```

Menu commands for FCDRSRCH are separated into functional groups (Figure 8-2). Command 0 exits to the previous MAP level (in this case, MAPCI;MTC;APPL;OAMAP). Command 2 designates the billing file to search. Commands 3 through 11 define the search session parameters. Command 12 displays previous search status, and command 13 reinitializes search parameters. Commands 14 through 16 view the output buffer. Command 18 initiates the search session. A detailed description of each command is in the “CDR search menu commands and messages” section of this chapter.

The FCDRSRCH menu includes a status area (Figure 8-2) that provides search session feedback. The status of the search session and most of the search parameters are displayed in the status area. The blank fields are updated as menu commands are entered and as the search progresses.

Figure 8-2
FCDRSRCH menu detail

```
                              Basic CDR Maintenance
   CM       MS       IOD      Net      PM       CCS      Trks     Ext      APPL
   .        .        .        .        .        .        .        .        .

                         OAMAP    EIO                         SVMTC
  FCDRSRCH
  0   QUIT                         .                             .
  2   SRCHfil_              .
  3   CDRType_
  4   LSTflds  DRM:  .          AFT:  .             MNP:  .        SBS:   .
  5   SETKey_
  6   RSETkey_
  7   OPERator
  8   WINDow_  FILE:
  9   BLOCk_   R8504221205020CC      #TO SRCH:         FORMAT: CDR250
  10  NUMSrch_ BLOCK #:   0002       00052             KEY OPER:  A1
  11  NUMOut_  OUTBUF POS:           #TO OUT:  010               TYPE:
  12  STATus   WNDW START:              WNDW STOP:  1985042212
  13  REINit   1985042212
  14  FORWard_
  15  BACKup_
  16  DISPlay
  17
  18  EXECsrch

      TIME    12:00
        >
```

Search session parameters (items 3–11)

View output buffer (items 14–16)

Status area with entries

Note: The options used for this menu are examples only and differ between UCS DMS-250 switches.

Status fields

Status fields are divided into two functional categories: static (set with menu commands before start of search and not subject to change as search executes) and dynamic (change as search progresses or as output buffer updates).

Static status fields

Table 8-1 lists the static status fields, their description, and the command used to define the fields.

Table 8-1
Static status fields

Field name	Description	Command
FILE	Billing file name	SRCHfil
CDRTYPE	Call record format (CDR or OSR)	CDRType
KEY OPER	Search key logical operator (AND or OR)	OPERator
#TO SRCH	Number of records to search (optional)	NUMSrch
#TO OUT	Number of records to output (optional)	NUMOut
WNDW START	Beginning of time window (optional)	WINDow
WNDW STOP	End of time window (optional)	WINDow

Dynamic status fields

Table 8-2 lists the dynamic status fields. These fields are updated during the search.

Table 8-2
Dynamic status fields

Field name	Description	How it changes
BLOCK#	Current block number	SRCHfil or BLOCK command or when search reads a new data block
OUTBUF POS	Number of record being viewed in the output buffer	Set to one upon successful search, updated as output buffer is viewed.

FCDRSRCH commands

Table 8-3 lists the commands available for FCDRSRCH.

Table 8-3
FCDRSRCH commands

Command	Syntax	Description
BACKUP	BACKUP pos_backward	moves backward through the output buffer
BLOCK	BLOCK block#	sets starting block number
DISPLAY	DISPLAY display_rec	displays contents of output buffer
EXECSRCH	EXECSRCH	initiates search session
CDRTYPE	CDRTYPE cdrtype	selects the billing record (CDR or OSR)
FORWARD	FORWARD pos_forward	moves forward through the output buffer
LISTFLDS	LISTFLDS	lists valid file names for the selected CDR type
NUMOUT	NUMOUT recto_output	shows the number of records to output (optional)
NUMSRCH	NUMSRCH recto_search	shows the number of records to search (optional)
OPERATOR	OPERATOR	toggles logical condition (AND is default)
QUIT	QUIT	exits current menu level
REINIT	REINIT	reinitializes search parameters
RSETKEY	RSETKEY recfield recdata	edits current search key definitions
SETKEY	SETKEY recfield recdata	defines search keys
SRCHFIL	SRCHFIL stream block# or SRCHFIL path filename block#	selects desired billing file
STATUS	STATUS	provides information about last completed search
WINDOW	WINDOW begyear begmonth begday beghour begminute endyear endmonth endday endhour endminute	defines time window (a beginning time can be specified without an end time, but not vice versa)

Access modes

Before initiating a search, decide whether to use automatic or manual access mode.

Automatic mode

The automatic mode allows you to access the most recently rotated billing file directly without any knowledge of the billing file name or disk volume. The only information required is the billing stream; the block number is optional. The syntax is as follows:

>SRCHFIL stream block#

Note: DRM does not support the concept of parallel files; therefore, it is not necessary to specify active or parallel files in this mode.

Manual mode

To use the manual mode, you must know the disk on which the billing file resides. Before searching a file, choose a file to search. Use DISKUT or the DRM menu to get a list of valid files.

With the manual mode, you can access any billing file within the DMS file system. The block number is optional. The syntax is as follows:

>SRCHFIL path filename block#

Search guidelines

The asterisk (*) character is a wildcard character for the data item of a search key. Using an * as the only character in the key data means a match will always be made regardless of what is contained in the field. However, using an * at the end of the key data means that a match depends on only the characters that precede the *.

The command interpreter of the MAP terminal assumes that all numeric input of five characters or less is intended to be a number. Enclose all numeric input strings that are less than six characters with single quotation marks ('). This procedure is required only when specifying a numeric string for a key data item. In practice, it may be more convenient to use single quotes for all numeric strings.

Commands that accept one or more parameters are followed by the underscore character (_).

A prompt character indicates the feature is ready to accept input for one of the following:

- viewing output buffer
- altering search parameters (billing file, search keys, logical condition, or imposed constraints)
- initiating another search session

- exiting FCDRSRCH level of the MAP terminal

The EXECSRCH command reinitiates a search. A new search request erases any CDRs in the output buffer. Search parameters remain as defined, unless you update them. Therefore, you can access additional CDRs simply by changing the FORMAT parameters. Unless a new billing file is accessed or a new starting point is defined, a new search (by default) begins with the first record following the last referenced CDR.

To exit the FCDRSRCH menu, request a different MAP level or issue the QUIT command. In either case, the search process terminates and all temporary DMS system resources are free. You must define a new search each time you enter the FCDRSRCH level.

Search keys

Each field in the CDR can define search keys. To generate a list of key search fields, enter the following command:

>LISTFLDS

Specify one or more fields in the CDR to use as search keys. Use the following command to set the search parameters:

>SETKEY recfield recdata

Note 1: Searches based on CDID are limited to the last 12 bits (0–4095).

Note 2: If you specify multiple search keys, use the AND or OR boolean logic commands.

AND operation

Using the AND option means a match on every key must occur to return the CDR. The default option is AND. Therefore, if you do not specify a logical operator, the system adds the AND command between all keys defined.

OR operation

Using the OR option means only one key match must occur to return the CDR.

Search limits

The starting point for the search can be user-defined, as can the ending point. However, a search’s end point may be defined by the system. The following search limits include the user-defined and system-defined limits:

- user-defined limits
  - — time window with start and end time (defined by the WINDOW command)

- request a specific number of records to find (defined by the NUMSRCH command)
- request a specific number of records to output (defined by the NUMOUT command)
- start search at specified block (defined by the BLOCK command)

- system-defined limits
  - Output buffer filled – determined by the office parameter SRCHOUTPUT found in table OFCENG. When buffer is filled with matched records, and no other stopping conditions have been met, the search returns control to the user allowing results of the incomplete search session to be viewed.
  - Search error encountered – monitors problems with DMS system resources, ensures proper feature data fill, and maintains integrity of file system interaction. The search cannot fix problems, but does detect and report them. When an error is found, the search suspends itself and reports the error condition. The search continues to report errors upon each new search request until the problem has been cleared. Corrective action must be performed to eliminate problems reported by the search feature.
  - End of data on the billing file

If the search is successful, matched CDRs and OSRs may be displayed from within the output buffer by using the DISPLAY, BACKUP, and FORWARD commands.

FCDRSRCH menu commands

Appendix C, “FCDRSRCH menu commands,” describes the syntax, purpose, and semantics of the FCDRSRCH commands in alphabetical order.

User interface

The next section contains the procedure for performing a CDR/OSR search.

Search CDR/OSR records

Note: To search for either OSRs or CDRs, table CRSFMT, field CDRSRCH must be datafilled as UCSEOPS to enalbe CDR as a type choice.

User steps:

1 Enter the FCDRSRCH menu from MAP by typing:

>MAPCI;MTC;APPL;OAMAP;FCDRSRCH
and pressing the Enter key. ↵

Note: Commands may be entered by pressing the number corresponding to the command on the FCDRSRCH menu.

Example of a MAP response:

```
CM    MS    IOD    NET    PM    CCS    TRKS    EXT    APPL

FCDRSRCH
0  QUIT        OAMAP  EIO
2  SRCHfil_
3  CDRType_
4   LISTflds       DRM:      AFT:        MNP:      SBS:
5   SETKey
6   RSETkey
7   OPERator   FILE:
8   WINDow     BLOCK #:       #TO SRCH:         FORMAT:
9   BLOCk      OUTBUF POS:    #TO OUT           KEY OPER:
10  NUMSrch    WNDW START         WNDW STOP:         TYPE:
11  NUMOut
12  STATus
13  REINit
14  FORWard     FCDRSRCH:
15  BACKup
16  DISPlay
17
18  EXECsrch



      TIME  12:00
```

2 Select billing file by typing (block is optional):

>SRCHFIL stream block
and pressing the Enter key. ↵

Search CDR/OSR records (end)

3 OPTIONAL: Select CDR-type by typing:

>CDRType type
and pressing the Enter key. ↵

4 Define search keys by typing:

>SETKEY cdrfield cdrdata
and pressing the Enter key. ↵

5 OPTIONAL: Define search constraints using WINDOW, BLOCK, NUMOUT, or NUMSRCH commands.

6 Initiate search by typing:

>EXECSRCH
and pressing the Enter key. ↵

System steps:

1 Reads billing file.

2 Advances to requested or default starting point in file.

3 Compares each CDR against search parameters.

4 Appends matching CDRs to output buffer.

Note: To terminate execution of a search session, press the break key followed by HX or STOP, then press return.

5 Repeats steps 3–4 until end condition occurs.

6 Completes search and indicates success or failure.

Appendix A
User interface quick reference guide

In several of the procedures, the MAP terminal prompts the user for one of the actions in Table 9-1.

Table 9-1
User actions

Terminal prompt	User action
`Press <CR> to continue, or enter QUIT`	Press the enter key to continue the process or type QUIT to discontinue the process.
`Select a`	Enter the number of the file used in the process preceded by a space.

Table 9-2 lists the procedures, their full procedure location, and their shortened procedure location.

Table 9-2
List of procedures

Procedure	Full procedure location	Shortened procedure location
Activate Billing Server	7–17	9–4
Add Billing Server to DMS system and establish the MS-FP fiber links	4–2	9–4
Add DMS-Core, Billing Server-specific circuit packs to the MS	4–3	9–4
Add FLIS to DMS system	4–5	9–5

—continued—

Table 9-2
List of procedures (continued)

Procedure	Full procedure location	Shortened procedure location
Back up files to DAT	5–14	9–5
Back up files to to core IOC 9-track tape	5-16	9–5
Bring up file processor controllers and disks	5–19	9–6
Configure DRM settings	5–23	9–6
Create volumes (billing, FTAM, or OM)	5–25 (billing) 6–12 (FTAM)	9–7
Deactivate Billing Server	7–19	9–8
Define the EIUs	4–6	9–8
Define sequence of devices along the SCSI bus	4–8	9–8
Define shadow set	5–31	9–9
Define storage devices	4–9	9–9
Delete disk volumes (billing, FTAM, or OM)	5–39 (billing) 6–18 (FTAM)	9–10
Demount DAT	5–44	9–11
Demount a disk volume	5–46	9–11
Determine FTAM billing transfer	6–23	9–11
Determine what disks are in a shadow set	5–48	9–11
Determine which volumes are on the file processor	5–50	9–11
Ensure the MS is datafilled	4–10	9–11
Establish the inter-MS links	4–11	9–12
Format the disk	5–51	9–12
Manually rotate a DRM-managed file	5–55	9–13
Monitor the state of the active DRM file	5–57	9–13
Mount DAT	5–59	9–13
Mount disk volume	5–61	9–13

—continued—

Table 9-2
List of procedures (continued)

Procedure	Full procedure location	Shortened procedure location
Obtain file information	5–64	9–14
Perform a DRM audit	5–74	9–17
Rename a DRM-managed file	5–81	9–18
Reset volume counters	5–83	9–18
Restore files from DAT	5–85	9–18
Search CDR/OSR records	8–9	9–19
Set billing stream	7–21	9–19
Stop shadowing	5–92	9–20
Verify DAT contents	5–93	9–20
Verify firmware load is datafilled	4–12	9–20
Verify FP circuit pack and FP and device shelf power converter datafill	4–13	9–21
Verify software load is datafilled	4–15	9–22
View contents of a file	5–95	9–22
View mounted volumes	5–97	9–22

—end—

Procedures

Activate Billing Server

At the CI prompt

7 **>MAPCI;MTC;APPL;OAMAP;SBS**

8 **>ACT**

9 **>YES**

Billing Server is activated.

Add Billing Server to DMS system and establish the MS-FP fiber links

At the CI prompt

1 **>TABLE APINV**

2 **>ADD smntype smnno function loadname selfload floor row frame shelf shelfpec quadrant linkrate links $**

Sample entry: **>ADD fp 0 sbs fcc02bq n 1 a 6 0 nt9x81aa 0 r256 12 0 0 15 0 15 0 1 15 0 12 1 1 16 0 15 1 0 16 0 $**

Billing Server is added to the DMS system and the MS-FP fiber links are established.

Add DMS-Core, Billing Server-specific circuit packs to the MS

Note: Table MSINV must be datafilled before table MSCDINV.

At the CI prompt

1 **>TABLE MSCDINV**

2 Add the MS-FLIS link circuit packs to the MS using the following format:

>ADD mscdkey slotinfo

Sample entry: **>ADD 0 0 12 ds512 nt9x17ad nt9x62ba 1**

3 Add the first inter-MS link circuit packs to the MS using the following format:

>ADD mscdkey slotinfo

Sample entry: **>ADD 0 0 18 chain ds512 nt9x17da nt9x20bb $**

4 Repeat step 3 to add the second inter-MS link circuit pack.

5 Add the first MS-FP link circuit packs to the MS using the following format:

>ADD mscdkey slotinfo

Sample entry: **>ADD 0 0 20 chain ds512 nt9x17da nt9x62ba 2 $**

6 Repeat step 5 to add the second MS-FP link circuit pack.

DMS-Core, Billing Server-specific circuit packs are added to the MS.

Add FLIS to DMS system

Note: Table MSCDINV must be datafilled before table SUSHELF.

At the CI prompt

1 **>TABLE SUSHELF**

2 Add FLIS to the DMS system using the following format:

>ADD shelfkey floor row framepos frametyp framenum shelfpos shelfpec cardinfo

Sample entry: **>MS NIL 12 0 1 0 A 1 emc 0 26 nt9x7204 7 nt9x96aa nt9x98aa 30 nil ntex20aa $ 32 nt9x96aa nt9x98aa 8 nil ntex20aa $**

FLIS is added the DMS system.

Back up files to DAT

> **CAUTION**
> **Lost data**
> Do not attempt to back up active files to DAT. Active files cannot be backed up and data will be lost!

At the CI prompt

1 **>MAPCI;MTC;APPL;OAMAP;DRM**

2 Mount DAT by following the procedure entitled *Mount DAT*.

3 **>DAT WRITE stream filetype**

4 Select the file to back up.

5 Follow the screen instructions.

File is backed up to DAT.

Back up files to core IOC 9-track tape

At the CI prompt

1 **>MAPCI;MTC;APPL;OAMAP;DRM**

2 **>TCOPY 9track_dev applid filetmpl**

Sample entry: **TCOPY 4 occ u**

3 Select the file to back up.

4 Follow the screen instructions.

File is backed up to core IOC 9-track tape.

Bring up file processor controllers and disks

At the CI prompt

1 **> MAPCI;MTC;PM**

2 **> POST FP 0**

3 **> DEVICES**

4 **> BSY CTRL 0**
> BSY CTRL 1

5 **> RTS CTRL 0**
> RTS CTRL 1

6 **> BSY DEV 0 ALL**
> BSY DEV 1 ALL

7 **> RTS DEV 0 ALL**
> RTS DEV 1 ALL

FP controllers and disks are brought up.

Configure DRM settings

At the CI prompt

1 **>TABLE DRMAPPL**
Note: Table DRMAPPL does not allow additions, only changes to existing data.

2 **>LIST ALL**

3 **>POS applname**

4 **>CHANGE**

5 Enter the changes required as prompted by the system.

The user-defined DRM defaults are set.

Create volumes (billing, FTAM, or OM)

ATTENTION
Identical billing volumes must be created on each disk belonging to shadow set.

At the CI prompt

1 **> MAPCI;MTC;PM**

2 **> POST FP 0**

3 **> DEVICES**

4 **> BSY DEV scsi_bus# scsi_dev#**
Sample entry: **BSY DEV 0 0**

5 **> DISKADM devname nodetype nodenum**

Sample entry: **DISKADM dk00 fp 0**

6

If	**Do**
disk needs formatting	step 7
disk does not need formatting	step 9

7

CAUTION
Lost data
The following command causes all data on the disk to be lost.

> FD QUICK FORCE

8 **> YES**

9 **> CV volname volsize voltype [LBLOCK blksize]**

Sample entry: **CV occ 600 ftfs LBLOCK 16**

10 **> YES**

11 **> QUIT**

12 **> RTS DEV scsi_bus# scsi_dev#**

Volume is created.

Deactivate Billing Server

At the CI prompt

1. **>MAPCI;MTC;APPL;OAMAP;SBS**
2. **>ROUTECM DIRP OCC**
3. **>Yes**

   Billing Server is deactivated.

Define the EIUs

Note: There are no prerequisite datafill requirements for table APINV.

At the CI prompt

1. **>TABLE LIUINV**
2. Define the first EIU using the following format:

   >ADD liuname location load procinfo pfipec cardinfo

   Sample entry: **>EIU 102 MS 12 0 0 12 OSU02A0 NTEX22BB NT9X84AA NT9X85AA YES 0000075F00101**

3. Define the second EIU.

   Sample entry: **>EIU 208 MS 13 0 0 12 OSU02A0 NTEX22BB NT9X84AA NT9X85AA YES 0000075F00101**

   The EIUs are defined.

Define sequence of devices along the SCSI bus.

Note: Tables PMLOADS, APINV, and APCDINV must be datafilled before table FPDIPINV.

At the CI prompt

1. **>TABLE FPDIPINV**
2. Add the first device sequence using the following format:

   >ADD fpno scsibus sdipno shelf shelfpec floor row frame dipids

   Sample entry: **>ADD 0 0 0 3 nt9x83aa 1 a 6 0 8 nt9x89aa 1 9 nt9x89aa $**

3. Add the remaining device sequences:

   Sample entry: **>ADD 0 0 1 3 nt9x83aa 1 a 6 0 20 nt9x89aa 1 21 nt9x89aa $**
   >ADD 0 1 0 3 nt9x83aa 1 a 6 0 14 nt9x89aa 1 15 nt9x89aa $
   >ADD 0 1 0 3 nt9x83aa 1 a 6 0 26 nt9x89aa 1 27 nt9x89aa $

   Device sequences are defined.

Define shadow set

At the CI prompt

1 **> MAPCI;MTC;PM**

2 **> POST FP 0**

3 **> DEVICES**

4 **> BSY DEV scsi_bus# scsi_dev#**
for the master and pressing the Enter key. ↵

> BSY DEV scsi_bus# scsi_dev#
for the slave and pressing the Enter key. ↵

Sample entry: **BSY DEV 0 0**
BSY DEV 1 0

5 **> SHADOWUT nodetype nodenum**

Sample entry: **SHADOWUT fp 0**

6 **> SS setname [INSVSYNC]**

Sample entry: **SS ss1 INSVSYNC**

7 **> YES**

Sample entry: **AM ss1 dk10**

8 **> DIS setname**

Sample entry: **DIS ss1**

Note: The disks will return to service (RTS) when finished.

9 **> QUIT**

Shadow set is defined.

Define storage devices

Note: Tables APINV, APCDINV, and FPDIPINV must be datafilled before table FPDEVINV.

At the CI prompt

1 **>TABLE FPDEVINV**

2 **>ADD fpno scsibus devno devtype devpec scsiid quadno shelf shelfpec floor row frame**

Sample entry: **>ADD 0 1 5 dk nt9x90aa 3 0 0 nt9x83aa 0 r 1**

3 Repeat step 2 for each storage device.

The storage devices are defined.

Delete disk volumes (billing, FTAM, or OM)

> **CAUTION**
> **FTAM volume is required for normal file transfer**
> The FTAM volume is required by Billing Server for normal downstream file transfer.

At the CI prompt

1 **> MAPCI;MTC;PM**

2 **> POST FP 0**

3 **> DEVICES**

4 **> BSY DEV scsi_bus# scsi_dev#**

Sample entry: **BSY DEV 0 0**

5 **> DISKADM devname nodetype nodenum**

Sample entry: **DISKADM dk00 fp 0**

> **CAUTION**
> **Lost data**
> Verify that no data exists on the volume to be deleted. Confirming deletion causes all data on the disk to be lost. DEMOUNT volume takes it out of table DRMPOOL.

6 **> DDV volname**

Sample entry: **DDV occ**

7 **> YES**

8 **> QUIT**

9 **> RTS DEV scsi_bus# scsi_dev#**

Volume is deleted.

Demount DAT

At the CI prompt

1 **>MAPCI;MTC;APPL;OAMAP;DRM**

2 **>DAT EJECT**

DAT is mounted.

Demount a disk volume

At the CI prompt

1 **>MAPCI;MTC;APPL;OAMAP;DRM**

2 **>DEMOUNT stream volname**

Sample entry: **DEMOUNT occ fp0ss0occ5**

Disk volume is demounted.

Determine FTAM billing transfer

At the CI prompt

1 **>QUERYXFER**

Determine what disks are in a shadow set

At the CI prompt

1 **>SHADOWUT nodetype nodenum**

Sample entry: **SHADOWUT fp 0**

2 **>DIS**

3 **>DIS setname**

Sample entry: **DIS ss1**

Determine which volumes are on the file processor

At the CI prompt

1 **>DISKUT**

2 **>LV FP0**

3 **>QUIT**

Ensure the MS is datafilled

Note: There are no prerequisite datafill requirements for table APINV.

At the CI prompt

1 **>TABLE MSINV**

2 **>LIST ALL**

Establish the inter-MS links

> **ATTENTION**
> Inter-MS links are required for Billing Server to function.

Note 1: All data presented in this procedure is sample datafill only: the user should be familiar with data required at their site.

Note 2: Table MSCDINV must be datafilled before table MSILINV.

At the CI prompt

1 **>TABLE MSILINV**

2 Add the first inter-MS tuples using the following format:

>ADD imsl shelfno cardno info

Sample entry: **>ADD 0 0 18 DS512 0**

3 Add the second inter-MS link tuple:

Sample entry: **>ADD 1 0 19 DS512 0**

Inter-MS links are datafilled.

Format the disk

At the CI prompt

1 **> MAPCI;MTC;PM**

2 **> POST FP 0**

3 **> DEVICES**

4 **> BSY DEV scsi_bus# scsi_dev#**

Sample entry: **BSY DEV 0 0**

5 **> DISKADM devname nodetype nodenum**

Sample entry: **DISKADM dk00 fp 0**

6 **> FD QUICK FORCE**

7 **> YES**

8 See “Create billing volumes” procedure to allocate the disk.

Manually rotate a DRM-managed file

At the CI prompt

1 **>MAPCI;MTC;APPL;OAMAP;DRM**

2 **>ROTATE stream**

DRM-managed file is rotated.

Monitor the state of the active DRM file

At the CI prompt

1 **>MAPCI;MTC;APPL;OAMAP;DRM**

2 **>MONITOR stream**

Sample entry: **MONITOR occ**

Mount DAT

At the CI prompt

1 **>MAPCI;MTC;APPL;OAMAP;DRM**

2 **DAT MOUNT dat_drive nodename [INIT]**

Sample entry: **>DAT MOUNT ct11 fp0 INIT**

DAT is mounted.

Mount disk volume

At the CI prompt

1 **>MAPCI;MTC;APPL;OAMAP;DRM**

> **CAUTION**
> **Lost data**
> Do not mount any volume ending in the letters "FTAM" into DRM.

2 **>MOUNT stream volname nodename [priority]**

Sample entry: **MOUNT occ ss1occ5 fp0**

3 Discontinue the mounting process by typing:

If	Then
the operation successful response is received	the procedure is complete
an error message is received	continue to step 4

Note: Following is an example of an error message:

```
Volume SS1OCCX is not available.
Reason:  Device Unavailable

Perform Mount operation?  (Yes/No, or Quit)
```

Notice the X at the end of the volume name in the error message. This should be a number. The error response was received due to this spelling error.

4 **>No**

Disk volume is mounted.

Obtain file information

At the CI prompt

1 **>MAPCI;MTC;APPL;OAMAP;DRM**

2 **>INFO stream infotype [destnode]**

3 To obtain file information, choose one of the following and proceed to the indicated step:

To see this type of information,	Use this command	Sample entry
DRM nodes	INFO stream NODES	INFO occ NODES
application registration	INFO stream REG [destnode]	INFO occ REG
application audit schedules and next scheduled rotate time	INFO stream SCHED [destnode]	INFO occ SCHED fp0
application rotate information	INFO stream ROT [destnode]	INFO occ ROT fp0
application's mounted volumes, including status information	INFO stream VOL [volnode volume ACTV ALL] [destnode]	INFO occ VOL fp0

—continued—

To see this type of information,	Use this command	Sample entry
application’s mounted volumes, including trouble diagnostics	INFO stream VOLDIAG [volnode volume ACTV ALL] [destnode]	INFO occ VOLDIAG ACTV
application’s mounted volumes with a brief, textual description of the last error detected on each volume	INFO stream VOLERR [volnode volume ACTV ALL] [destnode]	INFO occ VOLERR ACTV
application’s mounted volumes, along with its files	INFO stream VOLFIL [volnode volume ACTV ALL] [filetmpl fstates ALL] [destnode]	INFO occ VOLFIL 0 u fp0
application’s files and file information	INFO stream FIL [volnode volume ACTV ALL] [filetmpl fstates ALL] [destnode]	INFO occ FIL fp0 cm
—end—		

4 >INFO stream NODES

Sample entry: **INFO occ NODES**

DRM node information is obtained and procedure is complete.

5 >INFO stream REG [destnode]

Sample entry: **INFO occ REG**

Registration information is obtained and procedure is complete.

6 >INFO stream SCHED [destnode]

Sample entry: **INFO occ SCHED fp0**

Schedule information is obtained and procedure is complete.

7 >INFO stream ROT [destnode]

Sample entry: **INFO occ ROT fp0**

Rotate information is obtained and procedure is complete.

8 >INFO stream VOL [volnode volume ACTV ALL] [destnode]

Sample entry: **INFO occ VOL fp0**

Mounted volume information is obtained and procedure is complete.

9 **>INFO stream VOLDIAG [volnode volume ACTV ALL] [destnode]**

Sample entry: **INFO occ VOLDIAG ACTV**

Mounted volume trouble diagnostic information is obtained and procedure is complete.

10 **>INFO stream VOLERR [volnode volume ACTV ALL] [destnode]**

Sample entry: **INFO occ VOLERR ACTV**

Mounted volume error information is obtained and procedure is complete.

11 View mounted volume file information by typing:

>INFO stream VOLFIL [volnode volume ACTV ALL] [filetmpl fstates ALL] [destnode]

Sample entry: **INFO occ VOLFIL 0 u fp0**

Mounted volume file information is obtained and procedure is complete.

12 **>INFO stream FIL [volnode volume ACTV ALL] [filetmpl fstates ALL] [destnode]**

Sample entry: **INFO occ FIL fp0 cm**

File information is obtained and procedure is complete.

Perform a DRM audit

> **ATTENTION**
> This process is CPU intensive. Use sparingly.

At the CI prompt

1 **>MAPCI;MTC;APPL;OAMAP;DRM**

2

If	Do
all audits are required	step 3
a recording audit is required	step 4
a space audit is required	step 5
a volume audit is required	step 6
an application audit is required	step 7
a DRMFDIR audit is required	step 8

3 Initiate all DRM audits by typing:

>AUDIT stream ALL

Sample entry: **AUDIT occ ALL**

All DRM audits are performed and the procedure is complete.

4 **>AUDIT applid REC [ACTN] [nodename]**

Sample entry: **AUDIT occ REC fp0**

The recording audit is performed and the procedure is complete.

5 **>AUDIT applid SPACE [ACTN] [nodename]**

Sample entry: **AUDIT occ SPACE fp0**

The space audit is performed and the procedure is complete.

6 **>AUDIT applid VOL volume [ACTN] [nodename]**

Sample entry: **AUDIT occ VOL 0 fp0**

The volume audit is performed and the procedure is complete.

7 **>AUDIT applid APPL [ACTN] [nodename]**

Sample entry: **AUDIT occ APPL fp0**

The application audit is performed and the procedure is complete.

8 **>AUDIT applid FDIR [ACTN] [nodename]**

Sample entry: **AUDIT occ FDIR fp0**

The FDIR audit is performed and the procedure is complete.

Rename a DRM-managed file

At the CI prompt

1 Access the DRM level of the MAP terminal by typing:

>MAPCI;MTC;APPL;OAMAP;DRM

2 Rename the file to be removed by typing:

>RENAME stream currenttype newtype

Sample entry: **RENAME occ u p**

DRM-managaed file is renamed.

Reset volume counters

At the CI prompt

1 **>MAPCI;MTC;APPL;OAMAP;DRM**

2 **>RESET stream counter**

Sample entry: **RESET occ err**

Volume counter is reset.

Restore files from DAT

At the CI prompt

1 Mount DAT by following the procedure entitled, *Mount DAT*.

Note: Each file type must be restored from DAT separately.

2 **>DAT RESTORE filetype [newstate]**

Sample entry: **>DAT RESTORE u**

CAUTION
Lost data
If a response appears asking if the file should be overwritten, enter **NO**. Overwriting files can cause loss of data.

3 Demount DAT by following the procedure entitled *Demount DAT*.

Search CDR/OSR records

Note: To search for OSRs, table CRSFMT, field CDRSRCH must be datafilled as UCSEOPS.

User steps:

1 **>MAPCI;MTC;APPL;OAMAP;FCDRSRCH**
Note: Commands may be entered by pressing the number corresponding to the command on the FCDRSRCH menu.

2 **>SRCHFIL stream block**

3 **>CDRType type**

4 **>SETKEY cdrfield cdrdata**

5 OPTIONAL: Define search constraints using WINDOW, BLOCK, NUMOUT, or NUMSRCH commands.

6 **>EXECSRCH**

System steps:

1 Reads billing file.

2 Advances to requested or default starting point in file.

3 Compares each CDR against search parameters.

4 Appends matching CDRs to output buffer.

Note: To terminate execution of a search session, press the break key followed by HX or STOP, then press return.

5 Repeats steps 3–4 until end condition occurs.

6 Completes search and indicates success or failure.

Set billing stream

At the MAP terminal

1 **>TABLE SBSFMT**

2 **>ADD key format**

Sample entry: **ADD occ cdr250fmt**

3 **>TABLE SBSMAP**

4 **>ADD key stream**

Sample entry: **ADD occ occ**

The billing stream is set.

Stop shadowing

At the CI prompt

1 **> SHADOWUT nodetype nodenum**

Sample entry: **SHADOWUT fp 0**

2 **> STS setname**

Sample entry: **STS fp0ss0**

3 Enter Y to stop shadowing.

Shadowing is stopped.

Verify DAT contents

At the CI prompt

1 Mount DAT by following the procedure entitled *Mount DAT*.

2 **>MAPCI;MTC;APPL;OAMAP;DRM**

3 **>DAT VERIFY**

5 Demount DAT by following the procedure entitled *Demount DAT*.

Verify firmware load is datafilled

Note 1: The NT9X17DA cards on the MS require firmware.

Note 2: There are no prerequisite datafill requirements for table APINV.

At the CI prompt

1 **>TABLE MSFWLOAD**

2 **>LIST ALL**

Verify FP circuit packs and FP and device shelf power converter datafill

Note 1: The FLIS circuit packs are datafilled in table SUSHELF.

Note 2: Table APINV must be datafilled before table APCDINV.

At the CI prompt

1 **>TABLE APCDINV**

2 Check the default data supplied by the system when the FP was added to table APINV by typing:

>LIST ALL

Example of a MAP response:

```
SMNTYPE SMNNO SHELF SLOT FRONTCRD FRONTPEC BACKCRD BACKPEC
----------------------------------------------------------
     FP      0      0    1    POWER NT9X15AB     NIL NIL
     FP      0      0    4    POWER NT9X15AB     NIL NIL
     FP      0      0    8   MEMORY NT9X14DB    TERM NT9X21AB
     FP      0      0    9   MEMORY NT9X14DB     NIL NIL
     FP      0      0   10   MEMORY NT9X14DB    PORT NT9X88AA
     FP      0      0   11   MEMORY NT9X87AA    PORT NT9X88AA
     FP      0      0   12     PORT NT9X86AA    PORT NT9X62AA
     FP      0      0   13      CPU NT9X13LA     TIF NT9X26AB
     FP      0      0   14      CPU NT9X13LA     TIF NT9X26AB
     FP      0      0   15     PORT NT9X86AA    PORT NT9X62AA
     FP      0      0   16   MEMORY NT9X87AA    PORT NT9X88AA
     FP      0      0   17   MEMORY NT9X14DB    PORT NT9X88AA
     FP      0      0   18   MEMORY NT9X14DB     NIL NIL
     FP      0      0   19   MEMORY NT9X14DB    TERM NT9X21AB
     FP      0      0   33    POWER NT9X15AA     NIL NIL
     FP      0      0   36    POWER NT9X15AA     NIL NIL
     FP      0      2    1    POWER NT9X15AA     NIL NIL
     FP      0      2    4    POWER NT9X15AA     NIL NIL
     FP      0      2   33    POWER NT9X15AA     NIL NIL
     FP      0      2   36    POWER NT9X15AA     NIL NIL
     FP      0      3    1    POWER NT9X15AA     NIL NIL
     FP      0      3    4    POWER NT9X15AA     NIL NIL
     FP      0      3   33    POWER NT9X15AA     NIL NIL
     FP      0      3   36    POWER NT9X15AA     NIL NIL
```

3 Change or add datafill as required using the change and add commands with the following format:

>ADD smntype smnno shelf slot frontcrd frontpec backcrd backpec
or
>CHANGE smntype smnno shelf slot frontcrd frontpec backcrd backpec

Sample entry: **>ADD fp 0 3 1 power nt9x90aa nil nil**

Sample entry: **>POS fp 0 0 10**
>CHANGE memory nt9x14db port nt9x88aa

Datafill is verified and corrected.

Procedures (end)

Note: There are no prerequisite datafill requirements for table APINV.

At the CI prompt

1 **>TABLE PMLOADS**

2 **>LIST ALL**

View contents of a file

At the MAP terminal

1 Access the DRM level of the MAP terminal by typing:

>MAPCI;MTC;APPL;OAMAP;DRM

2 View the contents of a file by typing:

>VIEW stream filetmpl charcode blocknum

Sample entry: **VIEW occ u as 1**

3 Follow the instructions on the screen to see more data or to quit the process.

View mounted volumes

At the MAP terminal

1 **>TABLE DRMPOOL**
Note: Table DRMPOOL does not allow any modifications.

2 **>LIST ALL**

Appendix B
Billing Server data schema

This appendix describes the data schema tables required for Billing Server. Tables are presented in alphabetical order.

The following table shows the data schema tables that must be datafilled for an office equipped with Billing Server:

Table 10-1
Related data schema tables

Short name	Long name
APCDINV	Application Processor Card Inventory
APINV	Application Processor Inventory
DRMAPPL	Distributed Recording Manager Application
DRMPOOL	Distributed Recording Manager Pool
FPDEVINV	File Processor Device Inventory
FPDIPINV	File Processor Device Interface Paddle Board Inventory
LIUINV	Link Interface Unit Inventory
MSCDINV	Message Switch Cards Inventory
MSFWLOAD	Message Switch Firmware Load
MSILINV	Message Switch Inter-MS Link Inventory
OSIPARMS	Open Systems Interconnect Parameters
OSIROUTE	Open System Interconnect Route
PMLOADS	Peripheral Module Loads
SBSFMT	SuperNode Billing System Format
SBSMAP	SuperNode Billing System Mapping

—continued—

Table 10-1
Related data schema tables (continued)

Short name	Long name
SHADOW	Shadow Sets
SUSHELF	Service Unit Shelf

—end—

Acronyms used in this section

The following acronyms are used in this appendix.

Table 10-2
Acronyms

Acronym	Meaning
CDR	call detail record
CI	command interpreter
CPU	central processing unit
DAT	digital audio tape
DDU	disk drive unit
DIRP	device independent recording package
DRM	distributed recording manager
EIU	Ethernet Interface Unit
FLIS	Fiberized Link Interface Shelf
FP	file processor
LIU	Link Interface Unit
INM	integrated node maintenance
MAC	media access control
MS	message switch
OCC	other common carrier
OSI	open systems interconnection
PEC	product engineering code

—continued—

Table 10-2
Acronyms (continued)

Acronym	Meaning
PM	peripheral module
RP	resource processor
RTIF	remote terminal interface
SCSI	small computer standard interface
SDIP	SCSI device interface paddle boards
SLM	system load module
SSLPP	single shelf link peripheral processor

—end—

APCDINV

Table name

Application Processor Card Inventory (APCDINV)

> **ATTENTION**
>
> The material found in this module regarding table APCDINV is UCS DMS-250 Billing Server-specific information. Refer to the *DMS-100 Customer Data Schema Reference Manual* for further details regarding datafill for table APCDINV.

Functional description

Table APCDINV is a look-up table containing a description of the FP circuit packs, as well as entries for all power converters located in the application processor cabinet.

Note: Table APCDINV is automatically filled with default data when the FP is added to table APINV. However, the user should verify the data in table APCDINV.

When an entry is added to table APINV for an FP, the following default tuples are added to table APCDINV:

- One tuple is added for each FP or device shelf power converter.
- One tuple is added for each front CPU card and RTIF paddleboard combination.
- One tuple is added for each front port card and port paddleboard combination.
- One tuple is added for each mandatory memory card.
- One tuple is added for each of the two slots containing bus terminators.
- One tuple is added for each front dual access buffer memory card and SCSI paddleboard combination.

Note: The tuples added by table APINV are deleted when the FP tuple is deleted from table APINV.

The following circuit packs and power converters should be included in the datafill for table APCDINV:

FP hardware	Front PEC	Physical slot	Back PEC
Global power converter ±5v	NTDX15AB	1–3	
Global power converter ±5v	NTDX15AB	4–6	
Memory 24 Mbyte circuit pack with bus terminator paddleboard	NT9X14DB	08	NT9X21AB
24 MByte memory circuit pack	NT9X14DB	09	
Memory 24 Mbyte circuit pack with SCSI interface paddleboard	NT9X14DB	10	NT9X88AA
Dual access buffer memory circuit pack with SCSI interface paddleboard	NT9X87AA	11	NT9X88AA
Dual port message controller circuit pack with 2 port subrate DS512 paddleboard	NT9X86AA	12	NT9X62AA
68030/40 processor circuit pack with remote terminal interface paddleboard	NT9X13LA	13	NT9X26AB
68030/40 processor circuit pack with remote terminal interface paddleboard	NT9X13LA	14	NT9X26AB
Dual port message controller circuit pack with 2 port subrate DS512 paddleboard	NT9X86AA	15	NT9X62AA
Dual access buffer memory circuit pack with SCSI interface paddleboard	NT9X87AA	16	NT9X88AA
Memory 24 Mbyte circuit pack with SCSI interface paddleboard	NT9X14DB	17	NT9X88AA
24 MByte memory circuit pack	NT9X14DB	18	
Memory 24 Mbyte circuit pack with bus terminator paddleboard	NT9X14DB	19	NT9X21AB
+5 /+12 V power converter	NT9X91AA		

Datafill sequence and implications

The following tables must be datafilled before table APCDINV:

- MSCDINV
- PMLOADS
- APINV

Table sizing

Memory for table APCDINV is dynamically defined. The maximum size of table APCDINV is 2800 tuples.

Field descriptions

The following table describes field names, subfield names, and valid data ranges for table APCDINV.

Field	Subfield or refinement	Entry	Explanation and action
SMNTYPE		FP	SYNC-MATCHED NODE TYPE. Enter FP to indicate a file processor. This is the first of a four-part key.
SMNNO		0 to 99	SYNC-MATCHED NODE INSTANCE NUMBER. Enter the instance number of the resource processor. This is the second of a four-part key.
SHELF		0	SHELF NUMBER. Enter the shelf number containing the FP.
SLOT		1 to 38	SYSTEM SLOT. Enter the circuit packs physical slot number. This is the fourth of a four-part key.
FRONTCRD		CPU MEMORY PORT POWER NIL	FRONT CARD ELEMENT CLASS. Enter the element classification of the card located in the front side of the card slot. Enter NIL if the card slot is empty or has a filler card.

—continued—

Field	Subfield or refinement	Entry	Explanation and action
FRONTPEC			FRONT CARD PEC. Enter the PEC for the card datafilled in field FRONTCRD.
		NT9X13LA	If the entry in field FRONTCRD is CPU, enter NT9X13LA.
		NT9X14DA NT9X14DB NT9X87AA	If the entry in field FRONTCRD is MEMORY, enter NT9X14DA, NT9X14DB, or NT9X87AA.
		NT9X86AA	If the entry in field FRONTCRD is PORT, enter NT9X86AA.
		NTDX15AA NT9X91AA	If the entry in field FRONTCRD is POWER, enter NT9X91AA or NTDX15AA. Any entry other than NT9X91AA or NTDX15AA is invalid.
		NIL	If the entry in field FRONTCRD is NIL, enter NIL (the card slot is empty or has a filler card).
BACKCRD		PORT, TERM, TIF, or NIL	BACK CARD ELEMENT CLASS. Enter the element classification of the card located on the back side of the card slot.
BACKPEC			BACK CARD PEC. Enter the PEC for the card datafilled in field BACKCRD.
		NT9X62AA NT9X88AA	If the entry in field BACKCRD is PORT, enter NT9X62AA or NT9X88AA.
		NT9X21AB	If the entry in field BACKCRD is TERM, enter NT9X21AB.
		NT9X26AA NT9X26AB NT9X26CA	If the entry in field BACKCRD is TIF, enter NT9X26AA, NT9X26AB, or NT9X26CA.
		NIL	If the entry in field BACKCRD is NIL, enter NIL (the card slot is empty or has a filler card).

—end—

Datafill example

The following example shows sample datafill for table APCDINV.

SMNTYPE	SMNNO	SHELF	SLOT	FRONTCRD	FRONTPEC	BACKCRD	BACKPEC
FP	0	0	1	POWER	NT9X15AB	NIL	NIL
FP	0	0	4	POWER	NT9X15AB	NIL	NIL
FP	0	0	8	MEMORY	NT9X14DB	TERM	NT9X21AB
FP	0	0	9	MEMORY	NT9X14DB	NIL	NIL
FP	0	0	10	MEMORY	NT9X14DB	PORT	NT9X88AA
FP	0	0	11	MEMORY	NT9X87AA	PORT	NT9X88AA
FP	0	0	12	PORT	NT9X86AA	PORT	NT9X62AA
FP	0	0	13	CPU	NT9X13LA	TIF	NT9X26AB
FP	0	0	14	CPU	NT9X13LA	TIF	NT9X26AB
FP	0	0	15	PORT	NT9X86AA	PORT	NT9X62AA
FP	0	0	16	MEMORY	NT9X87AA	PORT	NT9X88AA
FP	0	0	17	MEMORY	NT9X14DB	PORT	NT9X88AA
FP	0	0	18	MEMORY	NT9X14DB	NIL	NIL
FP	0	0	19	MEMORY	NT9X14DB	TERM	NT9X21AB
FP	0	0	33	POWER	NT9X15AA	NIL	NIL
FP	0	0	36	POWER	NT9X15AA	NIL	NIL
FP	0	1	1	POWER	NT9X15AA	NIL	NIL
FP	0	1	4	POWER	NT9X15AA	NIL	NIL
FP	0	1	33	POWER	NT9X15AA	NIL	NIL
FP	0	1	36	POWER	NT9X15AA	NIL	NIL
FP	0	2	1	POWER	NT9X15AA	NIL	NIL
FP	0	2	4	POWER	NT9X15AA	NIL	NIL
FP	0	2	33	POWER	NT9X15AA	NIL	NIL
FP	0	2	36	POWER	NT9X15AA	NIL	NIL
FP	0	3	1	POWER	NT9X15AA	NIL	NIL
FP	0	3	4	POWER	NT9X15AA	NIL	NIL
FP	0	3	33	POWER	NT9X15AA	NIL	NIL
FP	0	3	36	POWER	NT9X15AA	NIL	NIL

Table name

Application Processor Inventory APINV

> **ATTENTION**
> The material found in this module regarding table APINV is UCS DMS-250 Billing Server-specific information. Refer to the *North American DMS-100 Customer Data Schema Reference Manual* for further details regarding datafill for table APINV.

Functional description

Table APINV contains information identifying the type and instance of a file processor, as well as its location. Link and software configuration data required by the RP MAP facility and the Integrated Node Maintenance (INM) software is also datafilled in table APINV.

Datafill sequence and implications

The following tables must be datafilled before table APINV:

- PMLOADS

  Note: The entry in field LOADNAME of table APINV must match the entry in field LOADNAME of table PMLOADS.

- MSCDINV

Table sizing

Memory for table APINV is dynamically defined. The maximum size of table APINV is 200 tuples.

Field descriptions

The following table describes field names, subfield names, and valid data ranges for table APINV.

Field	Subfield or refinement	Entry	Explanation and action
SMNTYPE		FP	SYNC-MATCHED NODE TYPE. Enter FP to indicate a file processor. This is the first of a two-part key.
SMNNO		0 to 99	SYNC-MATCHED NODE INSTANCE NUMBER. Enter the instance number of the FP. This is the second of a two-part key.
FUNCTION		up to 12 alphanumeric characters	SYNC-MATCHED NODE FUNCTION. Enter a string of characters to identify the function of the FP.
LOADNAME		up to 8 alphanumeric characters	DEFAULT LOAD FILE NAME. Enter the name of the file loaded into the FP by system-initiated reload recovery operations. This is also the loadfile used by default when a user issues the LOADPM command at the PM level of a MAP. ***Note:*** This file must be datafilled in table PMLOADS and must reside on a permanent device (not a tape device).
SELFLOAD		Y or N	SELF-LOADING CAPABILITY. Enter Y (yes) to indicate the FP has self-loading capability when a system recovery action is initiated or a user issues the LOADPM command at the PM level of a MAP. Otherwise, enter N (no).
FLOOR		0 to 99	FLOOR POSITION. Enter the floor number where the cabinet containing the FP is located.
ROW		A to Z or AA to ZZ (except I, O, II, and OO)	ROW POSITION. Enter the row position of the cabinet containing the FP.
FRAME		0 to 99	FRAME POSITION. Enter the frame position in the row where the FP is located.

—continued—

Field	Subfield or refinement	Entry	Explanation and action
SHELF		0 to 3	SHELF NUMBER. Enter the shelf number containing the FP. Shelves are numbered from bottom to top, beginning with 0 (zero).
SHELFPEC		NT9X81AA	SHELF PEC. Enter the PEC of the shelf containing the FP.
QUADRANT		0 to 3	QUADRANT RANGE. Enter the quadrant range in which the RP is located. An FP can be located in quadrant ranges 0 to 1 or 2 to 3. Enter 0 (zero) or 1 for the first quadrant range. Enter 2 or 3 for the second quadrant range.
LINKRATE		R128 or R256	LINK RATE. Enter the bandwidth (number of channels) on the fiber connecting the FP to the DMS-bus.
LINKS		see subfields	SMN PORT CONFIGURATION. This field is a vector that consists of one or more entries IN subfields TCARDNO, TLINKNO, DNODENO, DCARDNO, and DLINKNO. Each subfield describes the location of one link on the node and the location at which the link is attached on the DMS-bus. Terminate the vector with a $.
	TCARDNO	1 to 38	TERMINATING CARD NUMBER. Enter the physical slot location of the FP paddleboard.
	TLINKNO	0 to 1	TERMINATING LINK NUMBER. Enter the terminating link number (0 or 1)
	DNODENO	0 to 1	DMS-BUS NODE NUMBER. Enter the MS node number (0 or 1).
	DCARDNO	1 to 26	DMS-BUS CARD NUMBER. Enter the physical slot location of the MS paddleboard.
	DLINKNO	0 to 7	DMS-BUS LINK NUMBER. Enter the link number of the destination link on the card.

—end—

APINV (end)

Datafill example

The following example shows sample datafill for table APINV.

```
SMNTYPE   SMNNO    FUNCTION  LOADNAME   SELFLOAD   FLOOR
ROW   FRAME  SHELF
SHELFPEC QUADRANT LINKRATE
       LINKS
________________________________________________________

 AP         0        LTS      LTS28IS        N        3
R       7     1
NT9X81AA    0     R256
(12 0 0 15 0) (15 0 1 15 0) (12 1 1 16 0) (15 1 0 16 0)
$
```

Table name

Distributed Recording Manager Applications (DRMAPPL)

Functional description

Table DRMAPPL associates an application with a pool of volumes and specifies operational parameters. When an application registers with the distributed recording manager (DRM), tables DRMPOOL and DRMAPPL are datafilled automatically. Tuples cannot be added to table DRMAPPL from the command interpreter (CI), but values can be changed.

Datafill sequence and implications

When an application registers, tables DRMPOOL and DRMAPPL are automatically datafilled.

Table sizing

Table size is defined automatically by the system.

Field descriptions

The following table describes field names, subfield names, and valid data ranges for table DRMAPPL.

Field	Subfield or refinement	Entry	Explanation and action
APPLNAME		alphanumeric (1 to 4 characters)	APPLICATION NAME. This field specifies the application name, which serves as the key to table DRMAPPL.
GROUPID		NIL, DRMTOOL, DRMTRANS, SBS, SOM, FILESEG, GSM	GROUP ID. This field specifies the group id for the application name.
APPLID		0 to 23	APPLICATION ID. This field specifies the application ID number.

—continued—

Field	Subfield or refinement	Entry	Explanation and action
POOLNAME		alphanumeric (up to 8 characters)	POOL NAME. Enter a pool name to associate a pool of volumes with the application. This entry must be identical to the corresponding entry in table DRMPOOL, since this field is used as the key into table DRMPOOL. Only one application can use a given pool name (pools cannot be shared by applications).
ALARMMN		0 to 32 767	ALARM MINOR. Enter the minor alarm threshold value, in megabytes. If the amount of total free volume space falls below the threshold value, a minor alarm is generated. The recommended and default minor alarm value is 50.
ALARMMJ		0 to 32 767	ALARM MAJOR. Enter the major alarm threshold value, in megabytes. If the amount of total free volume space falls below the threshold value, a major alarm is generated. The recommended and default major alarm value is 20.
ALARMCR		0 to 32 767	ALARM CRITICAL. Enter the critical alarm threshold value, in megabytes. If the amount of total free volume space falls below the threshold value, a critical alarm is generated. The recommended and default critical alarm value is 1.
RETPD		0 to 365	RETENTION PERIOD IN DAYS. Enter a value to specify the file retention period, in days. A file is considered expired if the file has been on the disk for longer than the retention period, since the filedate (the date in the filename). After this period, processed files are erased. See field FORCBKUP on page B–15.
FILEDATE		OPENED or CLOSED	FILE DATE. Enter OPENED to date the file when it is opened. Enter CLOSED to date the file when it is no longer the active file.

—continued—

Field	Subfield or refinement	Entry	Explanation and action
SHEDDAYS		vector of 7 values (Y or N)	SCHEDULED ROTATION DAYS. Enter Y (yes) for each day of the week, Monday through Sunday, that a rotation is to occur. Enter N (no) for each day of the week, Monday through Sunday, that a rotation is not to occur. Fields SHEDDAYS, SHEDBASE, and SHEDINCR together are used to define a scheduled rotation for the application.
SHEDBASE		0 to 23	SCHEDULED ROTATION BASE. Enter a value from 0 to 23 to indicate the hour of the day in which the first rotation is to occur. More than one rotation can be scheduled per day. See field SHEDINCR.
SHEDINCR		X1, X2, X3, X4, X6, X8, X12, X24, or NOROTATE	SCHEDULED ROTATION INCREMENTS. Enter a value representing the number of hours between scheduled rotations, using the first rotation as a base. Enter NOROTATE if no rotation is scheduled.
CLOSTATE		P, R, or U	FILE STATE WHEN CLOSED. This field specifies the state to which the closed file must go. Enter P for processed state, R for removed state, or U for unprocessed state.
MAXFSIZE		O to 600	MAXIMUM FILE SIZE. Enter a value to define the maximum size for distributed recording manager (DRM) files, in megabytes. This value is the maximum size enabled by DRM for files in the application defined in this tuple. If 0 (zero) is entered, the file size is limited only by the volume size.
FORCBKUP		Y or N	FORCE BACKUP ONTO DAT TAPES. Enter Y (yes) to instruct the DRM system audit to erase processed files (P–files) that have expired, only if the files have been backed up to digital audio tape (DAT). Enter N (no) to instruct the DRM system audit to erase P-files that have expired.

—end—

Datafill example

The following example shows datafill for table DRMAPPL.

```
APPLNAME GROUPID APPLID POOLNAME ALARMMN ALARMMJ ALARMCR
RETPD FILEDATE SHEDDAYS    SCHEDBASE  SHEDINCR  CLOSTATE
MAXFSIZE FORCBKUP

________________________________________________________
AMA      GROUPID   10    AMAPOOL   50      20      1
30    OPENED    YYYYYYY     3           X24       U
100         Y
```

DRMPOOL

Table name

Distributed Recording Manager Pool (DRMPOOL)

ATTENTION
The material found in this module regarding table DRMPOOL is UCS DMS-250 Billing Server-specific information. Refer to the *DMS-100 Customer Data Schema Reference Manual* for further details regarding datafill for table DRMPOOL.

Functional description

Table DRMPOOL defines pools of recording volumes for assignment to registered applications. When an application is registered with DRM during installation, default tuples for the application are automatically datafilled in tables DRMPOOL and DRMAPPL.

Table DRMPOOL assigns recording volumes to a pool name; table DRMAPPL assigns the pool name to the application. The recording volumes datafilled in table DRMPOOL cannot be added to or deleted through use of the table editor. Instead, the DRM commands MOUNT or DEMOUNT must be used to add or delete recording volumes in table DRMPOOL.

Datafill sequence and implications

When an application registers, tables DRMPOOL and DRMAPPL are automatically datafilled.

Table sizing

Table size is defined automatically by the system.

DRMPOOL (end)

Field descriptions

The following table describes field names, subfield names, and valid data ranges for table DRMPOOL.

Field	Subfield or refinement	Entry	Explanation and action
POOLNAME		up to 8 alphanumeric characters	POOL NAME. This is the key field of the table and contains the volume pool name.
VOLUME0–17		up to 20 alphanumeric characters or $	VOLUMES 0 TO 17. Fields VOLUME0 to VOLUME17 specify the volumes assigned to the pool. Each field contains a volume name; $ indicates that no volume is assigned to that position.

Datafill example

The following example shows sample datafill for table DRMPOOL.

```
FIELD     ENTRY
________________________________________

POOLNAME  OCCPOOL
VOLUME0   FP00DK00VOLUME01
VOLUME1   $
```

FPDEVINV

Table name

File Processor Device Inventory (FPDEVINV)

ATTENTION

The material found in this module regarding table FPDEVINV is UCS DMS-250 Billing Server-specific information. Refer to the *DMS-100 Customer Data Schema Reference Manual* for further details regarding datafill for table FPDEVINV.

Functional description

Table FPDEVINV contains a description of the devices installed on a DMS SuperNode FP. The description identifies the FP, the type of SCSI device, the location, and the hardware configuration required to fully identify a particular device. Table FPDEVINV is used to dynamically add and delete FP SCSI devices.

Table APINV contains the configuration data required by the FP. Table APCDINV contains the descriptions of all FP cards except those applicable to mass storage devices. Table FPDIPINV contains the configuration data for the FP device interface paddle boards.

The following rules apply when adding, deleting, or changing data entries in table FPDEVINV:

- Each new entry to table FPDEVINV is verified by the FP file maintenance software before a tuple is added to the table. The verification confirms that
  - — The processor portion of the FP is datafilled in table APINV.
  - — The shelf and quadrant is not already equipped with a device.
  - — The device PEC agrees with the device type (this is not automatically checked by the data dictionary).
  - — The MAP identifier on the specified SCSI bus is not a duplicate.
  - — The associated SCSI device interface paddle boards (SDIP) are datafilled in FPDIPINV.
  - — The SCSI identifier on the specified SCSI bus is not a duplicate.

- A tuple not meeting the above rules can still be added, but will result in warning messages for each datafill error. A tuple containing errors can be accepted but missing inventory information makes the FP device inoperable.
- The only entry that can be changed is the device PEC.
- Before a device can be deleted from the datafill it must be taken offline. If the maintenance software is not able to obtain device status, the deletion request is rejected.

Datafill sequence and implications

The following tables must be datafilled before table FPDEVINV.

- PMLOADS
- APINV
- APCDINV
- FPDIPINV

Table size

The FPDEVINV table size is 0 to 6400 tuples.

The maximum table size is set by calculating the number of possible tuples according to the following formula:

$$(\text{max FP}) \times (\text{max device types}) \times (\text{max device numbers}) = \text{possible tuples}$$

$$100 \times 4 \times 12 = 4800$$

Memory for table FPDEVINV is dynamically allocated.

Field descriptions

The following table describes field names, subfield names, and valid data ranges for table FPDEVINV.

Field	Subfield or refinement	Entry	Explanation and action
FPNO		0 to 99	FP INSTANCE NUMBER. Enter the instance number of the FP. This is an identifier that distinguishes one FP from another. This is the first of a three-part key.
SCSIBUS		0 to 1	SCSI BUS NUMBER. Enter the bus number on the FP where the device is attached. This is the second of a three-part key.
DEVNO		0 to 5	DEVICE INSTANCE NUMBER. Enter the instance number of the device type. This identifier distinguishes one device of the same type from another on the same FP. This is the third of a three-part key.
DEVTYPE		DK or CT	DEVICE DRIVE TYPE. Enter the type of device: DK for disk drive or CT for tape drive.
DEVPEC		NT9X90AB NT9X90BA	DEVICE DRIVE PEC. Enter the PEC for the device.
SCSIID		0 to 5	SCSI IDENTIFIER. Enter the SCSI bus identifier for the device. Devices can have specific priorities beginning with 0 (zero).
QUADNO		0 to 3	QUADRANT NUMBER. Enter the number of the quadrant containing the device.
SHELF		0 to 3	DEVICE SHELF. Enter the number of the shelf containing the device. The shelves are numbered from top to bottom, beginning with 0 (zero).
SHELFPEC		NT9X83AA	SHELF PEC. Enter the PEC for the device shelf.
FLOOR		0 to 99	FLOOR POSITION. Enter the floor number where the frame containing the device is located.

—continued—

FPDEVINV (end)

Field	Subfield or refinement	Entry	Explanation and action
ROW		A to H, J to N, P to Z, AA to HH, JJ to NN, or PP to ZZ	ROW POSITION. Enter the row position of the frame containing the device.
FRAME		0 to 99	FRAME POSITION. Enter the frame position in the row containing the device.

—end—

Datafill example

The following example shows sample datafill for table FPDEVINV.

```
FPNO SCSIBUS  DEVNO
DEVTYPE     DEVPEC   SCSIID   QUADNO   SHELF     SHELFPEC
FLOOR   ROW   FRAME
_____________________________________________________________________

   0       1      5
     DK   NT9X90AA        3        0       0     NT9X83AA
    0     R       1
```

FPDIPINV

Table name

File Processor Device Interface Paddle Board Inventory (FPDIPINV)

> **ATTENTION**
> The material found in this module regarding table FPDIPINV is UCS DMS-250 Billing Server-specific information. Refer to the *DMS-100 Customer Data Schema Reference Manual* for further details regarding datafill for table FPDIPINV.

Functional description

Table FPDIPINV contains the description of all the SDIP on a DMS SuperNode FP.

The description identifies the FPs, the SCSI bus to which the SDIPs are attached, and the SDIP identities, locations, and hardware configurations. Table FPDIPINV is used to dynamically add and delete SDIPs, and interacts directly with tables APINV, APCDINV, and FPDEVINV.

Datafill sequence and implications

The following tables must be datafilled before table FPDIPINV.

- PMLOADS
- APINV
- APCDINV

Consider the following when adding, deleting, or changing SDIP entries in table FPDIPINV:

- Each new entry to table FPDIPINV is verified by the FP file maintenance software before a tuple is added to the table. The verification confirms that
  - — The processor portion of the FP is datafilled in table APINV.
  - — The shelf and quadrant is not equipped with an SDIP.
  - — The SDIP number of the specified SCSI bus is not a duplicate.
  - — The SDIPs are in adjacent slots in one quadrant.
  - — The SDIPs are in specific slots in the quadrant.

— The slot numbers are different for both SDIPs.

— The controller numbers are different for both SDIPs.

— The SCSI Interface Paddleboard (SIP) associated with the SDIP's SCSI bus is not datafilled in APCDINV.

Note: The SIPs inserted in slots 11R, 16R, 23R, and 28R correspond to SCSI bus 0. These slots are closest to the CPU card and are datafilled by default when an FP is datafilled in table APINV. The optional SIPs in slots 10R, 17R, 22R, and 29R correspond to SCSI bus 1. The R refers to the rear slot positions.

- A tuple not meeting the previous rules can still be added resulting in the display of warning messages for each datafill error. A tuple containing errors can be accepted but missing inventory information makes the FP device inoperable.
- The only entry that can be changed is the SDIP PEC.
- Before an SDIP can be deleted from the datafill, the attached device, if present, must be deleted from the datafill. If the maintenance software is not able to obtain device status the deletion request is rejected.

Table sizing

Memory for table FPDIPINV is dynamically allocated. The maximum size of table FPDIPINV is 1600 tuples. The maximum table size is set by calculating the number of possible tuples according to the following formula:

$$A = b \times c \times d$$

where

A is the number of possible tuples
b is the maximum number of file processors
c is the maximum number of SCSI buses
d is the maximum number of SDIPs

Field descriptions

The following table describes field names, subfield names, and valid data ranges for table FPDIPINV.

Field	Subfield or refinement	Entry	Explanation and action
FPNO		0 to 99	FP INSTANCE NUMBER. Enter the instance number of the FP. This is the first of a three-part key.
SCSIBUS		0 to 1	SCSI BUS NUMBER. Enter the bus number on the FP where the SDIPs are attached. This is the second of a three-part key.
SDIPNO		0 to 5	SDIP NUMBER. Enter the particular SDIP pairs on the SCSI bus. This is the third of a three-part key.
SHELF		0 to 3	SDIP SHELF. Enter the number of the shelf containing the SDIPs. The shelves are numbered from top to bottom, beginning with zero (0).
SHELFPEC		NT9X83AA	SHELF PEC. Enter the PEC for the SDIP shelf.
FLOOR		0 to 99	FLOOR POSITION. Enter the floor number where the frame containing the SDIPs is located.
ROW		A to H, J to N, P to Z, AA to HH, JJ to NN, or PP to ZZ	ROW POSITION. Enter the row position of the frame containing the SDIPs.
FRAME		0 to 99	FRAME POSITION. Enter the frame position in the row containing the SDIPs.
DIPIDS		see subfields	SDIP IDENTIFICATION. This field contains a table indexed by a controller (CTRL) number that distinguishes one SDIP from the other. This field is a vector and consists of subfields CTRLNO, SLOTNO, and DIPPEC.

—continued—

FPDIPINV (end)

Field	Subfield or refinement	Entry	Explanation and action
	CTRLNO	0 or 1	CONTROLLER NUMBER. Enter the controller number of the SCSI bus to which the SDIP is attached.
	SLOTNO	1 to 38	SLOT NUMBER. Enter the number of the slot where the SDIP is located. SDIPs must be in adjacent slot numbers.
	DIPPEC	NT9X89AA NT9X89BA	SDIP PEC. Enter the PEC for the SDIP.

—end—

Datafill example

The following example shows sample datafill for table FPDIPINV.

```
FPNO SCSIBUS  SDIPNO
SHELF      SHELFPEC    FLOOR    ROW    FRAME                DIPIDS
_____________________________________________________________________

   0       1       2
    0     NT9X83AA       0     R       1   (0 8 NT9X89AA) (1 9
NT9X89AA) $
```

LIUINV

Table name

Link Interface Unit Inventory (LIUINV)

> **ATTENTION**
> The material found in this module regarding table LIUINV is UCS DMS-250 Billing Server-specific information. Refer to the *DMS-100 Customer Data Schema Reference Manual* for further details regarding datafill for table LIUINV.

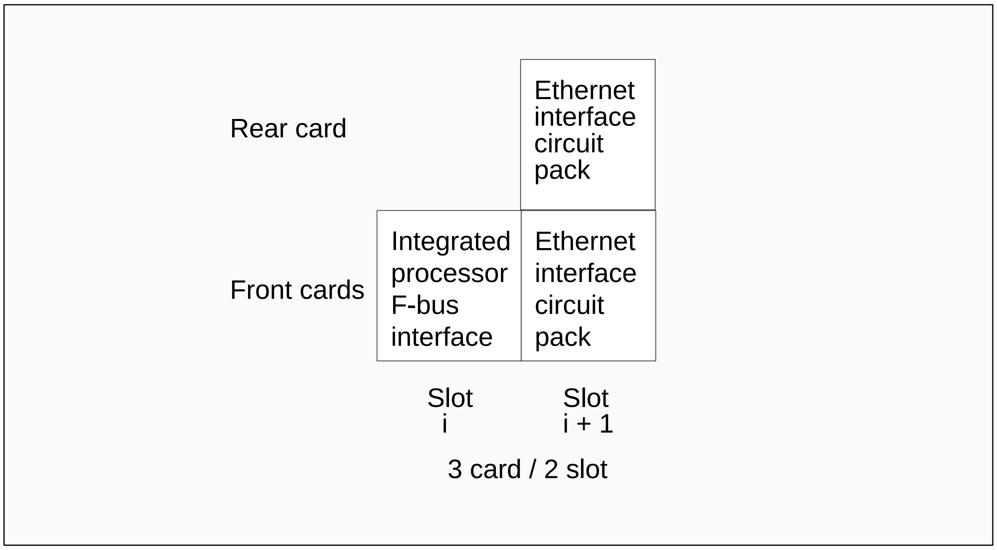
Functional description

Table LIUINV defines the EIUs for the SuperNode Billing Server.

EIUs are provisioned on a FLIS also known as an SSLPP.

The maximum number of LIUs or EIUs datafilled in a single office is 636.

The EIU consists of three cards, as shown below.



Datafill sequence and implications

The following tables must be datafilled before table LIUINV:

- PMLOADS
- SUSHELF

Table sizing

0 to 750 tuples

Field descriptions

The following table describes field names, subfield names, and valid data ranges for table LIUINV.

Field	Subfield or refinement	Entry	Explanation and action
LIUNAME		see subfields	LINK INTERFACE UNIT NAME. This is the key field, which consists of subfields LIUTYPE and LIUNO. This field uniquely identifies the EIU present on the FLIS.
	LIUTYPE	EIU	LINK INTERFACE UNIT TYPE. Enter EIU for Billing Server.
	LIUNO	0 to 511	LINK INTERFACE UNIT NUMBER. Enter the number assigned to the EIU.
LOCATION		see subfields	LOCATION. This field contains the location of the EIU and consists of subfields CTRL, SHELFNUM, and LIUSLOT.
	CTRL	see subfield	CONTROL INFORMATION. This field consists of subfield CONTROL.
	CONTROL	MS	CONTROLLING HOST ENTITY. Enter MS if the controlling host is a message switch and datafill subfields MSCARD and MSPORT.
	MSCARD	6 to 23	MESSAGE SWITCH CARD. Enter the message switch card number
	MSPORT	0 to 3	MESSAGE SWITCH PORT. Enter the message switch port number.
	SHELFNUM	0 to 3	SHELF NUMBER. Enter the shelf number, at the host APC, on which the EIU is located.
	LIUSLOT	8 to 31	LINK INTERFACE SLOT. Enter the physical slot number, at the host LIM, on which the EIU resides. The EIU occupies two slots. The leftmost card is chosen to represent the logical location of the card.
LOAD		up to 8 alphanumeric characters	SOFTWARE LOAD NAME. Enter the applicable software load name found in table PMLOADS.

—continued—

Field	Subfield or refinement	Entry	Explanation and action
PROCINFO		see subfield	PROCESSOR INFORMATION. This field specifies the PEC of the processors used in the EIU and consists of subfield PROCPEC.
	PROCPEC	NTEX22BA NTEX22BB	PROCESSOR PEC. Enter NTEX22BA or NTEX22BB as the PECs for the 8-Mbyte integrated processor and F-bus interface cards. The difference between the NTEX22BA and NTEX22BB cards is in firmware only, the hardware is identical.
CARDINFO		see subfields	CARD INFORMATION. This field specifies the card data and consists of subfields APPLPEC.
	APPLPEC	NT9X84AA	APPLICATION PEC. Card NT9X84AA is used with EIU applications.
		—end—	

APPLPEC = NT9X84AA

If the entry for field APPLPEC is NT9X84AA, datafill the following subfields:

Field	Subfield or refinement	Entry	Explanation and action
	PBPEC	NT9X85AA NT9X85BA	PADDLE BOARD PEC. Enter one of the PECs. • Enter NT9X85AA for EIU coax applications. • Enter NT9X85BA for EIU unshielded twisted pair applications.
	HEARTBEAT	YES or NO	HEARTBEAT. Enter YES if the EIU expects a heartbeat indication signal from the media access unit (MAU) connected to it; otherwise, enter NO.
	MAC_ ADDRESS	0075F00000 00 to 0075FFFFFF FF	MAC ADDRESS. Enter a 12-character hex string representing the Media Access Control (MAC) address.

LIUINV (end)

The following example shows sample datafill for table LIUINV.

```
 LIUNAME       LOCATION      LOAD          PROCINFO
                                 CARDINFO
______________________________________________________________________

EIU  102    MS 12 0 0 12   OSU02AO         NTEX22BB
   NT9X84AA  NT9X85AA  YES  000075F00101
```

MSCDINV

Table name

Message Switch Cards Inventory (MSCDINV)

> **ATTENTION**
> The material found in this module regarding table MSCDINV is UCS DMS-250 Billing Server-specific information. Refer to the *DMS-100 Customer Data Schema Reference Manual* for further details regarding datafill for table MSCDINV.

Functional description

Table MSCDINV is datafilled when the fiber links between the MS and the FP are established.

Table MSCDINV is a look-up table containing a description of the MS circuit packs and consists of the following types of circuit packs:

- system cards
- bus extension units
- interface cards

System cards

The system cards are vital to the operation of the owner MS node. They are automatically filled with default data when the MS node is added to table MSINV, and are deleted from table MSCDINV if the MS node is deleted from table MSINV.

Note 1: The slot positions of the system cards cannot be altered.

Note 2: Only the PEC-related fields may be altered in table MSCDINV.

Bus extension units

The bus extension units are vital to the operation of the owner extension shelf.

Interface cards

The interface cards are arranged in different configurations to provide for their respective functions. Each of the DS30 and non-channelized DS512 cards consists of one interface unit. The channelized DS512 card can have multiple interface units. These units are linked together serially to form a daisy chain. Each unit, in turn, consists of a main card, front card and its subordinate card, and back card. A chain can also have up to eight cards linked together. For non-interface units, the front and back cards can have unrelated functions.

The mechanism for modifying chain tuples has been enhanced to handle the updating of PECs for existing chains. The following rules apply when modifying chain tuples:

- A PEC can be changed to a compatible PEC for any individual card in a chain without affecting any other cards. A compatible PEC is one that can coexist on a chain with the existing PECs.
- Changing the head card to an incompatible PEC causes the change to be propagated to all the relevant cards in the chain.
- A PEC can be changed to an incompatible PEC at the head card only.

To datafill the different configurations, a card selector is provided in each tuple. Single system cards are considered to be cards with functionally linked front and back cards. The clock is an example of such a card. Double system cards have front and back cards that are functionally different.

Single interface cards are treated as independent cards. Multiple interface cards are datafilled sequentially, in the order they appear on the shelf.

The following circuit packs are required for Billing Server and should be included in the datafill for table MSCDINV:

MS hardware	Front PEC	Physical slot	Back PEC
MS 4 port card cp	NT9X17AD	18	NT9X62BA
MS 4 port card cp	NT9X17AD	24	NT9X20BB
MS 4 port card cp	NT9X17AD	25	NT9X20BB
DMS-Bus 64 port cp (circuit pack)	NT9X17DA	30	NT9X62BA
DMS-Bus 64 port cp (circuit pack)	NT9X17DA	31	NT9X62BA

Datafill sequence and implications

Table MSINV must be datafilled before table MSCDINV.

Table sizing

The MSCDINV table size is 20 to 52 tuples.

There are 20 default tuples in table MSCDINV: 10 for each default MS. There are 26 cards on each shelf, and all of these can be datafilled.

The following tables show the range of front and back card PECs.

Front PEC card type	Range of values
CHAIN	NT9X17AD, NT9X17DA
DS512	NT9X17AD, NT9X17DA

Back PEC card type	Range of values
CHAIN	NT9X20BB, NT9X62BA
DS512	NT9X20AA, NT9X20BB, NT9X62BA

Field descriptions

The following table describes field names, subfield names, and valid data ranges for table MSCDINV.

Field	Subfield or refinement	Entry	Explanation and action
MSCDKEY		see subfields	MS CARD KEY. This field consists of subfields MSNUM, SHELFNUM, and CARDNUM.
	MSNUM	0 to 1	MS NUMBER. Enter the MS number.
	SHELFNUM	0 to 3	SHELF NUMBER. Enter the shelf number of the card slot.
	CARDNUM	1 to 26	CARD POSITION NUMBER. Enter the logical card position number (1 for first, 2 for second, and so on) relative to the other 26 cards. The MS has a total of 38 physical slots.
SLOTINFO		see subfield	SLOT INFORMATION. This field consists of subfield CARDTYPE.
	CARDTYPE	CHAIN DS512	CARD TYPE. Enter the type of card. If the entry in this field is CHAIN, datafill fields PROTOCOL and HEADPECS.
	PROTOCOL	DS512	PROTOCOL. Enter the protocol of the chain.

—continued—

Field	Subfield or refinement	Entry	Explanation and action
	HEADPECS	see subfields	HEAD PEC. This field consists of subfields FRONTPEC and BACKPEC.
	FRONTPEC	alphanumeric (8 characters)	FRONT PEC. Enter the PEC for the front card.
	BACKPEC	alphanumeric (8 characters)	BACK PEC. Enter the PEC for the back card. If the entry in subfield CARDTYPE is CHAIN or DS512, and the entry in this subfield is NT9X62BA, datafill refinement NUMLINKS.
	NUMLINKS	1, 2, or 4	NUMBER OF LINKS. If the entry in subfield CARDTYPE is CHAIN or DS512, and the entry in field BACKPEC is NT9X62BA, enter the number of links on the NT9X62 card in the chain being datafilled as follows: • If the entry in subfield BACKPEC is NT9X62BA, enter 1, 2, or 4.
	BODYPECS	see subfields	BODY PEC. This field consists of subfields BODYFRONTPEC and BODYBACKPEC. Enter up to 7 groups of front PEC and allowable back PEC that form the body of the chain.
	BODYFRONT PEC	alphanumeric (8 characters)	BODY FRONT PEC. Enter the PEC of the front card connected to the body section of the chain.
	BODYBACK PEC	NT9X25AA or NT9X25BA	BODY BACK PEC. Enter the PEC of the back card connected to the body section of the chain. The last card in a chain must be NT9X25BA.

—end—

Datafill example

The following example shows sample datafill for table MSCDINV.

MSCDKEY	SLOTINFO
0 0 1	TBUSACC NT9X52AA NIL
0 0 2	CLOCK NT9X53AA NT9X54AB
0 0 3	MSP NT9X13DB NT9X26AA
0 0 4	MEMORY NT9X14DB
0 0 5	MAPPER NT9X15AA
0 0 20	DS30 NT9X17AA NT9X23BA
0 0 21	DS30 NT9X17AA NT9X23BA
0 0 22	DS30 NT9X17AA NT9X23AA
0 0 23	DS30 NT9X17AA NT9X23AA
0 0 24	CMIC NT9X17AA NT9X20AA
0 0 25	CMIC NT9X17AA NT9X20AA
0 0 26	PBUS NT9X49CB NIL
1 0 1	TBUSACC NT9X52AA NIL
1 0 2	CLOCK NT9X53AA NT9X54AB
1 0 3	MSP NT9X13DB NT9X26AA
1 0 4	MEMORY NT9X14DB
1 0 5	MAPPER NT9X15AA
1 0 20	DS30 NT9X17AA NT9X23BA
1 0 21	DS30 NT9X17AA NT9X23BA
1 0 22	DS30 NT9X17AA NT9X23AA
1 0 23	DS30 NT9X17AA NT9X23AA
1 0 24	CMIC NT9X17AA NT9X20AA
1 0 25	CMIC NT9X17AA NT9X20AA
1 0 26	PBUS NT9X49CB

MSFWLOAD

Table name

Message Switch Firmware Load (MSFWLOAD)

> **ATTENTION**
> The material found in this module regarding table MSFWLOAD is UCS DMS-250 Billing Server-specific information. Refer to the *DMS-100 Customer Data Schema Reference Manual* for further details regarding datafill for table MSFWLOAD.

Functional description

Table MSFWLOAD contains the name of the firmware load file for each of the firmware downloadable MS PEC. It also contains the name of the device where the file resides.

The switch uses the file name and device associated with the PEC of the card downloaded as the default file and device parameters for the command LOADCD.

Only the NT9X17CA (DMS-bus 128-port card), NT9X17DA (message switch 64-port card), and NT9X17BB (DMS-bus 32-port card) interface cards are firmware downloadable.

Datafill sequence and implications

There is no requirement to datafill other tables prior to table MSFWLOAD.

Table sizing

The MSFWLOAD table size is 1 to 32 tuples.

Field descriptions

The following table describes field names, subfield names, and valid data ranges for table MSFWLOAD.

Field	Subfield or refinement	Entry	Explanation and action
PECODE		see subfield	INTERFACE PEC. This field consists of subfield NAME.
	NAME	NT9X17DA	NAME. Enter the the PEC of the interface card.

MSFWLOAD (end)

Field	Subfield or refinement	Entry	Explanation and action
DEV		alphanumeric (1 to 16 characters)	DEVICE NAME. Enter the name of the device where the firmware load file resides.
FILENAME		alphanumeric (1 to 8 characters)	FIRMWARE LOAD FILE NAME. Enter the file name of the firmware load file.

Datafill example

The following example shows sample datafill for table MSFWLOAD.

```
    PECODE                DEV  FILENAME
________________________________________________________________________

 NT9X17CA       D000PMLOADS   MPF36CJ
 NT9X17DA       D000PMLOADS   MPF36CJ
 NT9X17BB       D000PMLOADS   MPF36CJ
```

MSILINV

Table name

Message Switch Inter-MS Link Inventory (MSILINV)

> **ATTENTION**
> The material found in this module regarding table MSILINV is UCS DMS-250 Billing Server-specific information. Refer to the *DMS-100 Customer Data Schema Reference Manual* for further details regarding datafill for table MSILINV.

Functional description

Table MSILINV defines inter-message-switch links that provide greater messaging reliability in the DMS-bus.

The interlinks are used by the frame transport system (FTS) messaging software to direct messages across DMS MS at the DMS-bus protocol level.

The interlinks must be used for the DMS-250 UCS Billing Server.

System reliability is improved by providing an additional path for messaging when one or more node message ports are not functional. The FTS software can detect nonfunctioning message points and reroute messages using the MS interlinks. Previously, if a path was not available, the message was lost.

For related information, refer to table MSCDINV.

Datafill sequence and implications

The following tables must be datafilled before table MSILINV:

- MSINV
- MSCDINV

Table sizing

The MSILINV table size is 0 to 2 tuples.

Field descriptions

The following table describes field names, subfield names, and valid data ranges for table MSILINV.

Field	Subfield or refinement	Entry	Explanation and action
IMSL		see subfield	INTER-MESSAGE SWITCH LINK. The field consists of subfield LINKNO.
	LINKNO	0 to 1	MS INTERLINK NUMBER. Enter the link number of the interlink to be defined. ***Note:*** Link 0 (zero) must be added before link 1 can be added.
SHELFNO		0 to 3	MS SHELF NUMBER. Enter the number of the MS shelf on which the interlink resides. For most DMS SuperNodes, only shelf 0 exists. ***Note:*** All MS shelves have MS 0 and MS 1 planes.
CARDNO		6 to 23	MS CARD NUMBER. Enter the number of the MS card on which the interlink resides. For example, if card 5 is chosen, card 5 of MS 0 and card 5 of MS 1 are linked together.
INFO		see subfields	MS INTERLINK INFORMATION. This field consists of subfields PROTOCOL and PORTNO.
	PROTOCOL	DS512	MESSAGING PROTOCOL. Enter DS512 and datafill subfield PORTNO.
	PORTNO	0	MS PORT NUMBER. Enter 0.

Datafill example

An example of datafill for table MSILINV is shown below. The example defines two interlinks.

Link 0 connects card 6, port 12 on MS shelf 0, MS 0, to card 6, port 12 on MS shelf 0, MS 1 using DS512 protocol.

MSILINV (end)

Link 1 connects card 19, port 0 on MS shelf 0, MS 0, to card 19, port 0, on MS shelf 0, MS 1 using DS512 protocol.

```
IMSL SHELFNO CARDNO     INFO
________________________________________________________________

   0       0    18      DS512 0
   0       0    19      DS512 0
```

Table name

Open Systems Interconnect Parameter (OSIPARMS)

> **ATTENTION**
> The material found in this module regarding table OSIPARMS is UCS DMS-250 Billing Server-specific information. Refer to the *DMS-100 Customer Data Schema Reference Manual* for further details regarding datafill for table OSIPARMS.

Functional description

Table OSIPARMS modifies the parameters within the various layers of the billing server International Standards Organization (ISO) implementation.

Table OSIPARMS allows the user to alter parameters for the following ISO protocol specifications:

- ISO 8073AD2 (Class 4 Service Over Connectionless Network)
- ISO 8073 (Connection Oriented Transport Protocol)
- ISO 8327 (Connection Oriented Session Protocol)
- ISO 8823 (Connection Oriented Presentation Protocol)
- ISO 8649 (Association Control Service Element Protocol)
- ISO 8571 (File Transfer Access Management Protocol)

Each tuple in table OSIPARMS corresponds to one OSI protocol stack running on a specific EIU.

The local area network controller for Ethernet (LANCE) chip is set to transmit the most significant byte first (byte flipping enabled). This results in real time savings (1 ms for each frame) for the Open Systems Interconnect unit (OSIU) stack. This mode for the LANCE chip is activated only if the OSIU is in service.

Datafill sequence and implications

The following tables must be datafilled before table OSIPARMS.

- LIUINV
- OSIROUTE

Table sizing

0 to 16 tuples

Field descriptions

The following table describes field names, subfield names, and valid data ranges for table OSIPARMS.

Field	Subfield or refinement	Entry	Explanation and action
OSIPARMS-KEY		see subfield	OSI PARAMETERS KEY, This field consists of subfield TUPLE_KEY.
	TUPLE_KEY	up to 8 alphanumeric characters	TUPLE KEY. This is the key field of the table. Enter the tuple key.
OSI_ HOSTNAME		see subfields	OSI HOST NAME. This field consists of subfields WORD_NODE and NODE_RNG.
	WORD_NODE	EIU	WORD NODE. Enter EIU for the Ethernet interface unit.
	NODE_RNG	0 to 750	NODE RANGE. Enter a node range value.
TRANSPORT			TRANSPORT LAYERS. Enter up to eight transport layers. If less than eight entries are required, end the list with a dollar sign ($).
		INACTMR	Enter INACTMR (inactivity timer) and datafill refinement INACTMR_VALUE.
	INACTMR_ VALUE	2 to 2	INACTIVITY TIMER VALUE. Enter the value that defines the maximum length of time, in minutes, that the transport layer waits for a transport time-out signal. The time-out signal indicates that the transport layer communication handshaking is inactive. The default value is 2.
		L4CONNS	Enter L4CONNS (number of connections) and datafill refinement L4NUMCON_VALUE.
	L4NUMCON_ VALUE	48 to 48	NUMBER OF CONNECTIONS. Enter the value that defines the maximum number of simultaneous transport layer connections. The default value is 48.
		LACKTMR	Enter LACKTMR (local acknowledgement delay timer) and datafill refinement LACKTMR_VALUE.

—continued—

Field	Subfield or refinement	Entry	Explanation and action
	LACKTMR_ VALUE	0 to 0	LOCAL ACKNOWLEDGMENT DELAY TIMER VALUE. Enter the value that defines the time to delay acknowledgement of received TPDUs. The default value is 0 (zero), which means the timer is disabled.
		RETRIES	Enter RETRIES (retries) and datafill refinement RETRIES_VALUE.
	RETRIES_ VALUE	8 to 8	RETIRES VALUE. Enter the value to define the maximum number of times the transport layer retransmits a single unacknowledged TPDU before issuing a disconnect. The default value is 8.
		TPCLASS	Enter TPCLASS (transport layer class) and datafill refinement CLASS_VALUE.
	CLASS_ VALUE	0 to 4	TRANSPORT LAYER CLASS VALUE. Enter the value to specify the transport layer classes of services supported. • Class 0 provides the most simple type of transport connection. • Class 1 provides a basic transport connection with minimum overhead. • Class 2 provides multiplexing of several transport connections into a single network connection. • Class 3 provides the capabilities of class 2 plus the ability to recover from network disconnect or reset. • Class 4 provides the capabilities of class 3 plus the ability to detect and recover from errors as a result of lower layer failure. The kinds of errors detected by class 4 include TPDU loss, TPDU delivery out of sequence, TPDU duplication, and TPDU corruption.
		TPDUSIZE	Enter TPDUSIZE (transport layer protocol disk units [TPDU]) and datafill refinement TPDUSIZE_VALUE.

—continued—

Field	Subfield or refinement	Entry	Explanation and action
	TPDUSIZE_ VALUE	T1024	TRANSPORT LAYER PROTOCOL DISK UNIT SIZE VALUE. Enter the value to define the maximum size of the TPDUs. The default value is T1024.
		WINSIZE	Enter WINSIZE (window size) and datafill refinement WINDSIZE_VALUE.
	WINDSIZE_ VALUE	8 to 8	WINDOW SIZE VALUE. Enter the value to define the maximum number of outstanding TPDUs awaiting acknowledgement. The default value is 8.
		WINTMR	Enter WINTMR (window timer) and datafill refinement WINTMR_VALUE.
	WINTMR_ VALUE	15 to 15	WINDOW TIMER VALUE. Enter the value to define the frequency, in seconds, with which the transport layer sends an acknowledgement TPDU, if there is no activity, to determine if its peer entity is still active. The default value is 15.
SESSION			SESSION LAYER. Enter up to seven session layers. If less than seven layers are required, end the list with a $ (dollar sign).
		EXPEDTRP	Enter EXPEDTRP (expedited transport) and datafill datafill refinement EXPEDTRP_VALUE.
	EXPEDTRP_ VALUE	NO	EXPEDITED TRANSPORT VALUE. Enter the value to specify whether the transport layer expedited flow is available to the session connection. The default value is NO.
		EXTCONCAT	Enter EXCONCAT (extended concatenation) and datafill refinement EXCONCAT_VALUE.
	EXCONCAT_ VALUE	YES	EXTENDED CONCATENATION VALUE. Enter the value to specify whether extended concatenation is supported. The default value is YES.
		L5CONNS	Enter L5CONNS (number of connection) and datafill refinement L5NUMCON_VALUE.

—continued—

Field	Subfield or refinement	Entry	Explanation and action
	L5NUMCON_ VALUE	48 to 48	NUMBER OF CONNECTIONS. Enter the value to define the maximum number of simultaneous session layer connections. The default value is 48.
		PROTVER	Enter PROTVER (protocol version) and datafill refinement PROTVER_VALUE.
	PROTVER_ VALUE	VER_1 VER_2 or VER_1_2	PROTOCOL VERSION VALUE. Enter the value to specify the session layer protocol version. The default value is VER_1_2.
		REUSETRP	Enter REUSETRP (reuse existing transport connections) and datafill refinement REUSETRP_VALUE below.
	REUSETRP_ VALUE	YES	REUSE EXISTING TRANSPORT CONNECTIONS VALUE. Enter the value to enable or disable the attempt to reuse an existing transport connection once it has been released by a closed session connection. The default value is YES.
		SEGMENT	Enter SEGMENT (segmentation) and datafill refinement SEGMENT_VALUE.
	SEGMENT_ VALUE	NO	SEGMENTATION VALUE. Enter the value to specify whether segmentation is allowed. The default value is NO.
		TSDUSIZE	Enter TSDUSIZE (TSDU size) and datafill refinement TSDUSIZE_VALUE.
	TSDUSIZE_ VALUE	S4096	TRANSPORT SERVICE DATA UNIT MASIMUM SIZE. Enter the value to define the maximum size of the transport service data unit. The default value is S4096.
PRESENT- ATION		L6CONNS	PRESENTATION LAYER. Enter the presentation layer, L6CONNS (number of connections) and datafill refinement L6NUMCON_VALUE.
	L6NUMCON_ VALUE	48 to 48	NUMBER OF CONNECTIONS. Enter a value to define the maximum number of simultaneous presentation layer connections. The default value is 48.

—continued—

Field	Subfield or refinement	Entry	Explanation and action
APPLICATION			APPLICATION LAYER. Enter up to five application layers. If less than five layers are required, end the list with a $ (dollar sign).
		ASCECONN	Enter ASCECONNS (association control service element [ASCE] connections) and datafill refinement ASCECONN_VALUE.
	ASCECONN_ VALUE	48 to 48	ASSOCIATION CONTROL SERVICE ELEMENT CONNECTIONS VALUE. Enter the value to define the maximum number of ASCE connections. The default value is 48.
		FTAMASSOC	Enter FTAMASSOC (file transfer access management [FTAM] associations) and datafill refinement FTAMASSO_VALUE.
	FTAMASSO_ VALUE	1 to 4	FILE TRANSFER ACCESS MANAGEMENT ASSOCIATION VALUE. Enter the value to define the maximum number of connections an FTAM application can have established at any one time. The default value is 4.
		FTAMBKSZ	Enter FTAMBLKSZ (FTAM block size) and datafill refinement FTAMBLK_VALUE.
	FTAMBLK_ VALUE	A1024 A2048 A4096	FILE TRANSFER ACCESS MANAGEMENT BLOCK. Enter a value to define the size of FTAM blocks in bytes. The default value is A2048.
		FTAMCKWN	Enter FTAMCKWN (FTAM checkpoint window) and datafill refinement FTAMCKWN_VALUE.
	FTAMCKWN_ VALUE	1 to 8	FILE TRANSFER ACCESS MANAGEMENT CHECKPOINT WINDOW. Enter a value to define the maximum number of checkpoints that remain unacknowledged. The default value is 8.

—continued—

Field	Subfield or refinement	Entry	Explanation and action
		FTAMQOS	Enter FTAMQOS (FTAM quality of service) and datafill refinement FTAMQOS_VALUE.
	FTAMQOS_ VALUE	0 to 3	FILE TRANSFER ACCESS MANAGEMENT QUALITY OF SERVICE VALUE. Enter the value to define the quality of service provided by the FTAM application. The default value is 0 (zero). • Enter 0 (zero) for no susceptibility to errors. • Enter 1 for susceptibility to errors that can damage the data transfer regime. • Enter 2 for susceptibility to errors that can damage the open or data transfer regime. • Enter 3 for susceptibility to errors that can damage the select, open, or data transfer regime.
—end—			

OSIPARMS (end)

Datafill example

The following example shows sample datafill for table OSIPARMS.

The example consists of one tuple representing the entire OSI stack for that specific EIU.

```
OSIPARMSKEY OSI_HOSTNAME
                                                        TRANSPORT
                                                          SESSION
                                                     PRESENTATION
                                                      APPLICATION
_____________________________________________________________________
       BNR1      EIU 208
(INACTMR 2) (LACKTMR 0) (L4CONNS 48) (RETRIES 8) (TPCLASS 4)
(TPDUSIZE T1024) (WINSIZE 8) (WINTMR 15) $
(EXPEDTRP NO) (EXTCONCAT YES) (L5CONNS 48) (PROTVER VER_1_2)
(REUSETRP YES)
(SEGMENT NO) (TSDUSIZE S4096) $
(L6CONNS 48)$
(ASCECONNS 48) (FTAMASSOC 4) ( FTAMBKSZ A2048) ( FTAMCKWN 8)
(FTAMQOS 0) $
```

Table name

Open Systems Interconnect Routing (OSIROUTE)

> **ATTENTION**
> The material found in this module regarding table OSIROUTE is UCS DMS-250 Billing Server-specific information. Refer to the *DMS-100 Customer Data Schema Reference Manual* for further details regarding datafill for table OSIROUTE.

Functional description

Table OSIROUTE describes the OSI static routing information for the billing server network layer. The routing information supports the ISO 8473 standard, connectionless Internet Protocol (IP). Table OSIROUTE functions as a look-up table for static route information to map the network entity title (NET) portion of the network services access point (NSAP) into the subnetwork point of attachment (SNPA) in the billing server network layer. The key field is OSIKEY. Table LIUINV lists the inventory of processor types and their network addresses.

Datafill sequence and implications

Table LIUINV must be datafilled after table OSIROUTE.

Table sizing

The OSIROUTE table size is 2 to 32 tuples.

A maximum of 32 network addresses can be entered. The table size is set by the datafill in table SYSDATA. Each tuple requires 132 bytes of physical store. The storage requirement is therefore 4224 bytes (32 tuples × 132 bytes/tuple = 4224 bytes).

Field descriptions

The following table describes field names, subfield names, and valid data ranges for table OSIROUTE.

Field	Subfield or refinement	Entry	Explanation and action
OSIKEY		alphanumeric	OSI KEY. Enter up to a maximum of eight characters for the network address. Entries must begin with a letter, not the number. This is the key field to table OSIROUTE.
AFI		0 to 9 and A to F	AUTHORITY AND FORMAT IDENTIFIER. Enter the two-character hexadecimal byte value for the authority and format identifier used for the network address.
DSP		0 to 9 and A to F	DOMAIN SPECIFIC PART. Enter the nine two-character hexadecimal byte values for the domain-specific part of the network address. The first two bytes are the area address identifying the subnet within the network. The following six bytes are the logical node identifier identifying the node in the specific subnet The last byte is the NSAP selector identifying the user of the network service.

—continued—

Field	Subfield or refinement	Entry	Explanation and action
SYSSEL		see subfields	SYSTEM SELECTOR. This field consists of subfield TYPE and its refinements HOSTNAME, REQ REM SYS, WORDNODE, NODERNG, PREFNET, REACHNET, and MACADDR.
	TYPE	ESWIOSIU ESWOOSIU ISWIOSIU REM_ES_ SYS REM_IS_ SYS	SYSTEM TYPE. Enter the selector for the type of system. The following selectors apply to the open systems interconnect unit (OSIU) on the SuperNode/DMS billing server: • Enter ESWIOSIU for an end system (ES) running on the OSIU. • Enter ESWOOSIU for an ES not running on the OSIU. • Enter ISWIOSIU for an intermediate system (IS) running on the OSIU. The following selectors apply to other ESs that are not part of the DMS, but are part of the LAN connected to the DMS: • Enter REM_ES_SYS for an ES. • Enter REM_IS_SYS for an IS. The default value is ESWIOSIU.
—end—			

TYPE = ESWIOSIU, ESWOOSIU, or ISWIOSIU

If the entry in subfield TYPE is ESWIOSIU, ESWOOSIU, or ISWIOSIU, datafill refinements HOSTNAME, WORDNODE, and NODERNG.

Field	Subfield or refinement	Entry	Explanation and action
	HOSTNAME	see subfields	HOSTNAME. This field consists of subfields WORDNODE and NODERNG.
	WORDNODE	APUX, CM, EIU, LIU7, FRIU, XLIU, LCOM, VPU, or FP	WORD NODE NAME. Enter the name for the DMS processor type. The entry must match the entry in field LIUTYPE in table LIUINV. Entries outside this range are invalid. The default value is EIU.
	NODERNG	numeric	NODE RING IDENTIFIER. Enter the node ring identifier for the DMS processor type. The entry must match the entry in field LIUNO in table LIUINV. The following are valid entries for the corresponding values entered in WORDNODE: • APUX 1 to 98 • CM 0 only • EIU 1 to 710 • FP 1 to 99 Entries outside this range are invalid. The default value is 1.

TYPE = ESWOOSIU

If the entry in subfield TYPE is ESWOOSIU, datafill the additional refinement PREFNET to complete the datafill for field SYSSEL.

Field	Subfield or refinement	Entry	Explanation and action
	PREFNET	alphanumeric (vector of up to 8 characters)	PREFERRED NETWORK NAME. Enter a name vector to define the ES router within the DMS. The name must match one of the names listed in field OSIKEY.

TYPE = REM_ES_SYS or REM_IS_SYS

If the entry in subfield TYPE is REM_ES_SYS or REM_IS_SYS, datafill refinements REACHNET and MACADDR.

Field	Subfield or refinement	Entry	Explanation and action
	REACHNET	alphanumeric	REACHABLE NETWORK NAMES. Enter up to two name vectors of up to eight characters each to define paths to other systems in the LAN. Names must match values in field OSIKEY. A minimum of one name must be entered.
	MACADDR	0 to 9 and A to F	MAC ADDRESS. Enter up to two media access control (MAC) address vectors of up to 12 hexadecimal characters each to define the physical addresses of remote systems able to reach this network address. A minimum of one address must be entered.

Datafill example

The following example shows sample datafill for table OSIROUTE.

The example consists of two ESs with an OSIU, two ESs without an OSIU, and two non-DMS remote systems.

```
 OSIKEY AFI                 DSP
                                                SYSSEL
________________________________________________________
   NET1  49 0000000000000000001
                       ESWIOSIU  EIU    07
NETLAST  49 0000000000000000011
                       ISWIOSIU  EIU    08
   NET2  49 0000000000000000021
                       ESWOOSIU   FP    01   NET 1
   NET3  49 0000000000000000031
                       ESWOOSIU   CM     0 NETLAST
  CUST1  49 0000000000000000041
                       REM_ES_SYS NET1 $
                               00007     5F00022
  CUST2  49 0000000000000000051
                       REM_IS_SYS NET2 $
                               00007     5F00023
```

PMLOADS

Table name

Peripheral Module Loads (PMLOADS)

> **ATTENTION**
> The material found in this module regarding table PMLOADS is UCS DMS-250 Billing Server-specific information. Refer to the *DMS-100 Customer Data Schema Reference Manual* for further details regarding datafill for table PMLOADS.

Functional description

Table PMLOADS is used to store the device location of every PM load file. It also stores mapping between the load names and devices on which the loads reside. This permits autoload to locate load files without the intervention of operating company personnel.

> ***Note:*** Autoloading is not enabled for tape devices. PM loads must be stored on a disk device. Tuples entered during initial datafill are not verified. All other add operations verify that the device and files exist.

Office alarms

Use of the autoload option is only possible if the load files are stored on disk. A minor alarm is raised if a PM load is not located on the disk, because the ability of the magnetic tape center to recover PMs requiring reloads is lost if the load file is not located.

Datafill sequence and implications

The following tables must be datafilled after table PMLOADS.

- LTCINV
- RCCINV
- XESAINV
- MSBINV

Table sizing

The PMLOADS table size is 0 to 255 tuples.

Field descriptions

The following table describes field names, subfield names, and valid data ranges for table PMLOADS.

Field	Subfield or refinement	Entry	Explanation and action
LOADNAME		alphanumeric (up to 32 characters)	PM LOADFILE NAME. Enter the PM loadfile name. The loadfile name in this field must be the same as the loadfile name specified in the inventory tables.
ACTFILE		alphanumeric (up to 32 characters)	ACTIVE PM LOADFILE NAME. Enter the active PM loadfile name. ***Note:*** The active loadfile can be the original loadfile or a patched loadfile.
ACTVOL		alphanumeric (up to 16 characters)	ACTIVE LOADFILE STORAGE DEVICE. Specify the device on which the active loadfile is stored. The range is the set of DDU volumes and SLM disks that are available to the computing module (CM).
BKPFILE		alphanumeric (up to 32 characters)	BACKKUP PM LOADFILE NAME. Enter the backup loadfile name.
BKPVOL		alphanumeric (up to 16 characters)	BACKUP LOADFILE STORAGE DEVICE. Specify the device on which the backup loadfile is stored. The range is the set of DDU volumes and SLM disks that are available to the CM.
UPDACT		Y or N	AUTOMATIC LOADFILE NAME UPDATE. Enter Y (yes) to update field ACTFILE automatically with the patched loadfile name. Otherwise, enter N (no). The default value for this field is Y. ***Note:*** This field also controls whether the loadfile is eligible for loadfile patching.
—end—			

The modified table PMLOADS stores data for the names of the active loadfile name, backup loadfile name, file locations, and an update active loadfile to indicate whether the user wants to activate loadfile patching.

Datafill example

The following example shows sample datafill for table PMLOADS.

```
LOADNAME   ACTFILE   ACTVOL    BKPFILE   BKPVOL
  UPDACT
________________________________________________________

LTI02C     LTI02C    S01DVOL1  LTI02C    S01DVOL1
   N
RMM34C     RMM34C    S00DVOL2  RMM34C    S00DVOL2
   N
M7CQA01    M7CQA01   S00DVOL1  M7CQA01   S00DVOL1
   N
MPCX33AB   MPCX33AB  S01DVOL1  MPCX33AB  S01DVOL1
   N
```

SBSFMT

Table name

SuperNode Billing System Format (SBSFMT)

Functional description

Table SBSFMT creates and defines service data streams for raw billing data. This table works with table SBSMAP to map billing data to a stream. These tables support the multiple stream/multiple collector feature. The formatter/storage agent (FSA) and collector process allow multiple streams and multiple collectors. The collector process receives raw billing data from an application and sends it to an FSA process. The FSA process, or billing server, formats the raw billing data and transfers the formatted billing records to the distributed recording manager (DRM), which then writes the data to the disk.

Multiple streams allow applications to separate different call types into different streams. For example, there can be different streams for operator service records and normal call records.

The stream connection manager (SCM) supervises connections between collectors and FSAs. The SCM uses the following factors to determine which collector connects to a FSA:

- DRM priority assignments
- FSAs registered with the SCM
- FSAs that are active
- FSAs that have disk space available for streams

For this release, the FSA process resides on one file processor. An FSA can also reside on the computing module (CM) for emergency backup purposes, but application-specific software must support this. The collector process, or application processor, resides on the CM.

Datafill sequence and implications

Datafill the CRSFMT and CRSMAP tables prior to table SBSFMT.

Adding or deleting streams does not require restarts.

Table sizing

The SBSFMT table size is 16 tuples.

Field descriptions

The following table describes field names and valid data ranges for table SBSFMT.

> **CAUTION**
> **Possible loss of billing data**
> Do not delete streams before all data is written to disk.

Field	Entry	Explanation and action
KEY	1–4 characters	Represents a 1–4-character vector that creates the service data stream. This field is in the range sbs_service_datastream_fullrange. The effective range for application recording streams is 0 to 15. Element 16 is assigned as a NIL stream by the initialization software for SBSFMT and cannot be changed by the table editor.
FORMAT	1–8 characters	This field determines billing data format and defines the characteristics for the service data stream. The field is in the range sbs_stream_format_code. The default is 16.

Datafill example

The following example shows datafill for table SBSFMT.

```
KEY                                    FORMAT
________________________________________________________
OCC                                    CDR250FMT
```

SBSMAP

Table name

SuperNode Billing System MAP (SBSMAP)

Functional description

Table SBSMAP table defines service data types. This table works with the SBSFMT table to map service data types to service data streams. These tables support the multiple stream/multiple collector feature. The formatter/storage agent (FSA) and collector process allow multiple streams and multiple collectors. The collector process receives raw billing data from an application and sends it to an FSA process. The FSA process, or billing server, formats the raw billing data and transfers the formatted billing records to the distributed recording manager (DRM), which then writes the data to disk. Applications that generate data for recording on disk with this feature provide the service data types that can be mapped to a service data stream.

Multiple streams allow applications to separate different call types into different streams. For example, there can be different streams for operator service records and normal call records.

The stream connection manager (SCM) supervises connections between collectors and FSAs. The SCM uses the following factors to determine which collector connects to an FSA:

- DRM priority assignments
- FSAs registered with the SCM
- FSAs which are active
- FSAs which have disk space available for streams

The FSA process resides on one file processor. An FSA can also reside on the computing module (CM) for emergency backup purposes to automatic message accounting/device independent recording package, but application-specific support must support this. The collector process resides on the CM.

Datafill sequence and implications

Service data streams in SBSFMT table must be datafilled before datafilling service data types in table SBSMAP.

Adding or deleting dynamically created tuples in SBSMAP does not require restarts.

Table sizing

The SBSMAP table size 32 tuples.

Field descriptions

The following table describes field names and valid data ranges for table SBSMAP.

> **CAUTION**
> **Possible loss of billing data**
> Do not change stream mapping to NIL before all data is written to disk.

Field	Entry	Explanation and action
KEY	0–31	Represents a 16-character vector that defines the service data type. The service data type is in the range sbs_service_datatype_range. Key assignments can be manually or predefined by software.
STREAM	0–15	This field maps the service data type to a service data stream in the KEY field of table SBSFMT. This field is in the range sbs_service_datastream_fullrange. The effective range is 0 to 15. The NIL stream occupies position 16. Service data types cannot be mapped to the NIL stream.

Datafill example

The following example shows datafill for table SBSMAP.

```
KEY                                 STREAM
________________________________________________
OCC                                 OCC
```

Shadow

Table name

Shadow Sets (SHADOW)

Functional description

Table SHADOW contains a description of all shadow sets defined on a DMS SuperNode switch. Feature AR0517 (Shadow Set Maintenance) allows shadow sets to be created and deleted, and their members to be added and deleted, through datafill operations.

A tuple in table SHADOW contains data for an entire shadow set. A tuple identifies

- the node
- the shadow set's name
- location of the shadow set's permanent device
- location of each of the shadow set's current members

Note: File management uses a shadow set's permanent device to access the shadow set.

Successful datafill operations are communicated by way of dynamic, Distributed Data Manager (DDM) downloads of the tuple of the node that houses or must house the shadow set. Datafill changes take effect on the node only after such communication has taken place. If the node is isolated, member disks can continue to be read from and written to by applications on the node until a DDM audit of or bulk download to the node.

Datafill sequence and implications

The following tables must be datafilled before table SHADOW:

- APINV
- FPDEVINV

Table size

On each FP, there are at most 12 shadow sets, and there can be at most 100 FPs. This gives the tabulated limit of 1200 shadow sets on FPs. The table will expand and contract dynamically as tuples are added and deleted. Such addition and deletion does not require a restart.

Datafill

Field names, subfield names, and valid data ranges for table SHADOW are described below.

Field descriptions

Field	Subfield or refinement	Entry	Explanation and action
NODETYPE		FP	NODETYPE. This field is the first of a three-part key. This field identifies the node on which the shadow set resides.
NODENO		0 to 99	NODENUMBER. This field is the second part of a three-part key. Enter the instance number of the node identified in field NODETYPE, distinguishing one node from another of the same type. This field is applicable only to remote processors, for example, file processors (FP), since there is only one computing module (CM).
SETNAME		alphanumeric (up to 8 characters)	SHADOW SET NAME. This field is the third of a three-part key. Enter the shadow set's name.
DEVTYPE		SCSIDK, NILDEV	DEVICE TYPE. This is the device type. It is either a SCSI disk or nothing.
MEMBERS		see subfields	MEMBERS. List of members. This field is refined based on DEVTYPE.

—continued—

Field descriptions (continued)

Field	Subfield or refinement	Entry	Explanation and action
	DEVTYPE	SCSIDK, NILDEV	DEVICE TYPE. This is the device type. It is refined as: • SCSIDK. SCSI–Bus Location contains the subfields: — PERMSCSI. Permanent SCSI Disk. This is the shadow set's permanent device and is specified by a SCSI–bus number and a disk number. – SCSIBUS. SCSI–Bus. Enter a value of 0 to 1 – DEVNO. Disk Number. Enter a value of 0 to 5 — MEMSCSI. Other Permanent SCSI Device. This is a shadow set other than the permanent SCSI disk. Up to 4 other shadow sets can be specified. This is specified by a SCSI–bus number and a SCSI–disk number. – SCSIBUS. SCSI–Bus. Enter a value of 0 to 1 – DEVNO. Disk Number. Enter a value of 0 to 5 • NILDEV. NILDEV is not a valid entry. — NILPERM. value of 0 to 15
—end—			

Datafill example

The following example shows datafill for table SHADOW.

NODETYPE	NODENO	SETNAME	DEVTYPE	MEMBERS
FP	0	SS00	SCSIDK 0 0	(1 0)$
FP	2	SS00	SCSIDK 0 0	(1 0)$
FP	4	SS00	SCSIDK 0 0	(1 0)$
FP	4	SS01	SCSIDK 0 2	(1 2)$
FP	6	SS00	SCSIDK 0 0	(1 0)$
FP	6	SS01	SCSIDK 0 2	(1 2)$
FP	8	SS00	SCSIDK 1 0	$
FP	10	SS00	SCSIDK 0 0	(1 0)$

SUSHELF

Table name

Service Unit Shelf (SUSHELF)

> **ATTENTION**
> The material found in this module regarding table SUSHELF is UCS DMS-250 Billing Server-specific information. Refer to the *DMS-100 Customer Data Schema Reference Manual* for further details regarding datafill for table SUSHELF.

Functional description

For related information, refer to table LIUINV.

Table SUSHELF contains the components of the FLIS. An LIS can be supported by a transaction bus (T-bus) and F-bus interface (TFI) card on either a link peripheral processor (LPP) or a DMS-bus, or by an LIS F-bus controller (LFC) on the LIS if it is connected as a stand-alone LIS to subrate DS512 (SR512) fiber links of a DMS-bus. TFI-supported F-buses on an LPP or a DMS-bus are composed of a maximum of three or two LIS shelves, respectively. A DMS-bus can support a maximum of two stand-alone LIS shelves.

Datafill information

The following information must be taken into consideration when using table SUSHELF:

- The interface card and the port allocation must be symmetric.

  Note: Stand-alone LIS F-buses can be datafilled on NT9X17 cards with different link numbers (NUMLINKS) if the port number identified is compatible for links on both NT9X17 cards.

- Both F-buses of an LPP or MS must be in the offline state when their related entries in table SUSHELF are deleted or modified.
- No LIU can be datafilled in table LIUINV on a shelf being deleted or modified by table SUSHELF.
- LIU shelf PEC and card PECs must be compatible with the supporting interface card (NT9X73, LMS F-bus rate adapter card, or NT9X17AA, NT9X17AC, or NT9X17AD, message switch four-port cards, or NT9X17BB or NT9X17DA, DMS-bus 64-port cards).
- The physical location of an LIS supported by a TFI card must have the same cabinet location as the host LPP or DMS-bus. This is not a requirement for a stand-alone LIS supported by a DMS-bus.

- Variable shelf configurations must have the appropriate TFI F-bus terminations engineered.
- LIU shelves equipped with NT9X74BA/CA, F-bus repeater cards, or NT9X96AA, LIS F-bus controller cards must have proper NTEX20 intra-shelf termination paddle board terminations datafilled.

Datafill sequence and implications

The following tables must be datafilled before table SUSHELF.

- LIMCDINV
- MSCDINV
- TAPIDTAB

Table NIUINV must be datafilled after table SUSHELF.

Table sizing

0 to 55 tuples

The requirement is determined by multiplying the maximum number of F-bus controllers (max_lims + 1 (for DMS-bus) = 18) by the maximum number of LIU shelves (max_number_liu_shelves = 3).

Field descriptions

The following table describes field names, subfield names, and valid data ranges for table SUSHELF.

Field	Subfield or refinement	Entry	Explanation and action
SHELFKEY		see subfields	SHELF KEY. This field consists of subfields CONTROL, CTRLNUM, CARDNUM, PORTNUM, and LIUSHELF.
	CONTROL	LIM or MS	CONTROL. Enter LIM to indicate that the link interface module is the controlling entity. Enter MS to indicate that the message switch is the controlling entity.

—continued—

Field	Subfield or refinement	Entry	Explanation and action
	CTRLNUM	0 to 16 or NIL	CONTROL NUMBER. Enter the control number for the LIM. ***Note:*** The specified LIM must already be datafilled in table LIMINV. Enter NIL if the MS is the controlling entity.
	CARDNUM	5 to 23	CARD NUMBER. Enter the interface card number on the MS or the LIS. This entry identifies the interface card pair and must be a TFI card or an NT9X62BA, (four-port subrate DS512 paddle board) that supports the subrate DS512 (SR512) message links. The only valid entry for a TFI card is 12. Card allocation must be symmetric.
	PORTNUM	0 to 3	PORT NUMBER ON CARD. Enter a value that identifies the port on the interface card. Port allocation is symmetric. The only valid entry for ports on TFI cards is 0 (zero). Port numbers are validated against the front card and number of fiber links supported for ports on SR512 cards.
	LIUSHELF	0 to 3	LINK INTERFACE UNIT SHELF. Enter the shelf number identifying the LIU shelf. Each cabinet contains a possible four shelves. This shelf number must be the shelf address within the frame. • Enter 1 to 3 for LIU shelves with an LPP. • Enter 1 to 2 for LIU shelves with an MS, regardless of the position within the frame. A maximum of two LIU shelves can be equipped with MS tuples. • Any entry outside the range indicated for this field is invalid.

—continued—

Field	Subfield or refinement	Entry	Explanation and action
FLOOR		0 to 99	FLOOR. Enter the floor on which the cabinet resides
ROW		A to Z AA to ZZ (except I, O, II, OO)	ROW. Enter the row on the floor in which the cabinet resides, with the exception of I, O, II, and OO. The row numbers are indicated on the frame.
FRAMEPOS		0 to 99	FRAME POSITION. Enter the position of the LIS cabinet in the row.
FRAMETYP		EMC, LIM, or SCC	FRAME TYPE. Enter the frame type of the LIS cabinet. • Enter EMC for the MS LFC (fiber LIS) interface type cabinet. • Enter LIM for the LPP cabinet. • Enter SCC for the SuperNode SE TFI interface type cabinet. • Any entry outside the range of indicated values for this field is invalid.
FRAMENUM		0 to 511	FRAME NUMBER. Enter the number of the frame.
SHELFPOS		0 to 77	SHELF POSITION. Enter the base mounting position of the shelf. Standard base mounting positions are 0 (zero), 13, 26, and 39. For LIU shelves, enter 0 (zero) to 3.
SHELFPEC		NT9X72AA NT9X72BA NT9X72CA NT9X0810 NT9X7204	SHELF PEC. Enter the PEC of the shelf. This PEC identifies the maximum number of LIUs on the shelf. ***Note 1:*** All LIU shelves belonging to the same controller must have the same shelf PEC. ***Note 2:*** The NT9X72BA LIU shelf cannot be supported by an NT9X73AA TFI card. Any entry outside the range indicated for this field is invalid.

—continued—

Field	Subfield or refinement	Entry	Explanation and action
CARDINFO		see subfields	CARD INFORMATION. This field consists of subfields SLOT, FRONTPEC, and BACKPEC. It contains information on the cards for F-bus 0 and F-bus 1 on the LIS. Data is required for at least one card for each F-bus. All cards for F-bus 0 must be entered before F-bus 1 cards are entered.
	SLOT	7 to 32	SLOT. Enter the slot number of the card on the LIS • Enter 7 for required F-bus 0 card. • Enter 32 for required F-bus 1 card. • For optional F-bus 0 termination on an NT9X72AA shelf, enter 31. Otherwise, enter 30. • For optional F-bus 1 termination on an NT9X72AA shelf, enter 8. Otherwise, enter 10. ***Note:*** These slot numbers are corrected as required by table control.
	FRONTPEC	NT9X96AA	FRONT PEC. Enter the PEC of the front card. For SR512-supported LIS, enter NT9X96AA (LIS F-bus controller card).
	BACKPEC	NT9X98AA	BACK PEC. Enter the PEC of the back card. Enter NT9X98AA (LIS fiber interface paddle board) for SR512 supported LIS only. ***Note:*** PECs for TFI and LFC cards cannot be mixed.
—end—			

Datafill example

The following example shows sample datafill for table SUSHELF.

```
   SHELFKEY FLOOR ROW FRAMEPOS FRAMETYP FRAMENUM
SHELFPOS SHELFPEC
                                               CARDINFO
________________________________________________________

MS NIL 0 12 0 1  0   C   1      EMC      0
26       NT9X7204
                (7 NT9X96AA NT9X98AA) (30 NIL NTEX20AA)$
                (32 NT9X96AA NT9X98AA) (8 NIL NTEX20AA)$
```

Appendix C FCDRSRCH menu commands

This appendix describes the syntax, purpose, and semantics of the Flexible Call Detail Record Search (FCDRSRCH) commands. Table 11-1 lists and describes the commands. Additional information about the commands accessible in the FCDRSRCH directory can be found in either the *UCS DMS-250 Commands Reference Manual* or the *UCS DMS-250 Billing Records Application Guide.*

ATTENTION

The FCDRSRCH commands replace the CDRSRCH and DCDRSRCH commands.

With FCDRSRCH, you can extract records from a billing file residing on a disk volume. Records contain call information, catagorized by call types.

You can issue MAP terminal commands to

- select a billing file to be searched
- initiate a search
- define keys to compare against data in the call billing record fields
- set a logical condition among the defined keys
- impose constraints on the scope of the search
- view the records that are returned as a result of the search session

FCDRSRCH

FCDRSRCH commands for the computing module (CM) allow you to extract records of call information from billing files (or “core billing records”) on CM disk drive units (DDU) or Billing Server DDUs.

Office Parameters

Two office parameters are available to set FCDRSRCH usage limits:

- CDR_SEARCH_PROCESS_LIMIT – FCDRSRCH is a multi-user tool. CDR_SEARCH_PROCESS_LIMIT pertains to table OFCSTD and sets the number of simultaneous search processes allowed. The range is 0 – 20. Setting this parameter to 0 prevents any user from entering FCDRSRCH.
- SRCHOUTPUT – FCDRSRCH allocates output buffers to store CDRs which match the user's search keys. The SRCHOUTPUT office parameter pertains to table OFCENG and sets the number of output buffers allocated by each search process. The range is 0 – 200. Setting this parameter to 0 causes FCDRSRCH to return from execution without capturing any CDRs.

Access

FCDRSRCH commands are accessed through the following command:

>**MAPCI;MTC;APPL;OAMAP;FCDRSRCH**

Syntax

The syntax of the FCDRSRCH commands adhere to these guidelines.

- Commands that require one or more parameters are followed by a template of the parameter list.
- Each parameter in the input line must be separated by blank spaces.
- Commands that require non-optional parameters do not take effect until all non-optional parameters have been satisfactorily supplied. In fact, the MAP command interpreter requests all non-optional parameters until they are acceptably entered. Prompting does not occur for optional parameters unless they are optional as a group, and one or more, but not all, have been entered.
- The result of some FCDRSRCH commands is altered by the absence or presence of individual or groups of optional parameters when the command is invoked. These cases are fully described where applicable.

Related commands

Table 11-1 describes commands that are accessible in the FCDRSRCH directory. For further information about these commands, refer to either the *UCS DMS-250 Commands Reference Manual* or the *UCS DMS-250 Billing Records Application Guide.*

Table 11-1
FCDRSRCH commands

Command	Definition
BACKUP	BACKUP allows backward positioning within the output buffer from the current position.
BLOCK	BLOCK permits you to set the starting block number of the selected billing file for the next search session.
CDRTYPE	CDRTYPE changes the format form one search session to another.
DISPLAY	DISPLAY displays the contents of the ouput buffer.
EXECSRCH	EXECSRCH initiates each search session.
FORWARD	FORWARD allows forward positioning within the ouput buffer from the current position.
HELP	HELP displays the valid FCDSRCH commands when you use the NODISP option.
LISTFLDS	LISTFLDS provides a list of all billing record fields that pertain to the call record format for the selected billing file. It lists the names and correct spelling of the billing record fields to assist in defining search keys.
NUMOUT	NUMOUT provides an optional search constraint. NUMOUT limits the duration of a search session to the number of operations necessary to return a specific number of matched call detail records (CDR) to the output buffer.
NUMSRCH	NUMSRCH provides an optional search constraint. This constraint limits the duration of a search session to the number of operations necessary to search a specific number of CDRs within the billing file.
OPERATOR	OPERATOR toggles the logical condition (AND and OR) imposed among the defined search keys.
QUIT	QUIT enables you to exit from the current menu level and return to a previous menu level.
REINIT	REINIT, when invoked, reinitializes all search parameters currently defined with the exception of the search keys. All CDR data within the buffer is erased, and the current block number and time window for the last search are set to 0.
RSETKEY	RSETKEY is used to reset search keys, and allows the user to edit search key definitions.
SETKEY	SETKEY allows you to define the seach keys necessary to retrieve the desired CDRs from a selected billing file.
SRCHFIL	SRCHFIL is used to open a billing file in the DIRP or DRM recording systems.

—continued—

Table 11-1
FCDRSRCH commands (continued)

Command	Definition
STATUS	STATUS provides information about the last completed search, such as the number of records searched, the number of records output, and a time reference for the last record checked.
WINDOW	WINDOW allows you to conduct a search for call detail records (CDRs) within a defined time window of a billing file's active collection period.
—end—	

Appendix D DRM commands

This appendix describes the syntax and purpose of the distributed recording manager (DRM) commands and lists the commands in alphabetical order. The information in this appendix is intended to be a quick reference. For detailed information on DRM commands, refer to *Menu Commands Reference Manual* 297-1001-821.

All DRM commands are accessed at the following MAP level:

>MAPCI;MTC;APPL;OAMAP;DRM

AUDIT

Use the AUDIT command to request a comprehensive audit on an application and its resources, using the following syntax:

>AUDIT stream audit_type destnode

where

stream	is a valid stream datafilled in table SBSFMT
audit_type	is the kind of audit to be performed. See Table 1-1.
destnode	is the node to be audited (optional)

Sample entry: **>AUDIT occ appl fp0**

Table 12-1
AUDIT parameters

Audit_type	Description
all	performs all audits.
appl	sends an audit request to the application’s process.
fdir	rebuilds the DRM file directory.
rec	captures the recording measurements.

—continued—

Table 12-1
AUDIT parameters (continued)

Audit_type	Description
space	updates the space data for all in-service volumes; deletes P files as needed; raises or clears space alarms as needed; updates the rotate data used for prioritizing the selection of the next volume to rotate to.
vol	updates the status of the specified volume; if the status of a volume changes, a space audit is also performed.

—end—

DAT

Use the DAT command to copy DRM files to and from digital audio tape (DAT) or to query information about files on a DAT, using the following syntax:

>DAT subcommand

where

subcommand consists of QUERY, M(OUNT), INIT, W(RITE), V(ERIFY), R(ESTORE), or E(JECT). See Table 1-2.

Sample entry: **>DAT query**

The DAT subcommands have the following parameters:

Table 12-2
DAT parameters

Subcommand	Additional Parameters	Description
E (EJECT)		demounts the currently mounted DAT tape and ejects the tape from the drive. Mount another tape before issuing any other DAT commands.
M (MOUNT)		mounts the currently inserted tape. Issue this command first. Other parameters (w, v, r, and e) then reference the currently mounted tape. The tape remains mounted until an eject command is issued. This subcommand accepts the following parameters:

—continued—

Table 12-2
DAT parameters (continued)

Subcommand	Additional Parameters	Description
	dat_drive	specifies the name of the DAT tape drive to be mounted. The drive name format is CTyy, where yy is the DAT drive number.
	nodename	specifies the node where the DAT drive resides (FP0).
	tapeform	an optional parameter when NT and ANSI9 are the two values that can be selected for tape format.
	init	(optional) specifies an initial DAT tape mounting or to overwrite existing data. This command must be used when mounting a DAT tape for the first time; otherwise, the mount will fail. When you use init with an existing tape, the system writes a new VOLID onto the tape and overwrites previous content.
R (RESTORE)		copies a DRM file from tape back to disk. This subcommand accepts the following parameter:
	filetmpl	specifies the file name template, a required parameter if applid is supplied. It consists of the first character of a file name, followed by any number of characters that sufficiently identify one or more files. It must begin with U, P, R, or $. $ may be a wild card for any number of characters in the template. If the template is not a complete file name, a list of matched names displays, allowing you to select one.
	newstate	optional parameter that allows a P-file to be restored as an R-file. RE is the only valid value. Only P-files can be restored as R-files.
QUERY		displays information on currently mounted DAT devices.
V (VERIFY)		scans the contents of the mounted tape and displays information about the files found on the tape.

—continued—

Table 12-2
DAT parameters (continued)

Subcommand	Additional Parameters	Description
W (WRITE)		writes a file to the mounted DAT tape, placing it after the last file on the tape. If init was used in the mount process, the file writes at the beginning of the tape. This subcommand accepts the following parameters:
	applid	(optional) specifies the ID (key) of an application in table DRMAPPL. If not supplied, the system displays all keys from table DRMAPPL.
	filetmpl	specifies the file name template, a required parameter if applid is supplied. It consists of the first character of a file name, followed by any number of characters that sufficiently identify one or more files. It must begin with U, P, R, or $. $ may be a wild card for any number of characters in the template. If the template is not a complete file name, a list of matched names displays, allowing you to select one.
		—end—

DEMOUNT

Use the DEMOUNT command to remove a volume from a recording application's pool, making the volume and its contents inaccessible to the recording application/system, using the following syntax:

>DEMOUNT stream volname | volnum

where

stream	is a valid stream datafilled in table SBSFMT
volname	is the name of the volume being demounted. The name must match one of the volumes assigned to the stream in table DRMPOOL.
volnum	is the number of the volume in the application's pool in table DRMPOOL. The first position is 0.

Sample entry: **>DEMOUNT occ dk00occ1**

INFO

Use the INFO command to display registration, status, volume, file, audit, and rotate information, and to list the applications known to the DRM. Use the following syntax:

>INFO stream info_type

where

stream	is a valid stream datafilled in table SBSFMT
info_type	is the type of information being requested. See Table 1-3.

Sample entry: **>INFO occ all**

The info command has the following parameters:

Table 12-3
INFO parameters

Info_type	Description
fil	requests information on files matching the supplied filetmpl, fstates, or all files.
nodes	lists all DRM nodes used by the stream.
reg	lists registration information for each DRM node where the stream is registered.
rot	shows the rotate schedule and prioritized rotate volume list.
sched	shows the audit schedules.
vol	shows volume information
voldiag	shows trouble diagnostics information for all mounted volumes.
volerr	shows the last error detected on each mounted volume.
Optional Parameters	The following parameters are optional.
destnode	specifies the destination node name.
volume	specifies the volume name or reference number from DRMPOOL.
actv	specifies the active volume.
volnode	specifies the name of the node where volumes reside.

—continued—

Table 12-3
INFO parameters (continued)

Info_type	Description
filetmpl	specifies the first character of a file name, followed by any number of characters that sufficiently identify one or more files. $ may be used as a wild card for any number of characters in the template.
fstates	displays a list of file state characters. The list may be in any order and may not contain delimiter or separation characters.
all	refers to all volumes or all files, depending on the sub-command and position.
—end—	

MONITOR

Use the MONITOR command to remove an application's current active file in real time, using the following syntax:

>MONITOR stream

where

stream	is a valid stream datafilled in table SBSFMT

Samole entry: **>MONITOR occ**

The MONITOR command has no additional parameters.

MOUNT

Use the MOUNT command to assign a recording volume to an application, using the following syntax:

>MOUNT stream volname nodename [priority]

where

stream	is a valid stream datafilled in table SBSFMT
volname	is the name of the volume to mount. The name must match one of the volumes assigned to the stream in table DRMPOOL.
nodename	is the node (FP0) where the volume resides
priority	is 1 to 18

Sample entry: **>MOUNT occ dk00occ1 fpo**

RENAME

Use the RENAME command to rename a file by changing the file's state. See Chapter 5 of this manual for additional details. A file's state corresponds to the first character of its name: Active, Unprocessed, Processed, or Removed. This command allows the following file state transitions:

- U –> P, R
- P –> U, R
- R –> U, P

Use the following syntax for the RENAME command:

>RENAME stream filetmpl newstate

where

stream	is a valid stream datafilled in table SBSFMT
filetmpl	is the file template. It specifies the first character of a file name, followed by any number of characters that sufficiently identify one or more files. It must begin with U, P, R, or $. If the template is not a complete file name, a list of matched names displays, allowing you to select one. $ may be used as a wild card for any number of characters in the template.
newstate	is the new state for the file (U. P. or R)

Sample entry: **>RENAME occ an90102613r**

RESET

Use the RESET command to reset the error and rotate counters maintained by DRM per application and per node, using the following syntax:

>RESET stream counter_type destnode

where

stream	is a valid stream datafilled in table SBSFMT
counter_type	is the type of counter to be reset. See Table 1-4.
destnode	is the node (nodename or all) where the counters are to be reset. The default is the stream's active node.

Sample entry: **>RESET occ rot all**

The RESET command has the following parameters:

Table 12-4
RESET parameters

Counter_type	Description
err	resets only the error counters.
rot	resets only the rotate counters.

ROTATE

Use the ROTATE command to open a new recording file and close the old one, using the following syntax :

>ROTATE stream node volume

where

stream	is a valid stream datafilled in table SBSFMT
node	is the name of the node containing volumes from the stream's pool
volume	is the volume to which to rotate. It may be either a volume name or volume number from table DRMPOOL.

Sample entry: **>ROTATE occ fp0**

TCOPY

Use the TCOPY command to copy DRM files to a 9-track tape, using the following syntax :

>TCOPY device applid filetmpl

where

device	is the device drive. It specifies the name of a node containing volumes from the stream's pool.
applid	is the the volume to which to rotate. It may be either a volume name or volume number from table DRMPOOL.
filetmpl	is the file name template. It specifies the volume to which to rotate. It may be either a volume name or volume number from table DRMPOOL.

Sample entry: **>TCOPY 0 occ u**

VIEW

Use the VIEW command to get a hexadecimal dump of a given file, using the following syntax :

>VIEW stream filetmpl charcode blocknum

where

stream	is a valid stream datafilled in table SBSFMT
filetmpl	is the file name template. It specifies the first character of a file name, followed by any number of characters that sufficiently identify one or more files. It must begin with U, P, R, or $. If the template is not a complete file name, a list of matched names displays, allowing you to select one. $ may be used as a wild card for any number of characters in the template.
charcode	is the character code set to use (ASCII or EBCDIC) in translating the hex data
blocknum	is the location within the file to start viewing. The valid range is 1 to the number of blocks in the file.

Sample entry: **>VIEW occ un921216002600 as 1**

Appendix E
LOGUTIL commands

This appendix describes the syntax and purpose of the log utility (LOGUTIL) commands and lists the commands in alphabetical order. The information in this appendix is intended to be a quick reference. For detailed information on LOGUTIL commands, refer to *Nonmenu Commands Reference Manual* 297-1001-820.

All LOGUTIL commands are accessed through the following command:

>LOGUTIL

ADDCLASS

Use the ADDCLASS command to add classes to those printed by a device, using the following syntax:

>ADDCLASS io_device classnum

where

io_device	is the input/output (I/O) device
classnum	is the class number or numbers to add

Sample entry: **>ADDCLASS prt 2**

ADDREP

Use the ADDREP command to add reports to those handled by a device, using the following syntax:

>ADDREP io_device logname repnum

where

io_device	is the input/output (I/O) device
logname	is the log name or names
repnum	is the report number or numbers

Sample entry: **ADDREP prt topp 100**

BACK

Use the BACK command to display the log report entry before this current log report in the log buffer, using the following syntax:

>BACK number | all

where

number	is the number of entries back from the current report that you wish to display. The range is 1 to 32767; the default is 1.
all	specifies that all of the prior reports display

Sample entry: **>BACK 2**

BACKUP

Use the BACKUP command to make an archive copy of reports, using the following syntax:

>BACKUP io_device 1 by io_device 2

where

io_device 1	is the input device for the backup copy
by	is required to clarify the command syntax
io_device	is the output device for the backup copy

Sample entry: **>BACKUP d000scratch by d010scratch**

CLASS

Use the CLASS command to set the class of selected reports, using the following syntax:

>CLASS classnum logname repnum

where

classnum	is the class number, from 0 to 31
logname	is the log name or names
repnum	is the report number or numbers

Sample entry: **>CLASS 4 ddu 213**

CLEAR

Use the CLEAR command to delete all reports from a log, using the following syntax:

>CLEAR logname

where

logname	is the name of the log to clear

Sample entry: **CLEAR aud**

CONTEXT

Use the CONTEXT command to change the context of applicable nodes for the browsing command during the current session, using the following syntax:

>CONTEXT nodename nodenumber unitnumber

where

nodename	is the name of a type of node that generates logs in the switch. For ENET, this variable is the plane number
nodenumber	is a specific node within a given type. For ENET, this variable is the shelf number.
unitnumber	is the unit number for those nodes where it is needed

Sample entry: **>CONTEXT enet 0 0**

DELCLASS

Use the DELCLASS command to delete the classes from those printed by a device, using the following syntax:

>DELCLASS io_device classnum

where

io_device	is the input/output (I/O) device
classnum	is the class number, from 0 to 31

Sample entry: **>DELCLASS prt1 2**

DELDEVICE

Use the DELDEVICE command to delete a device from the log system, using the following syntax:

>DELDEVICE io_device

where

io_device	is the input/output (I/O) device to delete

Sample entry: **>DELDEVICE prt1**

DELREP

Use the DELREP command to delete report(s) from those handled by a device, using the following syntax:

>DELREP io_device logname repnum

where

io_device	is the input/output (I/O) device that handles the report(s)
logname	is the log name or names
repnum	is the report number or numbers to delete

Sample entry: **>DELREP prt aud 107**

DUMPLOGS

Use the DUMPLOGS command to display the log reports in a log buffer in chronological order as they were generated, using the following syntax:

>DUMPLOGS logname lognumber

where

logname	is the log name or names
lognumber	is the log number, from 0 to 999

Sample entry: **>DUMPLOGS aud**

FIRST

Use the FIRST command to print the first report entry.

This command has no parameters.

FORMAT

Use the FORMAT command to set and query the output format of the reports, using the following syntax:

>FORMAT normal | short

where

normal	prints the report(s) in normal format
short	shows only the first line of the normal log report(s)

Sample entry: **>FORMAT short**

FORWARD

Use the FORWARD command to display the report entry after the current one, using the following syntax:

>FORWARD number | all

where

number	specifies how many reports you wish to display. The range is 1 to 32767; the default is 1.
all	specifies that all reports after the current entry display

Sample entry: **>FORWARD 5**

LAST

Use the LAST command to print the last report entry.

The LAST command has no parameters.

LISTDEVS

Use the LISTDEVS command to list the input/output devices defined in the log system, using the following syntax:

>LISTDEVS brief | full

where

brief	specifies a brief report of the devices available. It is the default.
full	specifies a full report of the devices available.

Sample entry: **>LISTDEVS full**

LISTLOGS

Use the LISTLOGS command to list all the logs that have been defined.

The LISTLOGS command has no parameters.

LISTNODES

Use the LISTNODES command to list all the nodes in the switch.

The LISTNODES command has no parameters.

LISTREPS

Use the LISTREPS command to list all report types for a selected log or class, using the following syntax:

Note: Using this command without specific report types may take several minutes to list.

>LISTREPS special | syslog logname | class repnum | classnum

where

special	lists only the suppressed and threshold reports
syslog	lists only the syslog reports
logname	is the log name or names for which to list reports
class	specifies that the number that follows is a class of report
repnum	is the report number or numbers to list
classnum	is the class number, from 0 to 31

Sample entry: **>LISTREPS rman 131**

LISTROUTE

Use the LISTROUTE command to list routing information, using the following syntax:

>LISTROUTE device / class / report

where

device	specifies the routing for devices
io_device	is the device or devices
class	specifies the routing for a class of reports
classnum	is the class number or numbers, from 0 to 31
report	specifies the routing for reports
logname	is the log name or names
repnum	is the report number or numbers

Sample entry: **>LISTROUTE report iod 120**

LISTTIME

Use the LISTTIME command to list the reports that are on the reset schedule.

The LISTTIME command has no parameters.

MODE

Use the MODE command to set the query mode of logs for use with the browsing commands. The mode is set at the time you enter the LOGUTIL directory or when you run the mode command during the session. Use the following syntax:

>MODE craft | expert

where

craft	specifies that only craft logs are available for display. The mode is set to craft when you enter the LOGUTIL directory.
expert	specifies that all logs are available for display

Sample entry: **>MODE expert**

OPEN

Use the OPEN command to prepare a log buffer for display, using the following syntax:

>OPEN first | logname lognumber

where

first	specifies the first log in the buffer as a default
logname	is the log name or names to display
lognumber	specifies the log report number to display

Sample entry: **>OPEN aud 120**

RENUMBER

Use the RENUMBER command to assign report numbers to all reports without one.

The RENUMBER command has no parameters.

REROUTE

Use the REROUTE command to reroute the specified devices to their backups, using the following syntax:

>REROUTE English | lang io_dev

where

English	specifies the report is printed in English as a default
lang	specifies the report is printed in a language other than English
io_device	specifies the output device

Sample entry: **>REROUTE d000scratch**

RESET

Use the RESET command to reset all thresholds and turn off all suppression.

The RESET command has no parameters.

RESETROUTE

Use the RESETROUTE command to reset all the routing information from the LOGCLASS and LOGDEV tables. All temporary routing from CLASS, ADDCLASS, DELCLASS, and REROUTE commands is lost.

The RESETROUTE command has no parameters.

RESUME

Use the RESUME command to resume generating selected reports, using the following syntax:

>RESUME logname repnum

where

logname	specifies the log name
repnum	specifies the report number

Sample entry: **>RESUME iod 120**

RESUMEDEV

Use the RESUMEDEV command to resume printing logs at a particular device, using the following syntax:

>RESUMEDEV cm | allnodes io_dev

where

cm	specifies reports from the central node as a default
allnodes	specifies that logs generated on all nodes print on the given device(s)
io_device	specifies the device where printing is to occur

Sample entry: **>RESUMEDEV allnodes rp121**

START

Use the START command to start printing reports on this terminal as they are generated, using the following syntax:

>START ascii | ebcdic class

where

ascii	specifies that reports are generated in ASCII as a default
ebcdic	specifies that reports are generated in EBCDIC
class	specifies the class of report, from 0 to 31

Sample entry: **>START 2**

STARTDEV

Use the STARTDEV command to start printing reports at the particular device(s), using the following syntax:

>STARTDEV ascii | ebcdic English | lang central | allnodes io_dev

where

ascii	specifies that the log data is recorded in ASCII as a default
ebcdic	specifies that the log data is recorded in EBCDIC
English	specifies that the report is printed in English as a default
lang	specifies that the report is printed in a language other than English
central	specifies that the central node logs are printed, as a default
io_device	specifies the output device(s) for the logs

Sample entry: **>STARTDEV all nodes rp121**

STOP

Use the STOP command to stop printing reports on the current device.

The STOP command has no parameters.

STOPDEV

Use the STOPDEV command to stop printing reports at the particular device(s), using the following syntax:

>STOPDEV central | allnodes io_dev

where

central	specifies that the central node logs are printed, as a default
allnodes	specifies that the logs from all nodes are stopped on the given device(s)
io_device	specifies the output device(s) for the log reports

Sample entry: **>STOPDEV all nodes rp121**

SUPPRESS

Use the SUPPRESS command to stop generating selected reports, using the following syntax:

>SUPPRESS logname repnum

where

logname	specifies the log name or names
repnum	specifies the report number or numbers

Sample entry: **>SUPPRESS cm 108**

THRESHOLD

Use the THRESHOLD command to set the threshold for selected reports, using the following syntax:

>THRESHOLD threshold logname repnum

where

threshold	specifies the threshold value
logname	specifies the log name or names
repnum	specifies the report number or numbers

Sample entry: **>THRESHOLD 4 cm 108**

TIMERESET

Use the TIMERESET command to periodically reset report counts for the threshold, using the following syntax:

>TIMERESET minutes logname repnum

where

minutes	specifies the minutes to wait before resetting the report count
logname	specifies the log name or names
repnum	specifies the report number or numbers

Sample entry: **>TIMERESET 1 cm 108**

TYPE

Use the TYPE command to print the current report entry.

The TYPE command has no parameters.

Appendix F
Disabling the FP and the SBS

Purpose

This appendix provides information that allows an operator to disable the FP (Billing Server) and the SuperNode Billing System (SBS) from in a UCS DMS-250 system. This process is necessary after the transition billing streams from the UCS DMS-250 Billing Server billing application to the SuperNode Data Manager (SDM).

This is not a full Billing Server decommission procedure. Contact your next level-of–support for information on how to remove the UCS DMS-250 Billing Server hardware.

The following steps are necessary to migrate billing from the UCS DMS-250 Billing Server billing application to the SuperNode Data Manager (SDM):

1 Configure the SDM SuperNode Billing Application (SBA). Refer to the *SuperNode Billing Application Guide*, 297-8991-300, for information on how to do this.

2 Transition the FP billing information to the SDM SBA. Refer to the *SuperNode Data Manager User Guide*, 297-5051-900 for the FP to SDM SBA Billing Transition Procedure.

3 Execute the steps in the following procedure.

Constraints

The following constraints apply to performing this procedure:

- Personnel responsible for performing any of the steps in this procedure must read the procedure before starting it.
- SDM installation and configuration must be completed by Nortel Networks before you execute this procedure. The SuperNode Data Manager (SDM) and the Supernode Billing Application (SBA) must be in-service before performing this procedure.

- The Supernode Billing Application billing streams should be configured, set to “ON” mode, and should be in-service.

> **CAUTION**
> **Possible loss of service**
> Failure to properly confugure an SDM and transition billing data from the FP to the SDM will cause a loss of billing data.

Procedure

The following procedure provides steps to disable the UCS DMS–250 Billing Server Application software after the transition of billing to the SDM SBA. Be sure to read any Notes and “if” statements in the procedure steps carefully.

1 Check to see if the billing server is active. This is done by the following commands.

>MAPCI;MTC;APPL;OAMAP;SBS;STRMSTAT

An example response follows:

```
STREAM BS STATUS ALARM FP0
  OCC  D
```

Do not execute any of the following steps, if the ”BS” column still shows ”S”. If the ”BS” column still shows ”S”, return to the FP to SDM Transition Procedure in the *SuperNode Data Manager User Guide.*

Note: The area under BS indicates which billing system is currently assigned to the stream. S means billing is going to SBS; while D means the billing is being handled by DIRP.

2

If	Then
If the operational measurements (OMs) are collected, and will continue to be collected, in the distributed recording manager (DRM) in FP volumes	do not execute any of the following steps
If the operational measurements (OMs) are not collected in DRM (they must be rerouted from DRM to DIRP)	continue

IChange the OMTAPE table to enable the DIRP system to collect the OMs.

>TABLE OMDATA; LIS ALL; QUIT

Note: Check to see which classes are active and change the corresponding classes in OMTAPE table.

>TABLE OMTAPE; OVE; POS <number>

where

number is the class obtained from the previous command.

>CHA 2 Y
>Y (for confirmation)
>QUIT

3 Repeat these commands for classes that were active in the OMDATA table

Note: Change the OMDATA table to disable the distributed recording manager (DRM) system from collecting OMs.

>TABLE OMDATA
>CHA 2 N
>Y
>QUIT
(repeat these commands for all classes that are active)

Note: Ensure the same changes are made on the inactive node.

Note: Transfer remaining OMs before disabling the FP.

4 Rotate the active billing file on the Billing Server using the following command to close the file.

>MAPCI;MTC;APPL;OAMAP;DRM
>ROTATE OCC
>ROTATE OM (if applicable)

Note: Jot down the DRM file name prior to DRM rotation. You will need the file name in the next step.

See the active file in the DRM ”monitor” map, for example, ”monitor OCC”, or ”monitor OM.”

>MAPCI;MTC;APPL;OAMAP;DRM
>MONITOR OCC
>MONITOR OM

5 Ensure the currently transferring file has completed transferring before continuing. Verify that all ”U” files were transferred. Complete transfer of any untransferred ”U” files. Verify that most recent FTAM701 log indicates file transfer completed successfully (FTAM701 shows ”completed”).

>LOGUTIL
>CONTEXT EIU <*n*>
>OPEN FTAM 701

6 ***Note:*** Do not begin procedure from this step unless you are disabling the FP.

After the files have transferred to the downstream processor, Busy and OFFL FLIS EIUs. Use Table LIUINV to find the correct EIU numbers. Post only the EIUs with an OSU load.

>TABLE LIUINV
>FORMAT PACK
>LIST ALL
>MAPCI;MTC;PM;POST EIU # #
>BSY ALL
>OFFL ALL
Example: MAPCI;MTC;PM;POST EIU 100 101

7 Ensure all shadow sets are in sync before continuing. If a shadow set is *not* in sync and is *not syncing,* determine the cause for the condition and resolve before continuing. If a shadow set is *not* in sync and *is syncing*, wait for synchronization to complete before continuing.

>SHADOWUT FP <fp_num>
>DIS ALL
Note: Example response follows:

```
Information about the shadow set #0:

Node name:                         FP0
Shadow set name:                   SS0
Set definition state:              RUNNING
Set operational state:             IN SERVICE
Synchronization status:            SYNCHRONIZED
Multi-Writes:                      Serial
Capacity (blocks):                 1244655
Transfer Length:                   Optimal
Interval:                          0
```

8 Install and activate the ELD20 patch from the *Post–Release Software Manager (*PRSM).

For information on how to install and activate a patch, refer to the *Post–Release Software Manager Reference Manual*, 297-8991-540.

9 Demount all OCC volumes, and demount all OM volumes using the following commands:

>MAPCI;MTC;APPL;OAMAP;DRM
>INFO OCC VOL
>INFO OM VOL
Note: Make a note of the volumes mounted, including their numbers.

>QUIT
>DEMOUNT OCC <vol_number>
>DEMOUNT OM <vol_number> (if applicable)
Example: DEMOUNT OCC 0

Reset each demount until done.
Verify with that all volumes are demounted by entering:

>INFO OCC VOL
>INFO OM VOL

10 Stop shadowing for all shadow sets using the following command.

>STS <shadow_set>
>YES
Example: STS SS0OCC0

This step may take 10 minutes to complete. Please wait for Stop Shadow Completion Log (FP 505).

>QUIT

11 Busy the FP using the following command.

>MAPCI;MTC;PM;POST FP <fp_0>
>BSY
>YES

12 Offline the FP using the following commands.

>OFFL
>YES

13 Remove datafill from table APINV using the following commands.

>TABLE APINV

Postion the tuple

>POS FP 0
>DEL

Exit out of the table

>

>QUIT
or

>QUIT ALL

14 Remove billing stream processes from the CM. Type

>TABLE SBSMAP
>POS OCC; CHA 2 NIL

and press the **ENTER** key.

Verify that you want to remove billing stream from the CM by typing

>y
and press the **ENTER** key.

The CM responds with a major SBS DK alarm.

Note:

Note: When the OCC stream is removed from the billing stream format table (SBAFMT) and the SBSMAP, the core process SBSCLP will cease on the CM and the SBSFMP processes will cease on the FP. This means that there will no longer be an Format Storage Agent (FSA). For sites utilizing more than one stream (for instance you may have an OCC and OCCA stream) each stream will own a a set of core SBSCLP and FP SBSFMP processes.

15 Repeat previous step if you have more than one stream (such as an OCCA stream) to remove.

16 Verify the alarm condition by typing

>MAPCI;MTC;APPL;OMAP;SBS;SBSSTAT
and press the **ENTER** key.

The CM responds that the stream has no disk

17 Exit the MAPCI by typing

>QUIT ALL
and pressing the **ENTER** key.

18 Remove the billing stream format table from the CM. Type

>TABLE SBSFMT
and press the **Enter** key.

19 Type

>POS OCC DEL
and press the **Enter** key.

20 Verify that you want to remove the billing format table from the CM by typing

>Y
and press the **ENTER** key.

The CM responds by clearing the major SBS DK alarm.

21 Repeat step 19 and 20 if you have more than one stream (such as an OCCA stream) to remove.

22 Type

>TABLE IPHOST
and press the **ENTER** key.

23 Locate the FP entry number. Type

>LIST ALL
and press the **Enter** key.

24 Remove the FP entry number. Type

>POS ENTRY_NUM;DEL
and press the **ENTER** key.

where

entry_num   is the number of the entry beside the FP listing.

25 Confirm that you want to remove the entry from the table by typing

>Y
and press the **ENTER** key.

26 Display and record the name of the billing files on the billing server once again. Type

27 Deactivate ELD20 in PRSM (Post–Release Software Manager) system.

For information on how to deactivate a patch, refer to the *Post–Release Software Manager Reference Manual*, 297-8991-540.

28 The procedure is complete.

Note: Contact your next level–of–support for information on how to remove the UCS–250 Billing Server hardware.

List of terms

APC
Applications Processor Cabinet

CDR
call detail record

CI
command interpreter

CM
computing module

CPU
central processing unit

CSMA/CD
carrier sense multiple access with collision detection

DAT
digital audio tape

DDU
disk drive unit

DIRP
Device Independent Recording Package

DMS
Digital Multiplex Switching

DRM
Distributed Recording Manager

EIU
Ethernet Interface Unit

ENET
enhanced network

ETAS
emergency technical assistance

F-bus
frame transport bus

FCDRSRCH
flexible call detail record search

FLIS
fiberized link interface shelf

FP
file processor

FTAM
File Transfer, Access, and Management

FTFS
Fault Tolerant File System

FTP
File Transfer Protocol

FTS
file transport system

INM
integrated node maintenance

IOC
input/output carrier

IP
Interface Paddleboard

ISO
International Standards Organization

LAN
local area network

LIM
link interface module

LIS
link interface shelf

LIU
link interface unit

LPP
link peripheral processor

MAC
media access control

MAP
Nortel trademark. Testing and maintenance center for use with the switch.

MAU
media access unit

MS
message switch

NET
network

NFS
network file system

Nortel
Northern Telecom

OCC
Other Common Carrier

OM
operational measurement

OSI
Open Systems Interconnection

OSR
operator services record

PEC
product engineering code

PM
peripheral module

RFTFS
: remote fault tolerant file system

RP
: resource processor

RTIF
: Remote Terminal Interface

RTS
: return to service

SCSI
: small computer standard interface

SDIP
: SCSI device interface paddleboards

S/DMS
: Nortel trademark. SuperNode/DMS

S/OM
: Nortel trademark. SuperNode Operational Measurements

SIP
: SCSI interface paddleboard

SLM
: system load module

SSLPP
: single shelf link peripheral processor

T-bus
: transaction bus

TCP/IP
: transmission control protocol/internet protocol

TPDU
: transport layer protocol data unit

Ordering information

Use the following table for ordering Nortel NTPs (Nortel Networks Technical Publications) and Product Computing-Module Loads (PCLs):

Type of product	Source	Phone	Cost
Technical documents (paper or CD-ROM)	Nortel Product Documentation	1-877-662-5669	Yes
Individual NTPs (paper)	Merchandising Order Service	1-877-662-5669	Yes
Marketing documents	Sales and Marketing Information Center (SMIC)	1-800-4NORTEL (1-800-466-7835) * ESN 444-5930	No

When ordering publications on CD

Please have the CD number and software version available, for example, **HLM–2621–ENCDRPDF 06.02**.

When ordering individual paper documents

Please have the document number and name available, for example, **297-2621-001, UCS DMS-250 Master Index of Publications**.

When ordering software

Please have the eight-digit ordering code, for example, **UCS00012**, as well as the ordering codes for the features you wish to purchase. Contact your Nortel Networks representative for assistance.

Digital Switching Systems

UCS DMS-250

Billing Server Application Guide

Product Documentation–Dept 3423
Nortel Networks
P.O. Box 13010
RTP, NC 27709–3010
1–877–662–5669

Publication number: 297-2621-320
Product release: UCS15
Document release: Standard 09.01
Date: May 2001
Printed in the United States of America

NORTEL NETWORKS™
How the world shares ideas.