CONNECT:Direct™
for VM/ESA

Installation Guide

Version 3.2

CONNECT:Direct for VM/ESA Installation Guide
Version 3.2
First Edition

This document was prepared to assist licensed users of the Sterling Commerce, Inc., CONNECT:Direct for VM/ESA system; its contents may not be used for any other purpose without prior written permission. The material contained herein is supplied without representation or warranty of any kind and is based on typical use. Any unusual use may produce unpredictable results. Sterling Commerce, therefore, assumes no responsibility and shall have no liability of any kind arising from the supply or use of this document or the material contained herein.

References in this manual to Sterling Commerce products, programs, or services do not imply that Sterling Commerce intends to make these available in all countries in which Sterling Commerce operates.

Product Enhancement Request Form

Company Name:_____

Address:_____

City:_____**State:**_____**Zip:**_____

Requester:_____**Telephone:**_____

Product:

___ **CONNECT:Direct for MVS**	___ **CONNECT:Mailbox for VSE**
___ **CONNECT:Direct for VSE**	___ **CONNECT:Mailbox for MVS**
___ **CONNECT:Direct for VM**	___ **CONNECT:Mailbox for OS/400**
___ **CONNECT:Direct for MSP**	___ **CONNECT:Mailbox SPC Option**
___ **CONNECT:Direct for NetWare**	___ **CONNECT:Mailbox for UNIX**
___ **CONNECT:Direct for Tandem**	___ **CONNECT:Supertracs for MVS**
___ **CONNECT:Direct for UNIX**	___ **CONNECT:Supertracs for VSE**
___ **CONNECT:Direct for Stratus VOS**	___ **CONNECT:Tracs for OS/2**
___ **CONNECT:Direct for OpenVME**	___ **CONNECT:Tracs for MVS**
___ **CONNECT:Direct for OpenVMS**	___ **CONNECT:Tracs for MS-DOS**
___ **CONNECT:Direct for OS/400**	___ **CONNECT:Tracs for VSE**
___ **CONNECT:Direct for OS/2**	___ **CONNECT:Queue**
___ **CONNECT:Direct for OS/2 Warp**	___ **CONNECT:Firewall**
___ **CONNECT:Direct for MS-DOS**	___ **CONNECT:Remote for Windows NT**
___ **CONNECT:Direct for Windows NT**	
___ **CONNECT:Direct for RemoteWare**	
___ **CONNECT:Direct Requester for Motif**	
___ **CONNECT:Direct Requester for Windows**	

Current Product Release Level: _____

Describe enhancement:

_____

_____

_____

_____

Over

How will enhancement be used?

_____

_____

_____

_____

_____

_____

Would you install a new release of this CONNECT product to get the enhancement?
___ Yes   **___ No**

Enhancement priority:

___ Need ASAP
___ Have immediate plan to use when available
___ Good idea; nice to have

Additional comments:

_____

_____

_____

_____

_____

_____

_____

Mail or fax this completed form to:

Sterling Commerce
Communications Software Group
Attention: Product Marketing
5215 North O'Connor Boulevard, Suite 1500
Irving, Texas 75039-3771
Fax: (972) 868-5142

Thank you for your interest in the Sterling Commerce CONNECT Product Family.

Contents

Preface

Part I   Preparation

Chapter 1   About CONNECT:Direct for VM/ESA

Chapter 5   Preparing VTAM and NCP Parameters

Part II   Installation

Chapter 6   Installing CONNECT:Direct for VM/ESA

Chapter 7   Tuning CONNECT:Direct for VM/ESA for Performance

Appendix A   Installation Worksheets

Appendix B   CONNECT:Direct for VM/ESA Tape Exit

Appendix C   VTAM Definitions

Glossary

Index

Preface

The *CONNECT:Direct for VM/ESA Installation Guide* is for programmers and network operations staff who plan the installation, install the product, and tune CONNECT:Direct for VM/ESA.

Read the first part of the book to gain a general knowledge necessary to install CONNECT:Direct. These chapters introduce you to the basic components, general concepts, and preinstallation procedures.

Read the second part of the manual after reading the first part. The second part of the manual provides information necessary to install and tune CONNECT:Direct.

This manual assumes you have a working knowledge of the Virtual Machine/ Enterprise System Architecture (VM/ESA) operating system and Interactive Systems Productivity Facility (ISPF).

Chapter Overview

The *CONNECT:Direct for VM/ESA Installation Guide* contains the following chapters:

▶ Part I, *Preparation*, provides an introduction to CONNECT:Direct and an overview of preinstallation procedures.

– Chapter 1, *About CONNECT:Direct for VM/ESA*, describes the CONNECT:Direct product, including a brief description of internal components and concepts.

– Chapter 2, *Planning the Installation*, outlines the steps necessary to prepare for the installation of CONNECT:Direct.

- Chapter 3, *Planning the DASD Requirements*, presents DASD requirements for CONNECT:Direct VSAM files, a description of the VSAM files, and the virtual storage requirements for the Data Transmission Facility (DTF).

- Chapter 4, *Preparing VTAM Definitions*, explains how to set up VTAM definitions for CONNECT:Direct, complete mode table entries, and activate cross-domain resources.

- Chapter 5, *Preparing VTAM and NCP Parameters*, explains how to set up VTAM and NCP parameters that you may need to define for CONNECT:Direct for VM/ESA.

▶ Part II, *Installation*, provides installation procedures, configuration descriptions, and tuning information for CONNECT:Direct.

- Chapter 6, *Installing CONNECT:Direct for VM/ESA*, outlines the installation steps of CONNECT:Direct.

- Chapter 7, *Tuning CONNECT:Direct for VM/ESA for Performance*, provides tuning guidelines for statistics file pairs and the CMS Shared File System file server to enhance performance and usage.

▶ Appendices

- Appendix A, *Installation Worksheets*, provides installation worksheets to help you during your installation.

- Appendix B, *CONNECT:Direct for VM Tape Exit*, outlines the steps for using VMTAPE with CONNECT:Direct for VM/ESA.

- Appendix C, *VTAM Definitions*, contains examples to assist you in completing the installation.

Glossary defines terms used throughout the book.

Notational Conventions

This section describes the notational conventions used in this guide.

Uppercase Letters

Uppercase letters in the command format indicate that you type in information as shown.

Uppercase and Lowercase Letters

A statement, command, or parameter in uppercase letters followed by lowercase letters indicates an alternative to typing the entire command. For example, SELect PROCess means that you need only type SEL PROC for the command to be valid.

Lowercase Letters

Lowercase letters or words in commands or syntax boxes require substitution by the user. For example, PNODE=primary-node-name indicates that you must provide the name of the primary node.

Bold Letters

Bold print in syntax boxes indicates CONNECT:Direct commands and required parameters. For example, **DSN=filename** indicates that the parameter *DSN* is required.

Commands, Process statements, parameters, and special keys are sometimes bold in text to differentiate them from other words.

Underlined Letters

Underlining indicates default values for parameters and subparameters. For example, RETAIN=Yes|<u>No</u>|Initial specifies that the default for *RETAIN* is *NO*.

Vertical Bars

Vertical bars indicate that you can supply one of a series of values separated by the vertical bars. For example HOLD=Yes|No|Call specifies that *Yes* or *No* or *Call* is valid.

Brackets

Brackets indicate that information is optional. For example, STARTT=([date|day][,hh:mm:ssXM]) indicates that you can specify either a date or a day, a date or a day plus a time, or just a time.

Additional Notations

Code all commas and parentheses as they appear.

Process, as shown with a capital **P**, refers to a CONNECT:Direct Process.

`Monospaced characters` (characters of equal width) represent information for screens, commands, Processes, and reports.

Italics indicate book, chapter, and section titles or show emphasis in the text.

CONNECT:Direct for VM/ESA Documentation

Use the *CONNECT:Direct for VM/ESA Installation Guide* in conjunction with other CONNECT:Direct product documents. The following manuals make up the CONNECT:Direct for VM/ESA library:

▶ *CONNECT:Direct Release Notes* is a document shipped with CONNECT:Direct for VM/ESA that lists system requirements, maintenance updates, and enhancements.

▶ *CONNECT:Direct for VM/ESA User's Guide* is a manual that defines all CONNECT:Direct commands and explains how to use them in the Interactive User Interface (IUI), Batch Interface, CMS Command Line Interface, and other utilities.

▶ *CONNECT:Direct for VM/ESA Administration Guide* is a manual for the administration and maintenance of CONNECT:Direct. This guide contains information about CONNECT:Direct tasks, native commands, and other administrative tasks.

▶ *CONNECT:Direct Process Guide* provides you with the information needed to write a CONNECT:Direct Process for the MVS, VSE/ESA, VM/ESA, MSP, OS/400, Tandem, UNIX, VOS, and VMS platforms.

▶ *CONNECT:Direct Console Operator's Guide* is a manual for the operator who initiates CONNECT:Direct for MVS, VSE/ESA, and VM/ESA data transfer activities from the operator console.

▶ *CONNECT:Direct Problem Isolation Guide* is a manual that explains how to fix errors encountered when performing CONNECT:Direct for MVS, VSE/ESA, and VM/ESA functions.

▶ *CONNECT:Direct Quick Reference* is a capsulized reference of CONNECT:Direct for MVS, VSE/ESA, and VM/ESA Process statements, commands, and installation parameters.

▶ *CONNECT:Direct Technical Overview* is an introduction to the CONNECT:Direct product family and its data transfer applications.

▶ *CONNECT:Direct Event Services Support System Guide* provides information on the system architecture, system operation, and event data format for the Event Services Support.

Part I

Preparation

About CONNECT:Direct for VM/ESA

CONNECT:Direct distributes information and manages production activities among multiple mainframes, minicomputers, workstations, and personal computers in diverse operating system environments.

CONNECT:Direct allows data centers within and across networks to:

▶ Move large amounts of data
▶ Share information
▶ Schedule related application activities
▶ Automate data distribution
▶ Control and audit network activities
▶ Maintain network security
▶ Use a common command structure and environment-specific interfaces

CONNECT:Direct goes beyond traditional file transfer systems by eliminating the time-consuming, error-prone operator procedures associated with moving data. CONNECT:Direct capabilities extend from basic data movement functions to the management of data movement activities.

CONNECT:Direct for VM/ESA Internal Components

This section provides an overview of the internal components of CONNECT:Direct for VM/ESA and a brief description of each of the CONNECT:Direct interfaces.

The three internal components of CONNECT:Direct for VM/ESA are:

▶ Various user interfaces to CONNECT:Direct for VM/ESA
▶ Application Program Interface (API)
▶ Data Transmission Facility (DTF)

While CONNECT:Direct is running as an application on the VM/ESA operating system, the system components interact to execute the statements and commands submitted through the interfaces.

Note: All statements and commands must pass through the API regardless of the interface from which they are submitted.

The following figure shows a single CONNECT:Direct for VM/ESA. Each piece of the diagram is discussed in the pages that follow.
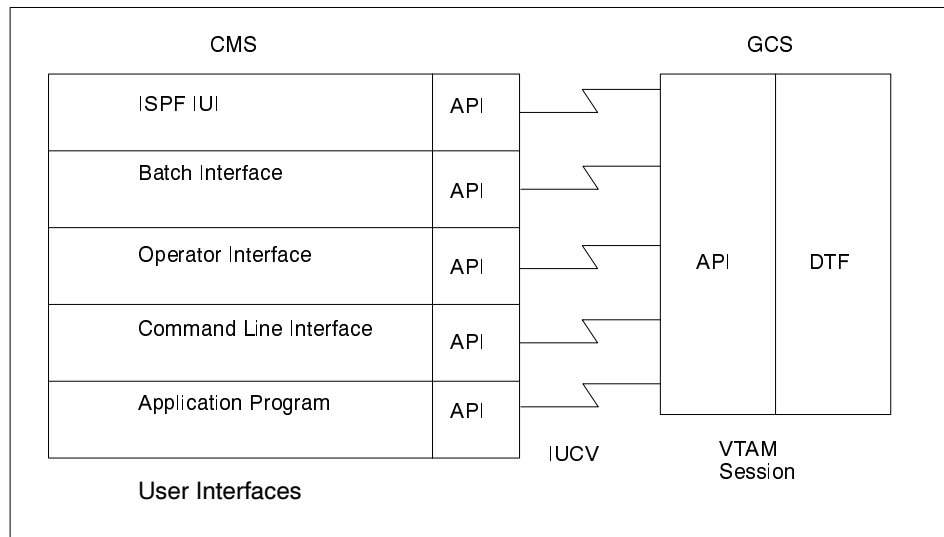
User Interfaces

There are five ways to communicate with the CONNECT:Direct API:

▶ ISPF Interactive User Interface (IUI)
▶ Batch Interface
▶ Operator Interface
▶ Command Line Interface (CLI)
▶ Application Program

The following sections describe each interface.

ISPF Interactive User Interface

The IUI is a screen and dialog component running under the Interactive System Productivity Facility (ISPF) in CMS.

The IUI builds CONNECT:Direct commands based on information provided on the panels and submits them to the API. The IUI and API validate the syntax of statements as they are created, issuing messages indicating acceptance or rejection of the request. CONNECT:Direct acts upon a request and then displays a message to the user.

Use the IUI to create and submit Processes, as well as perform the following tasks (based on the security level of the user):

▶ Initiate file transfers
▶ Schedule file transfers
▶ Establish defaults for COPY attributes
▶ Monitor and modify CONNECT:Direct task activity
▶ Concurrently signon to other CONNECT:Direct sessions
▶ Display statistics online or offline
▶ Display or print messages online
▶ Authorize new users
▶ Stop CONNECT:Direct
▶ Initiate traces
▶ Display, update, and insert Network Map entries
▶ Monitor and modify all CONNECT:Direct for VM/ESA Process activity (using commands such as SUBMIT, CHANGE, and SELECT)

See the *CONNECT:Direct for VM/ESA User's Guide* to learn how to use commands through the IUI.

Batch Interface

The Batch Interface allows you to issue CONNECT:Direct commands from a batch stream through the use of the DMBATCH program. Refer to the *Issuing Commands Through the Batch Interface* chapter of the *CONNECT:Direct for VM/ESA User's Guide* for more information on the batch environment.

Operator Interface

The Operator Interface, which executes as a subtask of the DTF, enables the console operator to submit commands through CLISTs for system administration and maintenance. Refer to the *CONNECT:Direct Console Operator's Guide* for a description of the Operator Interface.

Command Line Interface

The Command Line Interface (CLI) allows users to request CONNECT:Direct services from CMS. The command line interface runs under CMS and utilizes the DMCHLAPI interface to communicate with the DTF running under GCS. See the *Issuing Commands Through the Command Line Interface* chapter of the *CONNECT:Direct for VM/ESA User's Guide* for more information on this interface.

Application Programs

A high-level language Application Program Interface (API) allows user programs to interface directly to the CONNECT:Direct API. See the *Using the Program Interface* chapter in the *CONNECT:Direct for VM/ESA Administration Guide* for more information on the API.

Application Program Interface

The Application Program Interface (API) consists of the programs that allow the IUI, the Batch Interface, the Operator Interface, Command Line Interfaces, and application programs to communicate with the DTF or CONNECT:Direct Server. The API performs the following functions:

▶ Interprets commands from the various interfaces
▶ Validates the command format
▶ Passes the command across a VTAM session for DTF processing
▶ Receives the appropriate response or the requested data back from the DTF after the command is processed

Data Transmission Facility

The Data Transmission Facility (DTF) performs the following functions:

▶ Executes instructions, coded as Processes, passed to it from the API
▶ Starts and terminates all sessions
▶ Selects the next Process to execute
▶ Controls I/O requests for Shared File System (SFS) datasets
▶ Controls information distribution to other CONNECT:Direct nodes in the network

---

Note: Define start-up parameters that govern the overall activity of the DTF during CONNECT:Direct installation.

Transmission Control Queue

The Transmission Control Queue (TCQ) controls Process execution. CONNECT:Direct stores submitted Processes in the TCQ. The TCQ is divided into four logical queues: Wait, Execution, Hold, and Timer.

The Process is put in the appropriate queue based on Process statement parameters that affect scheduling. Examples of such parameters are the RETAIN and HOLD parameters and the STARTT parameter that indicates the day and time the Process should be executed.

CONNECT:Direct selects Processes in a first-in, first-out manner for execution within Process class and priority as sessions are available. You can access the queues and manipulate the Processes with commands entered through one of the user interfaces.

Chapter 2

Planning the Installation

This chapter provides information you need to plan your installation. Your installation is easier and more effective if you complete your planning before beginning the installation.

Before You Begin

Before you plan your installation, verify that you have the latest product information and that you have the proper resources available.

Read Your Release Notes

Before you begin the installation, read the *CONNECT:Direct for VM/ESA Release Notes* document for the latest product information including:

▶ Additional installation requirements
▶ Product enhancements
▶ Maintenance updates
▶ Documentation changes
▶ Product compatibility and connectivity
▶ Hardware and software requirements
▶ Online documentation instructions

You should be familiar with this document before you begin your installation. The information in this document may affect your installation procedures and definitions.

Complete the Worksheets

Complete the worksheets in the *Installation Worksheets* appendix of this manual. These worksheets help you to plan your installation.

Verify Hardware and Software Requirements

Refer to the *Product Requirements* section of the *CONNECT:Direct for VM/ESA Release Notes* document for complete information on the hardware and software requirements for CONNECT:Direct.

Determine the Virtual Storage Requirement

This section tells how to calculate your virtual storage requirements for:

▶ Data Transmission Facility (DTF)
▶ CONNECT:Direct Interactive User Interface (IUI)
▶ Shared File System (SFS) Server

Data Transmission Facility (DTF) Virtual Storage Requirement

This table describes the DTF virtual storage requirements.

Name	Space
CONNECT:Direct modules	2547.7K
CONNECT:Direct system storage	500.0K
Per interactive user	18.0K
Per node-to-node connection	12.0K
Per open VSAM KSDS	8.0K+2 x(CISIZE)
Per open VSAM ESDS or RRDS	6.0K+2x(CISIZE)
Per open SAM file	0.3K+BLKSIZE
VTAM requirement (approximate)	20.0K

The following table shows an example of the calculation of DTF storage requirements for 15 users and 20 node-to-node connections. In this

example, the 10 KSDS files, the 5 ESDS files, and the 5 SAM files make up the 20 node-to-node connections.

Name	Space
CONNECT:Direct modules	2547.7K
CONNECT:Direct system storage	500.0K
15 users (15 x 18 K)	270.0K
20 node-to-node connections (20 x 12 K)	240.0K
10 KSDS files (4K CI) (10 x 16 K)	160.0K
5 ESDS files (4 K CI) (5 x 14 K)	70.0K
5 SAM files (4K BLK) 5 x (0.3 K+4 K)	21.5K
VTAM requirement (approximate)	20.0K
Total	3,829.2K

IUI Virtual Storage Requirement

When you plan the IUI installation, you must first determine the virtual storage requirements for the IUI. The requirements for either the ISPF or Batch Interface are listed in the following table.

Name	Space
Required resident modules	40K
Dynamic virtual storage requirement	128K
Dynamic load area (largest module)	42K
Total per user	210K

In addition, IBM's ISPF product requires 500–1000K of virtual storage.

Shared File System Storage Requirement

When you plan to use the Shared File System (SFS) Server, you must first determine the virtual storage requirement for the server. Use the

following table to help you determine the SFS Server virtual storage requirements.

Name	Space
Required resident modules	384K
Dynamic virtual storage requirement	136K/concurrently SFS file

Prepare TCP/IP Configuration (Optional)

Read this section if you are using TCP/IP support. The type of support is indicated by the value of the TCP initialization parameter as follows:

▶ TCP=IBM specifies IBM TCP/IP support.

Note: Review the default settings of the TCP and NETMAP.CHECK initialization parameters to verify that it is consistent with your site requirements.

Refer to the *Initialization Parameters* appendix in the *CONNECT:Direct for VM/ESA Administration Guide* for a complete listing of the initialization parameters.

IBM TCP/IP Support

These procedures relate to the IBM implementation of TCP/IP only:

▶ Modify your *HLQ*.ETC.SERVICES data set to recognize CONNECT:Direct by adding the following information (*must be in lowercase*) to the end of the data set:

```
ndm-requestor    1363/tcp    #CONNECT:Direct api/iui
ndm-server       1364/tcp    #CONNECT:Direct dtf server
```

▶ If you are not using the default name for the IBM TCP/IP started task, you must code the name of the started task in the initialization parameter, TCP.NAME.

▶ The PROFILE EXEC on the TCP/IP service machine 191 disk contains system operation and configuration information for the TCP/IP address space. The PORT statement reserves a port for a given user ID and identifies the protocol to be used on that port.

> **Note**: Do not reserve a port number for CONNECT:Direct.

However, if you reserve a port number for the CONNECT:Direct server or requester, ensure that the userid specified on the PORT statement matches the userid of the CONNECT:Direct task. If PORT numbers are reserved and the userid does not match the userid of the CONNECT:Direct task, then the CONNECT:Direct server is unable to successfully bind a port to listen for incoming sessions. For additional information, refer to your *TCP/IP Planning and Customization* manual from IBM.

▶ CONNECT:Direct requires only the ETC files shipped with IBM's TCP/IP.

▶ To verify that a connection between nodes can be established, use the TCP/IP PING command. The PING command sends an echo request to a foreign host to determine if the computer is accessible. See the IBM TCP/IP documentation for information on the format of this command.

Planning for Security

CONNECT:Direct supports signon security checking through the CONNECT:Direct Authorization Facility. Use the Authorization Facility to control user access to CONNECT:Direct functions.

CONNECT:Direct security support includes, but is not limited to:

▶ Application programs through the Run Task exit
▶ Users through the CONNECT:Direct Authorization Facility
▶ CONNECT:Direct functions through the CONNECT:Direct Authorization Facility

Use the SECURITY.EXIT initialization parameter to specify a security exit. CONNECT:Direct invokes this exit during:

▶ Processing of the Signon command
▶ Process start
▶ File OPEN

When making a request to CONNECT:Direct for signon or file access, the request passes directly to the security module for authorization checking.

CONNECT:Direct provides sample security exits with the filetype ASSEMBLE on the distribution tape. You can customize these sample exits to fit a particular security need. See the *Controlling Security* chapter of the *CONNECT:Direct for VM/ESA Administration Guide* for more information about security exits.

Note: Customized security checking is an optional function. If SECURITY.EXIT is not specified, the CONNECT:Direct Authorization Facility performs authorization checking.

Planning for Parallel Sessions and Process Recovery

CONNECT:Direct provides facilities to recover from most errors that occur during Process execution. Recovery from the point of failure is usually accomplished quickly.

During the installation, you must establish values for various parameters which effect Process recovery and checkpoint restart. Refer to the *CONNECT:Direct for VM/ESA Administration Guide* for more information about process recovery and checkpoint restart.

Understanding Parallel Sessions

CONNECT:Direct uses the parallel sessions capability of VTAM to allow multiple Processes to execute at the same time between any two CONNECT:Direct nodes. To use parallel sessions support, you must specify the PARSESS=YES initialization parameter in the VTAM application definition for both nodes.

Note: If two nodes have different values for parallel sessions, transfers are limited by the maximum number of sessions in the ADJACENT.NODE definition on the node where the Process was submitted.

The maximum number of sessions between two nodes is defined in the Network Map. Because each session has a corresponding class value, the maximum number of sessions and the maximum number of classes are equal.

Selection of a Process for execution in a given node is based on Process priority (PRTY= on the PROCESS statement) within session class. User-specified class values allow a Process to execute on the session

having a matching or higher class values. Specify the default class in the CONNECT:Direct Network Map.

When one Process completes, CONNECT:Direct selects another Process to run on the available session. CONNECT:Direct searches until it finds the first Process with a class eligible to execute on the available session.

Define critical Processes with low class values so that more sessions are available for their execution. Use higher class numbers to specify time-consuming Processes. This allows sessions with corresponding lower class numbers to become available more frequently. An example of selection by class follows.

Example of Parallel Processes

CONNECT:Direct has just been brought up. Seven Processes have been submitted for NODE.B and are ready to run. All Processes have a user-specified class value and the same priority. Class determines which session CONNECT:Direct selects.

The Network Map has been defined so a maximum of four sessions can be started between NODE.A and NODE.B. Each session between NODE.A and NODE.B has its own corresponding unique class number.

1. In the following figure, NODE.A simultaneously starts four sessions. Processes are shown in the order they appear in the queue. Note that because PROCB's user-specified class is 1, it can run on the class 2, 3, or 4 sessions if needed.

TCQ (NODE.A)		
Search Order	Process Name	User-Specified Class
	PROCA†	1
	PROCB†	1
	PROCC†	4
	PROCD†	3
	PROCE	3
	PROCF	4
	PROCG	1

NODE-TO-NODE SESSIONS

Process Running — Session Class

NODE.A	PROCA	1	NODE.B
	PROCB	2	
	PROCD	3	
	PROCC	4	

† Indicates an executing Process
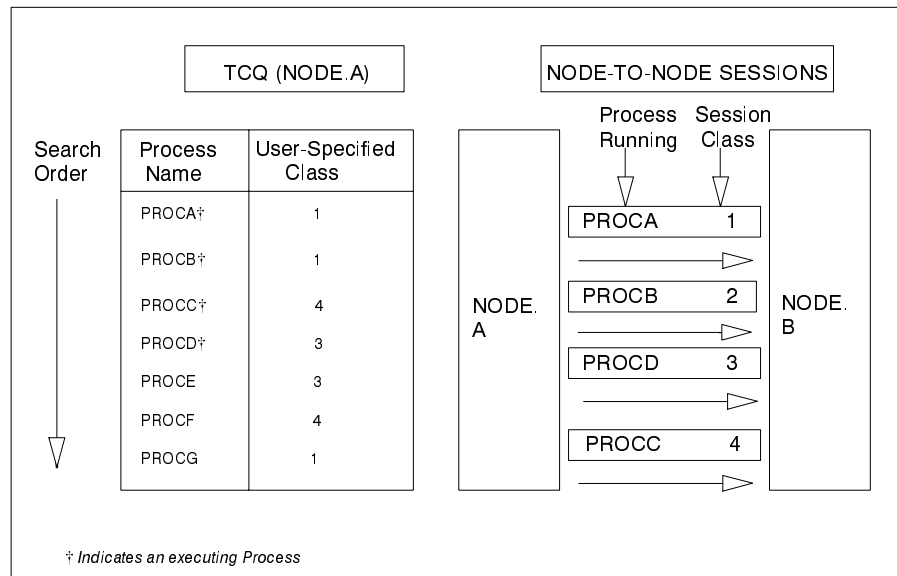
2. In the following figure, PROCB has completed execution, making a session available. CONNECT:Direct looks through the TCQ for the first eligible Process for that session. PROCG is the next Process available to run on the class 2 session because all other Processes have a class value higher than 2.
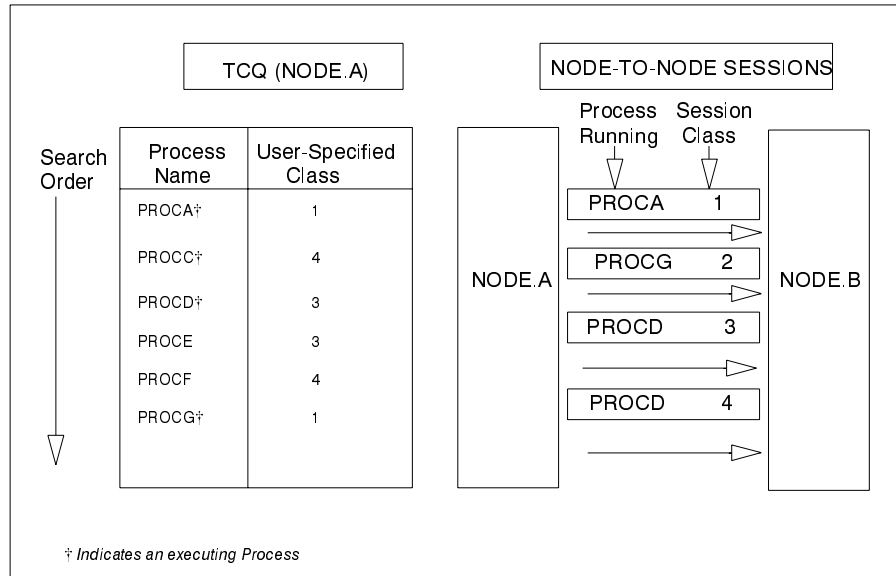
3. In this figure, PROCA, PROCG, and PROCD have completed execution. Sessions for class 1 and 2 cannot be used at this time because only a Process for class 4 is left in the queue.
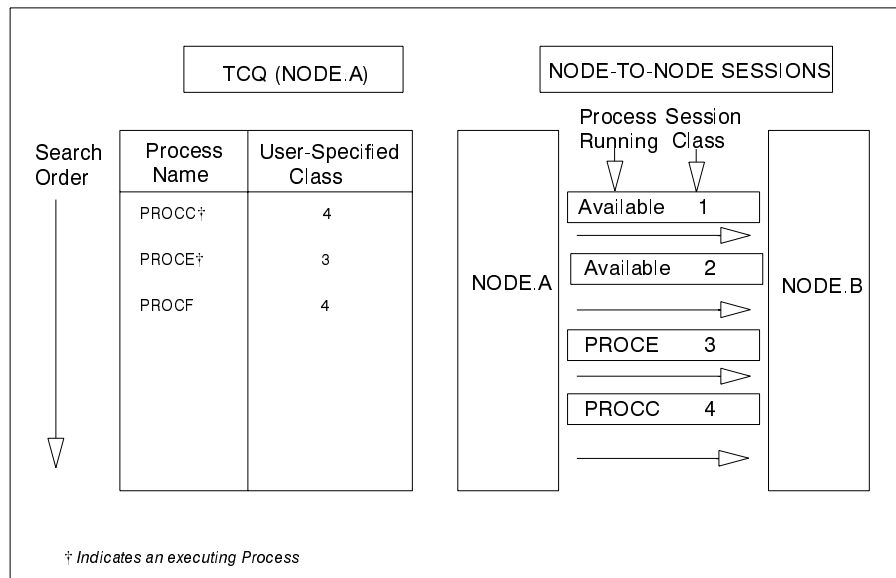
4. In this figure, PROCH and PROCI are submitted.

TCQ (NODE.A)

Search Order	Process Name	User-Specified Class
	PROCC†	4
	PROCE†	3
	PROCF	4
	PROCH†	1
	PROCI†	1

NODE-TO-NODE SESSIONS

Process Session
Running Class

NODE.A

	Process	Session Class
	PROCH	1
	PROCI	2
	PROCE	3
	PROCC	4

NODE.B

† Indicates an executing Process
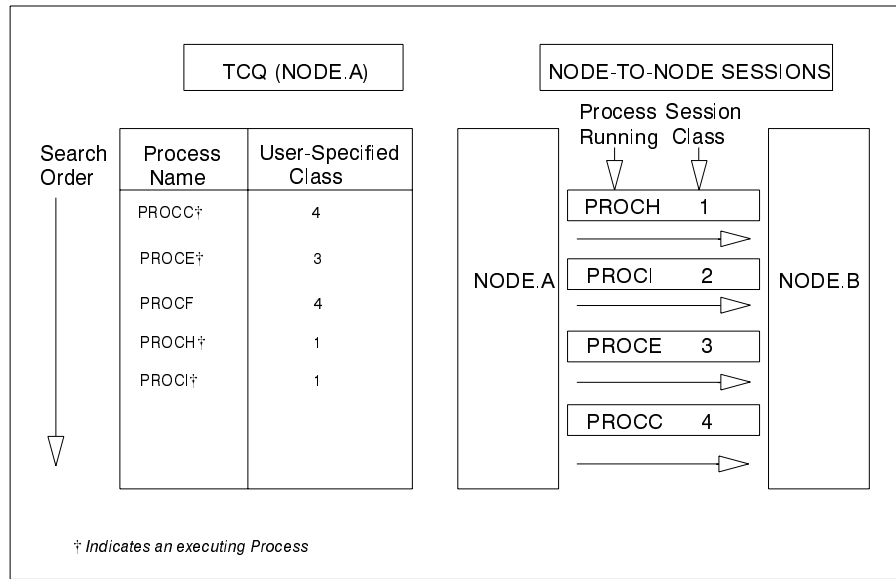
5. In this figure, PROCE completes. PROCF will not execute until
    PROCC has completed because PROCF can run only on the class 4
    session. If at this point any other Process is submitted for class 1, 2,
    or 3 it can use a class 3 session.

TCQ (NODE.A)

Search Order	Process Name	User-Specified Class
	PROCC†	4
	PROCF	4
	PROCH†	1
	PROCI†	1

NODE-TO-NODE SESSIONS

Process Session
Running Class

NODE.A

	Process	Session Class
	PROCH	1
	PROCI	2
	Available	3
	PROCC	4

NODE.B

† Indicates an executing Process
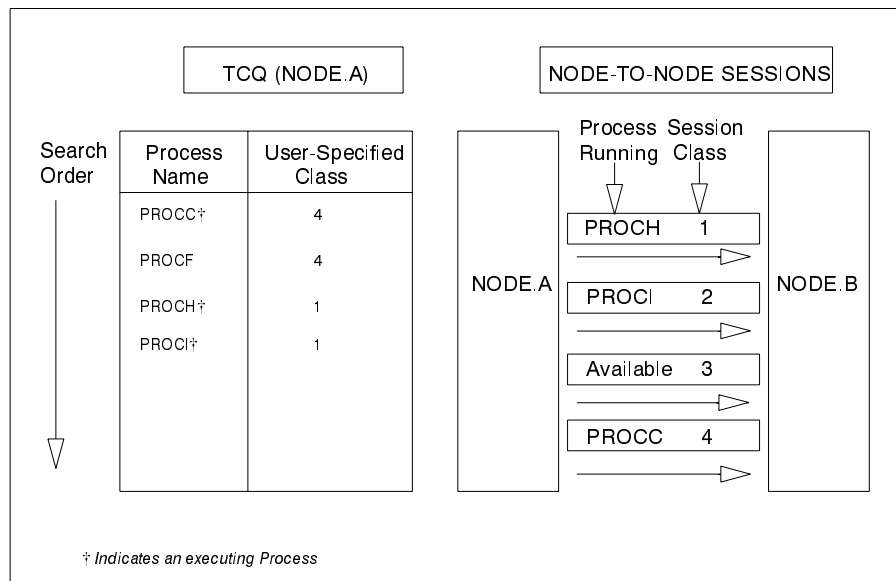
Planning the Network Map

The Network Map identifies the local CONNECT:Direct node and the
nodes with which it can communicate. It consists of a *local node* entry
and one or more *adjacent node* entries. Each entry identifies the

communications name and protocol associated with a CONNECT:Direct node. The source form of the Network Map is generated during the installation process and is input to the Network Map load utility, DMCNTMPL. This utility creates the VSAM form of the Network Map to be used by CONNECT:Direct.

To prepare for the installation, fill in the Network Map worksheets. Review the *Maintaining the Network Map* chapter of the *CONNECT:Direct for VM/ESA Administration Guide* for a description of the content of the Network Map and for sample entries.

Planning the DASD Requirements

This chapter discusses DASD requirements for CONNECT:Direct for VM/ESA distribution libraries and VSAM files and the virtual storage requirements for the Data Transmission Facility (DTF) and the optional ISPF Interactive User Interface (IUI).

Distribution Libraries Description

The following table describes the CONNECT:Direct for VM/ESA distribution files. The files are grouped according to file type.

Distribution File	Description
AMSERV	Installs AMSERV skeletons for CONNECT:Direct
ASSEMBLE	Interfaces with various CONNECT:Direct utilities, sample programs, and exits
EXEC	Interfaces with various CONNECT:Direct utilities, sample programs, and exits
GLOBALV	The CONNECT:Direct installation file
GCS	Use GCS EXEC to define CONNECT:Direct for VM/ESA programs that will execute as commands
INITPARM	A file containing a sample set of CONNECT:Direct initialization parameters
INPUT	The input for CONNECT:Direct batch utility
ISREDIT	The ISPF edit macros

(continued)

Distribution File	Description
LOADLIB	The CONNECT:Direct for VM/ESA modules
MACLIB	The CONNECT:Direct source macros, the CONNECT:Direct ISPF message library, and the CONNECT:Direct ISPF panel library
MODULE	The CONNECT:Direct CMS Command Line Interface program and CONNECT:Direct sample programs
NDMMSG	The input for the CONNECT:Direct message load program
NDMOP	The CONNECT:Direct command lists (CLISTs) for the Operator Interface
NDMPANEL	A CONNECT:Direct ISPF Submit Process panel
NDMPROC	A CONNECT:Direct Process and CONNECT:Direct statement model
NDMVTAM	Displays examples of various VTAM definitions
XEDIT	Runs CONNECT:Direct installation XEDIT macros

Determine VSAM Files DASD Requirements

CONNECT:Direct uses VSAM files to control and monitor execution. The following table describes the VSAM files and their space requirements. The installation procedure creates the files using these minimum space requirement values for each VSAM file.

Note: Use the values in the *Minimum Primary Entries* column for testing purposes. Use the values in the *Number of Entries* column for production purposes.

Data Set	DSORG	Approx. Install Supplied Size	Number of Entries	Approx. Minimum Size	Minimum Primary Entries	BLK SIZE (CI)
MSG	KSDS	3 MB	9000	3 MB	9000	4096
AUTH	KSDS	132 KB	100	88 KB	5	4096
STATS log1	ESDS	1.8 MB	6750	197 KB	1000	4096

(continued)

Data Set	DSORG	Approx. Install Supplied Size	Number of Entries	Approx. Minimum Size	Minimum Primary Entries	BLK SIZE (CI)
STATS index1	KSDS	222 KB	5000	88 KB	750	4096
STATS log2††	ESDS	1.8 MB	6750	197 KB	1000	4096
STATS index2††	KSDS	222 KB	5000	88 KB	750	4096
STATS directory†	KSDS	88 KB	500	88 KB	500	4096
TCQ	RRDS	200 KB	100	44 KB	21	1536
TCX	RRDS	44 KB	1	44 KB	1	1024
TYPE	KSDS	88 KB	20	88 KB	5	4096
CKPT	KSDS	600 KB	40	44 KB	5	4096
NETMAP	KSDS	88 KB	50	88 KB	2	4096
LU1 Script †	KSDS	44 KB	5	44 KB	1	8192
TOTAL		8 MB (approx.)		4 MB (approx.)		

†   optional

††   The default configuration for the Statistics facility uses two statistics file pairs, which requires four VSAM files. Refer to the *Statistics Files* section on page 3-4 for details about the VSAM files.

Description of VSAM Files

The following sections describe each of the VSAM files listed in the previous table.

CONNECT:Direct Message File (MSG)
> The Message file, a VSAM key-sequenced data set (KSDS), holds all messages, except panel-related messages, used by CONNECT:Direct. Each message record contains the issuing module name, short message text, and message explanation.

Authorization File (AUTH)
> The Authorization file, a VSAM KSDS, can contain a record for each authorized CONNECT:Direct user. If you are using the CONNECT:Direct Authorization Facility,

use the following formula to calculate the size of your
Authorization File requirements.

```
Number of CONNECT:Direct Users X 100 Bytes = Size of Authorization File
```

If you are *not* using the CONNECT:Direct Authorization
Facility, you still must define an Authorization file. Use
the value specified in the *Minimum Primary Entries*
column of the table on page 3-2.

Statistics Files

The CONNECT:Direct Statistics facility logs
CONNECT:Direct statistics to VSAM file pairs. The
default and minimum configuration uses two such file
pairs, making four VSAM files. The maximum number
of file pairs that can be used is twenty.

Specify the number of file pairs to be used and the VSAM
cluster names with the STAT.DSN.BASE and
STAT.FILE.PAIRS initialization parameters. Each file pair
consists of the following:

▶   A VSAM entry sequenced cluster
▶   A VSAM key sequenced cluster

Refer to the *Administering Statistics* chapter of the
CONNECT:Direct for VM/ESA Administration Guide for a
detailed discussion of the format and use of the file pairs.
Refer to the *Tuning CONNECT:Direct for VM/ESA for
Performance* chapter beginning on page 7-1 to learn how
to tune the statistics file pairs.

Statistics Directory of Archive Files

Allocate the STATS Archive Directory file if you plan to
archive statistics records and maintain a directory of the
archive files. Archiving is the process of copying
statistics records from the CONNECT:Direct statistics
files to other data sets for long-term storage. Refer to the
Archiving Statistics section of the *Administering Statistics*
chapter in the *CONNECT:Direct for VM/ESA
Administration Guide* for details about archiving statistics.

Use the directory to track the data set names of the
archive files and the date/time range of the statistics
records in the archive files. Each record in the directory
contains information about a single archive file. The
value of the RECORDS parameter that defines the
directory determines how many archive files can be

represented in the directory. Use the INQUIR STATDIR command to maintain and display the directory.

Transmission Control Queue (TCQ) and Index (TCX)

The Transmission Control Queue (TCQ) is a VSAM RRDS file that CONNECT:Direct uses to store Processes that are executing, queued for execution, held for retries, or held for future execution. The TCQ index (TCX) is also a VSAM RRDS file. It consists of a record that is a map for controlling the allocation of space in the TCQ.

CONNECT:Direct accesses the TCQ in CI mode. The CI size of the TCQ must be 1,536 bytes. Each Process that is defined in the TCQ occupies one or more CIs, depending on the number of statements within the Process. Because TCQ access characteristics prohibit secondary allocation, allow enough room for the maximum number of potential Processes when defining the TCQ.

The TCX controls space use and access to the TCQ. CONNECT:Direct uses the maximum record length of the TCX to determine the number of CIs that can be controlled in the TCQ, thus dictating the size of the TCQ.

The following formula determines the number of CIs that can be controlled:

Number of CIs in TCQ=((MAXLRECL of TCX – 12) / 2) X 8

The following is an example of determining the number of CIs that can be controlled.

TCX is defined with CISIZE=1,024

MAXLRECL of TCX: 1,024 – 7=1,017 (7 bytes of VSAM control information)

Therefore:

Number of CIs controlled=((1,017 – 12) /2) x 8=4,020

You can define the TCQ with up to 4,020 records, thereby allowing it to hold 1,000 to 4,000 Processes, depending on the number of statements in each. CONNECT:Direct does not use more than 4,020 records. You can define fewer records, based on your needs. *Do not modify the CI size of the TCQ or TCX.*

Note: Once you allocate the TCQ and TCX, they are immovable. If you must redefine the TCQ and TCX, you must also cold-start the CONNECT:Direct DTF to reformat these files.

Type File (TYPE)

The Type file is a VSAM KSDS that consists of records containing file attribute defaults for the source and destination file allocations specified in the CONNECT:Direct COPY statement.

For more information on the Type file contents, see the *Maintaining the Type File* chapter in *CONNECT:Direct for VM/ESA Administration Guide*.

Checkpoint File (CKPT)

The Checkpoint file is a VSAM KSDS that contains checkpoint information generated by the DTF during execution of a COPY statement. The Checkpoint file is made up of variable length records; one per Process that has checkpointing specified. This includes previous copies that have been checkpointed but have not completed yet or are not currently in execution. The average record length is 256 bytes. The equivalent of one cylinder of space is allocated during the installation.

The size of the checkpoint file is influenced by the number of days you retain checkpoint information according to CKPT.DAYS in your initialization parameters.

Network Map File (NETMAP)

The Network Map is a VSAM KSDS that contains network definition information used by CONNECT:Direct. The information includes the following:

▶ Network names for the local node
▶ Network names for the adjacent nodes in the network
▶ Control information used by CONNECT:Direct
▶ VTAM application identifiers (applids) for use by the CONNECT:Direct Application Program Interface (API)

Note: If you are using a VSAM cache program, you must remove the CONNECT:Direct network map from its control. Otherwise, it will rewrite the date-time stamp, and signons will fail with SAFA013I.

To estimate the size of the network map, calculate the number of nodes in your network, including the local node, and the number of applids defined for each node.

For example, if a network map contains 25 nodes with 20 API applids in each of the 25 nodes, it requires approximately the space shown in the following table.

Name	Space
Network Map Header	176 bytes (1 record)
CONNECT:Direct Control	52 bytes (1 record)

Node records=
    25 nodes x 88 bytes per
    node record
	2,200 bytes (25 records)

API Applids=
    25 nodes x
    (32–byte control record
    size+(8–byte maximum
    length applid name x
    20 API applid records per
    ode))
	4,800 bytes (25 records)

Total	7,228 bytes (52 records)

Define the Network Map with sufficient space to allow for this amount of data. The amounts are approximate as space requirements may vary.

LU1 Script File

CONNECT:Direct stores information in this VSAM KSDS LU1 Script file to allow it to contact a CONNECT:Direct for MS-DOS node across an X.25 network. To use an LU1.SCRIPT data set refer to the *CONNECT:Direct for MS-DOS LU1/Async Interface Installation Guide*. Define this file only if you are using the LU1 Script Facility.

CONNECT:Direct for VM/ESA Installation Guide

Chapter 4

Preparing VTAM Definitions

Before you bring up CONNECT:Direct, the VTAM application definitions must be active, the mode table entries must be completed, and the cross-domain resources must be active. VTAM definitions must be provided for CONNECT:Direct. These VTAM definitions include:

▶ API applids
▶ DTF interface applids
▶ a single mode table entry
▶ cross-domain definitions

This chapter explains how to set up VTAM definitions for CONNECT:Direct.

Description of VTAM Definitions

The sample VTAM definitions for CONNECT:Direct shown in this section are on the distribution tape with the filetype NDMVTAM. This table lists the VTAM definitions required for each node location and the file name of each example.

Sample Member	VTAM-related Definitions
NDMAPPL	VTAM definitions for CONNECT:Direct DTF
NDMIAPPL	VTAM definitions for IUI or Batch Interface

(continued)

Sample Member	VTAM-related Definitions
NDMMAP01	Network Map definition
NDMMODET	Mode Table used with CONNECT:Direct
NDMCDRM	Cross-domain resource manager node
NDMCDRSC	Cross-domain resource definition for other nodes

Refer to the *VTAM Definitions*, appendix for additional sample VTAM APPLID and logmode table definitions.

Sample Network

This CONNECT:Direct network example corresponds to the VTAM definitions presented on the following pages.

```
        CD.NODE.A                              CD.NODE.B

   LOCAL.NODE=CD.NODE.A                 LOCAL.NODE=CD.NODE.B
   DTF APPLID=NDMDAPP1                  DTF APPLID=NDMDAPP2
   SUBAREA 20                          SUBAREA 30
   API APPLIDS=(NAN01,NAN02)            API APPLIDS=(NBN01,NBN02)


   ADJACENT.NODES      APPLIDS          ADJACENT.NODES      APPLIDS
    CD.NODE.B  NDMDAPP2                   CD.NODE.A  NDMDAPP1
```
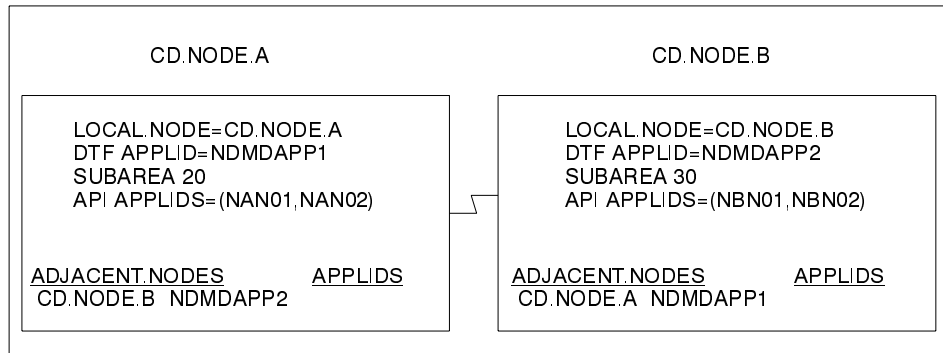
Define APPLID of Local DTF

Define the APPLID of the local CONNECT:Direct DTF. Two example definitions for the APPLID of the local DTF follow. The examples are located in the member NDMAPPL of *$CD*.CNTL.

VTAM APPL for DTF

The following figure shows a VTAM application definition for the CONNECT:Direct DTF.

```
NDMAPP1 APPL   ACBNAME=NDMAPP1, VTAM APPLICATION ID        X
               AUTH=(ACQ,NOCNM,NOPASS,NOPO,NOTSO,VPACE),    X
               EAS=N,           APPROXIMATE # CONCURRENT SESS X
               MODETAB=NDMTAB,  MODE TABLE NAME             X
               SONSCIP=NO,      NO UNBIND IN SCIP EXIT      X
               SRBEXIT=NO,      NO SRB PROCESSING           X
               VPACING=N,       RECEIVE PACING              X
               DLOGMOD=NDMLOGM, MODE TABLE ENTRY            X
               PARSESS=YES,     PARALLEL SESSIONS CAN BE USED X
               VTAMFRR=NO
```

Sterling Commerce recommends the following operand settings to ensure proper session startup and shutdown:

▶ The PARSESS=YES parameter is required.

▶ EAS represents the approximate number of concurrent sessions that this APPL will have with all other APPLs (CONNECT:Direct adjacent nodes).

▶ DLOGMOD (the default log mode) must refer to an LU0 logmode entry.

▶ MODETAB must point to the mode table that contains entries used with each APPLID. See page 4-7 for the definition of the mode table.

▶ Set VPACING to the number of RUs that this DTF will receive before being obligated to send a pacing response. For best results, use a minimum value of **7**. Larger values will increase throughput when receiving data from another DTF.

Note: Do not specify a value of **0**, or omit this parameter, because this causes no pacing, which could lead to a VTAM buffer shortage.

VTAM APPL for DTF for LU6.2

The following figure shows a VTAM application definition for the CONNECT:Direct DTF for LU6.2.

```
NDMAPP4 APPL   ACBNAME=NDMAPP4, VTAM APPLICATION ID              X
               APPC=YES,        ENABLE TO RUN LU6.2 SESSIONS     X
               AUTH=(ACQ,NOCNM,NOPASS,NOPO,VPACE),               X
               EAS=N,           APPROXIMATE # OF CONCURRENT SESS  X
               MODETAB=NDMTAB,  MODE TABLE NAME                  X
               SONSCIP=NO,      NO UNBIND IN SCIP EXIT           X
               SRBEXIT=NO,      NO SRB PROCESSING                X
               VPACING=N,       RECEIVE PACING                   X
               DLOGMOD=NDMLOGM, MODE TABLE ENTRY                 X
               PARSESS=YES,     PARALLEL SESSIONS CAN BE USED    X
               DSESLIM=N,       # OF CONCURRENT LU6.2 SESSIONS   X
               DMINWNL=N,       # OF LOCAL CONTENTION WINNERS    X
               DMINWNR=N,       # OF REMOTE CONTENTION WINNERS   X
               AUTOSES=N,       # OF AUTOMATIC LU6.2 SESSIONS    X
               DDRAINL=ALLOW,   ALLOW CNOS TO DRAIN SESSIONS     X
               DRESPL=ALLOW,    DEF RESPONSIBILITY FOR LOCAL CNOS X
               LMDENT=N,        SIZE OF HASH TABLE               X
               VTAMFRR=NO
```

Sterling Commerce recommends the following settings:

▶ APPC=YES allows LU6.2 support and is required.

▶ Set AUTOSES to **1** if this DTF will be used in a mixed environment of sending and receiving files. Set AUTOSES to **0** if this DTF will be used primarily to receive files.

> **Note**:   You can still use this DTF to send files if you have set AUTOSES to **0**.

▶ DDRAINL=ALLOW allows Change Number of Sessions (CNOS) to drain sessions.

▶ Set DMINWNL to a value that is no larger than one-half the value of DSESLIM.

▶ Set DMINWNR to a value that is no larger than one-half the value of DSESLIM.

▶ DRESPL=ALLOW allows the application program to accept responsibility for deactivating sessions.

▶ Set DSESLIM to the sum of the values specified in DMINWNL and DMINWNR. That is, DSESLIM should be equal to or greater than the largest PARSESS max value in the Network Map.

▶ DLOGMOD (the default log mode) must refer to an LU0 logmode entry to ensure compatibility with prior releases of

CONNECT:Direct. The LU6.2 logmode name will be extracted from the Network Map.

▶ EAS represents the approximate number of concurrent sessions that this APPL will have with all other APPLs (CONNECT:Direct adjacent nodes and interactive users). The default value is 509. Change the value to one that is realistic for your environment.

▶ LMDENT specifies the number of entries to be used for this application program's hash table of remote LUs. The default value is 19.

▶ MODETAB must point to the mode table that contains entries for LU0, LU6.2, and SNA Service Manager mode (SNASVCMG). See page 4-7 for the definition of the mode table.

▶ Set VPACING to the number of RUs that this DTF will receive before being obligated to send a pacing response. For best results, use a minimum value of **7**. Larger values will increase throughput when receiving data from another DTF.

Note: Do not specify a value of **0**, or omit this parameter, because this causes no pacing, which could lead to a VTAM buffer shortage.

Define APPLID for IUI and Batch Sessions

Define APPLID for IUI and batch session definition.

The following example shows three VTAM application definitions for the CONNECT:Direct IUI. These application IDs must match those specified in the CONNECT:Direct Network Map. The examples are located in member NDMIAPPL of *$CD*.CNTL.

```
NAI01    APPL  ACBNAME=NAI01,                                   x
               DLOGMOD=NDMLOGM,                                 x
               MODETAB=NDMTAB

NAI02    APPL  ACBNAME=NAI02,                                   x
               DLOGMOD=NDMLOGM,                                 x
               MODETAB=NDMTAB

NAI03    APPL  ACBNAME=NAI03,                                   x
               DLOGMOD=NDMLOGM,                                 x
               MODETAB=NDMTAB
```

Define APPLID for Loop-Back Processing

If you perform PNODE=SNODE (loop-back) processing, define the APPLID to be used for the processing. See page 2-9 for a discussion of the Network Map requirements.

The following figure shows an additional CONNECT:Direct DTF APPL (NDMAPP2) defined to VTAM that allows you to perform loop-back processing. This multiple definition is required because with PNODE=SNODE processing the communications name (APPLID) for the adjacent node defined in the Network Map must be different than the communications name (APPLID) for the local node in the Network Map. This example is located in member NDMAPPL of *$CD*.CNTL.

```
NDMAPP2  APPL  ACBNAME=NDMAPP2,     VTAM APPLICATION ID            X
               AUTH=(ACQ,NOCNM,NOPASS,NOPO,VPACE),                 X
               EAS=N,               APPROXIMATE # CONCURRENT SESS   X
               MODETAB=NDMTAB,      MODE TABLE NAME                X
               SONSCIP=NO,          NO UNBIND IN SCIP EXIT         X
               SRBEXIT=NO,          NO SRB PROCESSING              X
               VPACING=7,           RECEIVE PACING OF 7            X
               DLOGMOD=NDMLOGM,     MODE TABLE ENTRY               X
               PARSESS=YES,         PARALLEL SESSIONS CAN BE USED  X
               VTAMFRR=NO
```

See the *Maintaining the Network Map* chapter of the *CONNECT:Direct for VM/ESA Administration Guide* for examples and content of the CONNECT:Direct Network Map.

Define APPLID for MS-DOS

If a CONNECT:Direct for VM/ESA node sends and receives data to a CONNECT:Direct for MS-DOS node, you need to define a unique APPLID with AUTH=SPO through which CONNECT:Direct can terminate the connection to the MS-DOS node if it is unable to establish a session after a certain length of time.

The following figure is an example of such an APPL definition.

```
NDMPOA01       APPL    ACBNAME=NDMPOA01,                          X
               AUTH=(ACQ,NOCNM,NOPASS,SPO),                       X
               EAS=10,                                            X
               MODETAB=NDMTAB,                                    X
               SONSCIP=NO,                                        X
               SRBEXIT=NO,                                        X
               VPACING=7,                                         X
               DLOGMOD=NDMLOGM,                                   X
               PARSESS=YES,                                       X
               VTAMFRR=NO
```

The initialization parameters POA.APPL and CONNECT.WAIT define the APPLID and establish the length of time CONNECT:Direct waits before terminating the session initiation request.

Note: If you define the APPLID with AUTH=SPO, any LU0 or LU6.2 logmode can be affected.

Define Logmode Table Entries

Define the logmode table entries used with each APPLID. This step is required if you will communicate with a node defined to VTAM as a PU TYPE 4 node.

The following sections provide examples from the mode table used with CONNECT:Direct. The examples are located in member NDMMODET.

The optional name (NDMTAB) can be used as an assembler CSECT name for the mode table. The table can be assembled separately from other mode tables or the CONNECT:Direct entry can be inserted into an existing mode table.

Note: Users should modify COS, PACING, and RUSIZE values *only*.

MVS Nodes

Use the following entry with other CONNECT:Direct for MVS nodes.

```
NDMLOGM  MODEENT LOGMODE=NDMLOGM, ENTRY NAME                          X
                 TYPE=1,           NON-NEGOTIABLE BIND                X
                 FMPROF=X'04',     FUNCTION MGMT. PROFILE 4           X
                 TSPROF=X'04',     TRANS SERVICE  PROFILE 4           X
                 PRIPROT=X'B3',    PRIMARY PROTOCOL                   X
                 SECPROT=X'B3',    SECONDARY PROTOCOL                 X
                 COMPROT=X'6080',  COMMON PROTOCOL                    X
                 PSNDPAC=X'06',    PRIMARY SEND PACING                X
                 SRCVPAC=X'06',    SECONDARY RECEIVE PACING           X
                 SSNDPAC=X'06',    SECONDARY SEND PACING              X
                 RUSIZES=X'8989',  4K MAX RU SIZE FOR PRI AND SEC     X
                 PSERVIC=X'000000000000000000000000'
```

MS-DOS/IRMA

Use the following entry with MS-DOS/IRMA nodes.

```
D4A32782 MODEENT LOGMODE=D4A32782, ENTRY NAME                      X
                  FMPROF=X'03',      FUNCTION MGMT. PROFILE 3       X
                  TSPROF=X'03',      TRANS SERVICE  PROFILE 3       X
                  PRIPROT=X'B1',     PRIMARY PROTOCOL               X
                  SECPROT=X'90',     SECONDARY PROTOCOL             X
                  COMPROT=X'3080',   COMMON PROTOCOL                X
                  RUSIZES=X'87C7',   1K SEND RU, 1.5K RECEIVE RU    X
                  PSERVIC=X'02000000000000185000007E00'
```

MS-DOS Nodes with Structured Field Support

Use the following entry with MS-DOS nodes with structured field support.

```
NDMPCSF  MODEENT LOGMODE=NDMPCSF, ENTRY NAME                       X
                 FMPROF=X'03',      FUNCTION MGMT. PROFILE 3        X
                 TSPROF=X'03',      TRANS SERVICE  PROFILE 3        X
                 PRIPROT=X'B1',     PRIMARY PROTOCOL                X
                 SECPROT=X'90',     SECONDARY PROTOCOL              X
                 COMPROT=X'3080',   COMMON PROTOCOL                 X
                 RUSIZES=X'87F8',   1K SEND RU, 3840 RECEIVE RU     X
                 PSERVIC=X'02800000000000185000007E00'
```

MS-DOS COAX Attached to a Non-SNA Controller

Use the following entry with MS-DOS COAX nodes attached to a non-SNA local controller.

```
NDMPCLC  MODEENT LOGMODE=NDMPCLC, ENTRY NAME                       X
                 FMPROF=X'02',      FUNCTION MGMT. PROFILE 3        X
                 TSPROF=X'02',      TRANS SERVICE  PROFILE 3        X
                 PRIPROT=X'71',     PRIMARY PROTOCOL                X
                 SECPROT=X'40',     SECONDARY PROTOCOL              X
                 COMPROT=X'2000',   COMMON PROTOCOL                 X
                 RUSIZES=X'87C7',   1K SEND RU, 1.5K RECEIVE RU     X
                 PSERVIC=X'02000000000000185000007E00'
```

OpenVMS and Tandem Nodes

Use the following entry with OpenVMS and Tandem nodes.

```
NDMVMS   MODEENT LOGMODE=NDMVMS,                                       X
                 TYPE=1,                                               X
                 FMPROF=X'04',                                         X
                 TSPROF=X'04',                                         X
                 PRIPROT=X'B1',                                        X
                 SECPROT=X'B1',                                        X
                 COMPROT=X'7080',                                      X
                 SRCVPAC=X'04',      SRCVPAC,SSNDPAC, AND PSNDPAC      X
                 SSNDPAC=X'04',      VALUES CAN BE TUNED               X
                 PSNDPAC=X'04',                                        X
                 RUSIZES=X'8989',    4K SEND AND RECV RUSIZE           X
                 PSERVIC=X'0000000000000000000000000'
```

OS/400 SNUF (LU0) Nodes

Use the following entry with OS/400 SNUF (LU0) nodes.

```
SNUF4K   MODEENT LOGMODE=SNUF2K,                                       X
                 COS=NJE,                                              X
                 FMPROF=X'04',                                         X
                 TSPROF=X'04',                                         X
                 PRIPROT=X'B1',                                        X
                 SECPROT=X'B1',                                        X
                 COMPROT=X'7080',                                      X
                 SRCVPAC=X'07',      SRCVPAC,SSNDPAC, AND PSNDPAC      X
                 SSNDPAC=X'07',      VALUES CAN BE TUNED               X
                 PSNDPAC=X'07',                                        X
                 RUSIZES=X'8888',    2K SEND AND RECV RUSIZE           X
                 PSERVIC=X'0000000000000000000000000'
```

LU6.2 Nodes

Use the following entry with LU6.2 independent Nodes.

```
NDM624K  MODEENT LOGMODE=NDM624K,                                      X
                 TYPE=1,                                               X
                 COS=NJE,                                              X
                 FMPROF=X'13',                                         X
                 TSPROF=X'07',                                         X
                 PRIPROT=X'B0',                                        X
                 SECPROT=X'B0',                                        X
                 COMPROT=X'D0B1',                                      X
                 PSNDPAC=X'04',                                        X
                 SRVCPAC=X'04',                                        X
                 SSNDPAC=X'04',                                        X
                 RUSIZES=X'8989',    4K SEND AND RECV RUSIZE           X
                 PSERVIC=X'0602000000000000000000300'
```

Use the following entry with LU6.2 dependent Nodes.

```
CD624K MODEENT LOGMODE=CD624K,                                    X
            TYPE=1,                                              X
            COS=NJE,                                            X
            FMPROF=X'13',                                        X
            TSPROF=X'07',                                        X
            PRIPROT=X'B0',                                       X
            SECPROT=X'B0',                                       X
            COMPROT=X'50B1',                                     X
            RUSIZES=X'8989',                                     X
            PSERVIC=X'0602000000000000000002C00'
```

MS-DOS LU0

Use the following entry for MS-DOS LU0.

```
NDMPCLU0 MODEENT LOGMODE=NDMPCLU0,TYPE=1,                         X
              COS=NJE,                                           X
              FMPROF=X'04',                                      X
              TSPROF=X'04',                                      X
              PRIPROT=X'B1',                                     X
              SECPROT=X'B1',                                     X
              COMPROT=X'7080',                                   X
              RUSIZES=X'8686',                                   X
              SSNDPAC=X'07',                                     X
              SRCVPAC=X'07',                                     X
              PSERVIC=X'000000000000000000000000'
```

LU6.2 SNA Services Manager

LU6.2 requires the following SNA Services Manager mode table entry definition. This LOGMODE is for VTAM use. You must make this definition in the Mode Table, but do not specify it as the LOGMODE in the Network Map.

```
SNASVCMG MODEENT LOGMODE=SNASVCMG,                                X
             TYPE=1,                                             X
             COS=NJE,                                           X
             FMPROF=X'13',                                       X
             TSPROF=X'07',                                       X
             PRIPROT=X'B0',                                      X
             SECPROT=X'B0',                                      X
             COMPROT=X'D0B1',                                    X
             RUSIZES=X'8585',  256 SEND AND RECV RUSIZE          X
             PSERVIC=X'060200000000000000000300'
```

OS/2 LU0

Use the following entry for OS/2 LU0.

```
CDLM2K MODEENT LOGMODE=CDLM2K                                        X
                COS=CDCOS1,                                         X
                FMPROF=X'04',                                       X
                TSPROF=X'04,                                        X
                PSNDPAC=X'10',                                      X
                SRCVPAC=X'10',                                      X
                SSNDPAC=X'10',                                      X
                PRIPROT=X'B3',                                      X
                SECPROT=X'B3',                                      X
                COMPROT=X'70A0',                                    X
                RUSIZES=X'8888',                                    X
                PSERVIC=X'000000000000000000000000'
```

MS-DOS LU6.2

Use the following entry for MS-DOS LU6.2.

```
NDM62PC  MODEENT LOGMODE=NDM62PC,                                   X
                TYPE=1,                                            X
                COS=NDMCOS1,                                       X
                FMPROF=X'13',                                      X
                TSPROF=X'07',                                      X
                PRIPROT=X'B0',                                     X
                SECPROT=X'B0',                                     X
                COMPROT=X'50B1',                                   X
                PSNDPAC=X'05',                                     X
                SRCVPAC=X'05',                                     X
                SSNDPAC=X'05',                                     X
                RUSIZES=X'8686',                                   X
                PSERVIC=X'060200000000000000000000'
```

LU2 Nodes

Use the following entries for LU2 transfers.

```
*    3278 MODEL 2 DEFAULT PRESENTATION SPACE SIZE            X
  D4C32782 MODEENT LOGMODE=D4C32782,                         X
                 FMPROF=X'03',                                X
                 TSPROF=X'03',                                X
                 PRIPROT=X'B1',                               X
                 SECPROT=X'90',                               X
                 COMPROT=X'3080',                             X
                 RUSIZES=X'87F8',                             X
                 PSERVIC=X'0200000000000185000007E00'
*    3278 MODEL 3 DEFAULT PRESENTATION SPACE SIZE            X
 D4C32783 MODEENT LOGMODE=D4C32783,                          X
                 FMPROF=X'03',                                X
                 TSPROF=X'03',                                X
                 PRIPROT=X'B1',                               X
                 SECPROT=X'90',                               X
                 COMPROT=X'3080',                             X
                 RUSIZES=X'87F8',                             X
                 PSERVIC=X'02000000000000205020507F00'
*    3278 MODEL 4 DEFAULT PRESENTATION SPACE SIZE            X
 D4C32784 MODEENT LOGMODE=D4C32784,                          X
                 FMPROF=X'03',                                X
                 TSPROF=X'03',                                X
                 PRIPROT=X'B1',                               X
                 SECPROT=X'90',                               X
                 COMPROT=X'3080',                             X
                 RUSIZES=X'87F8',                             X
                 PSERVIC=X'0200000000000002B502B507F00'
*    3278 MODEL 5 DEFAULT PRESENTATION SPACE SIZE            X
 D4C32785 MODEENT LOGMODE=D4C32785,                          X
                 FMPROF=X'03',                                X
                 TSPROF=X'03',                                X
                 PRIPROT=X'B1',                               X
                 SECPROT=X'90',                               X
                 COMPROT=X'3080',                             X
                 RUSIZES=X'87F8',                             X
                 PSERVIC=X'02000000000000018501B847F00'
```

Structured Fields

Use the following entry for structured fields.

```
PC3270M2 MODEENT LOGMODE=PC3270M2,                           X
                 FMPROF=X'03',                                X
                 TSPROF=X'03',                                X
                 PRIPROT=X'B1',                               X
                 SECPROT=X'90',                               X
                 COMPROT=X'3080',                             X
                 RUSIZES=X'87F8',                             X
                 PSERVIC=X'0280000000000185000007E00'
```

Non-SNA Cluster Controller

Use the following entry for non-SNA Cluster controller.

```
NONSNALC MODEENT LOGMODE=NONSNALC,                              X
                 FMPROF=X'02',                                  X
                 TSPROF=X'02',                                  X
                 PRIPROT=X'71',                                 X
                 SECPROT=X'40',                                 X
                 COMPROT=X'2000',                               X
                 RUSIZES=X'87C7',                               X
                 PSERVIC=X'020000000000185000007E00'
```

Remote X.25/ASYNC SDLC Modem

Use the following entry for remote X.25/ASYNC SDLC modem.

```
X25NDMPC MODEENT LOGMODE=X25NDMPC,                              X
                 FMPROF=X'03',                                  X
                 TSPROF=X'03',                                  X
                 PRIPROT=X'B1',                                 X
                 SECPROT=X'90',                                 X
                 COMPROT=X'3040',                               X
                 RUSIZES=X'8989',                               X
                 PSERVIC=X'010000000000000000000000'
```

Define Cross-Domain Resource Manager Minor Nodes

Define the cross-domain resource manager nodes. The following figure
shows CONNECT:Direct Cross-Domain Resource Manager node
definitions. The sample is located in member NDMCDRM.

```
 CDRMA    VBUILD TYPE=CDRM
 AAAA     CDRM  SUBAREA=20,ELEMENT=1,ISTATUS=ACTIVE
 BBBB     CDRM  SUBAREA=30,ELEMENT=1,ISTATUS=ACTIVE
 CCCC     CDRM  SUBAREA=40,ELEMENT=1,ISTATUS=ACTIVE
```

Each NDMCDRM entry defines another VTAM domain or equivalent
with which this domain will communicate. These entries will be
defined in the domain in which this CONNECT:Direct is being defined.

Define Cross-Domain Resources

Define the cross-domain resources. The following figure shows a CONNECT:Direct Cross-Domain Resource definition. The example is located in member NDMCDRSC. Each CDRSC entry defines an applications (another CONNECT:Direct) in another domain.

```
CDRSCNA  VBUILD TYPE=CDRSC
NDMAPP2  CDRSC CDRM=BBBB
NDMAPP3  CDRSC CDRM=CCCC
```

Note: The Unformatted Systems Services (USS) table is not used by CONNECT:Direct; however, if you define a USS table for the LUs to be used with AS/400 communication devices, or with OpenVMS or Tandem logical units, you cannot include a MSG10 definition in the table.

Multiple DTFs

You can define multiple DTFs to run on the same machine or in the same VTAM. There is only one VTAM APPL per CONNECT:Direct DTF and normally only one VTAM application per site. If two DTFs are to run on the same machine or under the same VTAM, then define two VTAM APPLs.

When defining APPLs for multiple CONNECT:Directs, they may all reside in the same application major node (VBUILD TYPE=APPL) in a CMS file of type VTAMLST on the virtual machine VTAM's 191 minidisk.

Note: The default logmode specified with the DLOGMOD keyword of the APPL definition must be an LU0 logmode, *not* LU6.2 or LU2 logmode.

Network or Domain Name

The VTAM APPL definitions for CONNECT:Direct nodes are in the supplied examples with the name field value (column 1) equal to the ACBNAME value. However, these names may be different. If they are

different the network name value contained in the name field must be unique within the network, and the value in the ACBNAME field must be unique within the domain.

If these names are different, follow these rules when defining the Network Map:

1. The names specified in the APPLIDS keyword must be the domain names.

2. The name specified for the LOCAL.NODE DTF APPLID must be the network name.

3. The name specified for the ADJACENT.NODE DTF APPLID must be the network name.

Sample VTAM Definitions

The following examples show definitions for two nodes with unique network and domain names.

The first figure shows definitions for CD.NODE.A.

```
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
*           APPLICATION DEFINITION                               *
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
*
*APPLICATION DEFINITION FOR USER DTF INTERFACE
*
NAN01     APPL      ACBNAME=NAID01,
                    etc.
*
NAN02     APPL      ACBNAME=NAID02,
                    etc.
*
*APPLICATION DEFINITION FOR CONNECT:Direct
*
CDNAPP1   APPL      ACBNAME=CDDAPP1,
                    etc.
*
* * * * * * *  * * * * * * * * * * * * * * * * * * * * * * * * * *
*        CROSS-DOMAIN MANAGER DEFINITION
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
*
CDRMA     VBUILD    TYPE=CDRM
SA30      CDRM                SUBAREA=30,ELEMENT=1,ISTATUS=ACTIVE
SA40      CDRM                SUBAREA=40,ELEMENT=1,ISTATUS=ACTIVE
*
* * * * * * *  * * * * * * * * * * * * * * * * * * * * * * * * * *
*        CROSS-DOMAIN DEFINITION
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
*
CDRSCNA   VBUILD    TYPE=CDRSC
CDNAPP2   CDRSC     CDRM=SA30
*
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
*        NETWORK MAP DEFINITION*
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
*
LOCAL.NODE=((CD.NODE.A   CDNAPP1 ,,  XXYYZ)-
         TCQ=(CD.NODEA.TCX   CD.NODEA.TCQ))
ADJACENT.NODE=(PARSESS (12 2) (CD.NODE.A ,  CDNAPP6) -
         APPLIDS=(NAID01 NAID02))
ADJACENT.NODE=(PARSESS (12 2) (CD.NODE.B ,  CDNAPP2)-
         APPLIDS=(NAID01 NAID02))
```

The next figure shows definitions for CD.NODE.B.

```
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
*          APPLICATION DEFINITION                                   *
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
*
*APPLICATION DEFINITION FOR USER DTF INTERFACE
*
NAN01      APPL      ACBNAME=NAID01,
                     etc.
*
NAN02      APPL      ACBNAME=NAID02,
                     etc.
*
*APPLICATION DEFINITION FOR CONNECT:Direct
*
CDNAPP2    APPL      ACBNAME=CDDAPP1,
                     etc.
*
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
*          CROSS-DOMAIN MANAGER DEFINITION                          *
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
*
CDRMA      VBUILD    TYPE=CDRM
SA30       CDRM                SUBAREA=30,ELEMENT=1,ISTATUS=ACTIVE
SA20       CDRM                SUBAREA=20,ELEMENT=1,ISTATUS=ACTIVE
*
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
*          CROSS-DOMAIN DEFINITION                                  *
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
*
CDRSCNA    VBUILD    TYPE=CDRSC
CDNAPP1    CDRSC     CDRM=SA20
*
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
*          NETWORK MAP DEFINITION                                   *
* * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
*
LOCAL.NODE=((CD.NODE.B   CDNAPP2 ,,  XXYYZ)                      -
           TCQ=(CD.NODEA.TCX   CD.NODEA.TCQ))
ADJACENT.NODE=(PARSESS (12 2) (CD.NODE.B  , CDNAPP2)             -
           APPLIDS=(NAID01 NAID02))
ADJACENT.NODE=(PARSESS (12 2) (CD.NODE.A  , CDNAPP1)             -
           APPLIDS=(NAID01 NAID02))
```

Chapter 5

Preparing VTAM and NCP Parameters

This chapter explains how to set up VTAM and NCP parameters that you need to define for CONNECT:Direct operation.

Preparing for CONNECT:Direct Operation

Perform the following evaluations to prepare for CONNECT:Direct operation:

▶ Determine correct RU size for SNA sessions

The VTAM RUSIZES parameter is located in the VTAM MODEENT macro. It specifies the transmission buffer size and may affect CONNECT:Direct operation. For more information on the VTAM RUSIZES parameter, see the *Selecting RU Size for CONNECT:Direct SNA Sessions* section that follows.

▶ Determine the effect of NCP parameters

Several parameters located in the NCP macros used during NCP generation (GEN) may also affect CONNECT:Direct operation. For more information on the effect of these parameters on CONNECT:Direct operation, see the *Determining Effects of NCP Parameters* section beginning on page 5-4.

Selecting RU Size for CONNECT:Direct SNA Sessions

The VTAM RUSIZES parameter in the CONNECT:Direct VTAM log mode table specifies the request/response unit (RU) size for CONNECT:Direct. The following example shows a mode table entry with a secondary and primary logical unit RU size of 1,024 bytes.

```
NDMTAB   MODETAB
         MODEENT LOGMODE=NDMLOGM,                        X
         RUSIZES=X'8787'                 /*X'ABAB'*/
```

Selecting RU Size for Connections Between CONNECT:Direct Systems

The VTAM RUSIZES parameter in the CONNECT:Direct VTAM log mode table specifies the request/response unit (RU) size for CONNECT:Direct. In selecting an RU size for CONNECT:Direct, it is important to know how RU size relates to the NCP MAXDATA parameter.

The NCP MAXDATA value is specified within the NCP physical unit (PU) macro definition. It specifies, in bytes, the maximum amount of data that the NCP can receive from the PU at one time. This amount includes the transmission header and the request/response header, which totals 29 bytes for CONNECT:Direct.

For example, if the CONNECT:Direct VTAM log mode table entry specifies 4,096 bytes, the NCP MAXDATA value must be at least 4,125 bytes for CONNECT:Direct to function correctly. In other words, the MAXDATA value must be at least 29 bytes larger than the RU size specified in the mode table.

- RUSIZES for DTF-to-DTF transmission is from 256 bytes to 64K bytes. However, for LU6.2, the RUSIZE must be between 512 and 30,720 bytes.

- RUSIZES for CONNECT:Direct API-to-DTF transmission is 1K to 64K bytes. You may want to use a value of 7K.

Selecting RU Size and Pacing for Connections Between VM and PC

For the CONNECT:Direct-to-CONNECT:Direct for MS-DOS connection, use an RU size of 512 bytes (RUSIZES=X'8686') or 1,024 bytes (RUSIZES=X'8787'). For this connection, MAXDATA must equal 265 because the use of RU segmenting.

Note: For locally attached 3274 connections, specify RUSIZES=X'87C7'.

Understanding RU Sizing

The first two digits of the RUSIZES value apply to the secondary logical unit (LU). The last two digits apply to the primary LU. The first and third digits are represented as **a** in the following table , and the second and fourth digits are represented as **b**. In the example, 87 indicates an RU size of 1,024.

The following table will help you to find the appropriate value to specify for the RUSIZES parameter. The asterisks, along with the bold-faced and underlined number (1,024), indicate how to use the chart to determine the RU size for the preceding example.

	a							
b	8*	9	A(10)	B(11)	C(12)	D(13)	E(14)	F(15)
0	8	9	10	11	12	13	14	15
1	16	18	20	22	24	26	28	30
2	32	36	40	44	48	52	56	60
3	64	72	80	88	96	104	112	120
4	128	144	160	176	192	208	224	240
5	256	288	320	352	384	416	448	480
6	512	576	640	704	768	832	896	960
7*	**1,024**	1,152	1,280	1,408	1,536	1,664	1,792	1,920
8	2,048	2,304	2,560	2,816	3,072	3,328	3,584	3,840
9	4,096	4,608	5,120	5,632	6,144	6,656	7,168	7,680
A(10)	8,192	9,216	10,240	11,264	12,288	13,312	14,336	15,360
B(11)	16,384	18,432	20,480	22,528	24,576	26,624	28,672	30,720
C(12)	32,768	36,864	40,960	45,056	49,152	53,248	57,344	61,440
D(14)	65,536	73,728	81,920	90,112	98,304	106,496	114,688	122,880
E(14)	131,072	147,456	163,840	180,224	196,608	212,992	229,376	245,760
F(15)	262,144	294,912	327,680	360,448	393,216	425,984	458,752	491,520

Calculating RUSIZES

The table is derived from the following calculations:

The RUSIZES of **ab** means RUSIZES equals **a X 2 **b**. The **b** of each **ab** pair is used as an exponent of base two. The resulting value is multiplied by **a** to get the RUSIZE for that logical unit.

Using the same example, **RUSIZES=87F8**, the SLU and PLU values are evaluated as follows:

```
 If     RUSIZES=X'87F8' then

        the SLU RUSIZE is 87 which indicates
        8  X  2**7  = 8 X 128 = 1024
        (8 times 2 to the 7th power)

        the PLU RUSIZE is F8 which  indicates
        F  X  2**8 = 15 X 2**8 = 15 X 256 = 3840
        (15 times 2 to the 8th power)
```

# Determining Effects of NCP Parameters

Review the following NCP parameters carefully during the CONNECT:Direct installation.

GEN Macro	Parameter	Definition
BUILD	BFRS	Size of NCP buffers.
HOST	BFRPAD	Number of pad characters inserted by NCP.
HOST	MAXBUFRU	Number of buffers the access method (VTAM) allocates to receive data from the NCP.
HOST (VTAM)	UNITSZ	Size of access method buffers used for data transfer from NCP to CONNECT:Direct.
LINE	TRANSFR	Number of NCP buffers corresponding to maximum amount of data NCP can receive from another NCP.
PU unit	MAXDATA	Maximum path information (PIU) size.
MODEENT	RUSIZES	CONNECT:Direct (APPL) buffer size.

The following figure illustrates the relationship between these VTAM and NCP parameters. When data passes from VM1 to NCPA, the MAXDATA parameter determines the amount of data that NCPA can receive in one segment of a path information unit (PIU).

Between NCPA, NCPB, and NCPC, if you do not specify the TRANSFR parameter on the LINE statement, VTAM will search the BUILD statement. If you do not define TRANSFR on either statement, the default is taken. See the IBM *VTAM Customization* manual.

When NCPB passes data to VM2, the MAXBUFRU times the UNITSZ determines the amount of data that can be passed to VM2. The data cannot exceed the size in bytes of the VTAM IOBUF buffer pool allocation parameters.
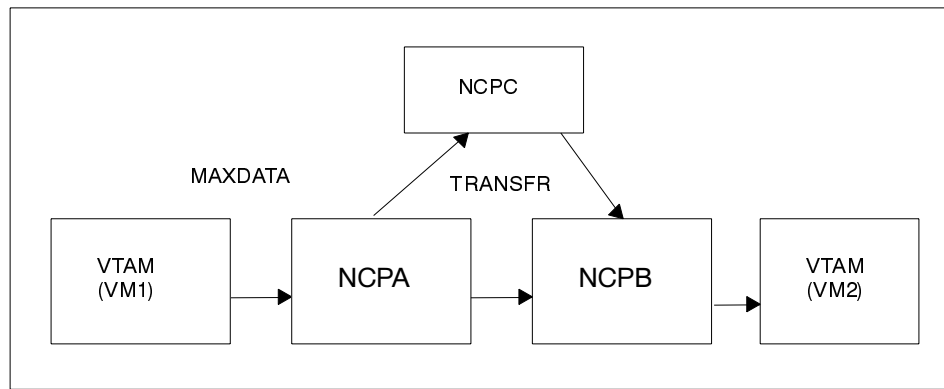
Calculating Minimum Value of NCP TRANSFR

To calculate the minimum value of an NCP TRANSFER, each NCP-to-NCP connection is defined to accommodate the maximum RU size defined for a CONNECT:Direct DTF-to-DTF session. This value is controlled by the NCP TRANSFR parameter of each NCP LINE macro defining an NCP-to-NCP link. The following is a summary of the calculations required to determine the NCP TRANSFER value:

1. Determine the maximum amount of data to be received on this line definition. For CONNECT:Direct, this will be the RU size plus 29 bytes for the request/response headers.

2. Add 24 bytes to this value for required NCP overhead (BFRPAD listed in the table on page 5-4).

3. Divide the sum by the NCP buffer size (BFRS listed in the table on page 5-4).

4. Round the result to the next highest integer. This is the minimum value that you can specify for TRANSFR which corresponds to the specified RU size.

Changing the TRANSFR Parameter

If you change the TRANSFR parameter in one NCP in a network, all other NCPs in that network require the same change. If you cannot easily change the NCP TRANSFR parameter, use the reverse process described in the following steps to calculate the maximum RU size for use with CONNECT:Direct.

1. Multiply the NCP TRANSFR parameter by the NCP BFRS value. This is the maximum amount of data which NCP can receive.

2. Subtract 24 bytes for required NCP overhead (BFRPAD listed in the table on page 5-4).

3. Subtract 29 bytes for the request/response header.

4. Find the next lowest RU size value from the table on page 5-3. This is the largest RU which can be specified for use by CONNECT:Direct.

The amount of data VTAM can receive from the NCP is determined by the product of MAXBUFRU times UNITSZ. This value must be less than or equal to the amount specified on the IOBUF buffer pool allocation parameters in the ATCSTRxx start options list of VTAMLST.

Part II

Installation

Chapter 6

Installing CONNECT:Direct for VM/ESA

This chapter explains how to install CONNECT:Direct for VM/ESA. CONNECT:Direct job streams initially run to install CONNECT:Direct and apply maintenance.

Executing an installation EXEC fills in skeletons and produces these job streams. The EXEC displays panels that allow you to define the CONNECT:Direct installation to the specific requirements of your site.

Before starting an installation, notify your tape management group that you are installing a tape with a given volume serial number to eliminate the possibility of accessing a different tape with the same volume serial number.

Installation Considerations

Before you begin your installation, consider the following:

▶ The *CONNECT:Direct for VM/ESA Release Notes* documents any changes in the product or the installation process. Read this document before attempting your installation.

▶ Refer to the *Initialization Parameters* appendix in the CONNECT:Direct for VM/ESA Administration Guide for complete definitions of each initialization parameter.

▶ See the guidelines in the *Tuning CONNECT:Direct for VM/ESA for Performance* chapter beginning on page 7-1 after installing CONNECT:Direct to enhance the performance of statistics file pairs and the CMS File Request Server.

Step 1 — Configure the GCS Virtual Machine for CONNECT:Direct

CONNECT:Direct for VM/ESA requires one GCS virtual machine. The GCS virtual machine runs the Master Application Program, which runs the following programs as subtasks:

▶ The GCS API, which is the portion of the API providing VTAM services.

▶ The GCS Message File Program, which provides VSAM services for the CONNECT:Direct message facility running under ISPF/PDF.

▶ The CONNECT:Direct DTF, which executes CONNECT:Direct commands and Process statements.

The Master Application Program allows these programs to be run in one virtual machine, rather than dedicating a virtual machine to each program.

Caution: For VM/ESA, the NSSs (Named Saved Segments and Named Saved Systems) for VSAM, GCS, GCSBAM, and VTAM cannot overlap in the GCS machine definition. You can check this by reviewing an NSS display. Enter **QUERY NSS ALL MAP** to display all NSSs or enter **QUERY NSS NAME** *segname* **MAP** to display a single NSS. These segment addresses must not overlap, otherwise CONNECT:Direct for VM/ESA will not initialize or function properly. The **QUERY NSS** command requires a privilege class of **E** to execute.

Complete the following steps to configure the GCS virtual machine for CONNECT:Direct for VM/ESA.

1. Assign the CONNECT:Direct GCS virtual machine a CP privilege class of BG for System Resource Operator and General User.

2. Verify the GCS virtual machine has at least 6 megabytes of memory available. Sterling Commerce recommends 24 megabytes.

    Saved segments within this storage range will subtract from CONNECT:Direct's available storage. You must allow two segments for GCS, two segments for VSAM, and one segment for VTAM when you plan your storage requirements.

3. Add the following control statements and parameters to the Control Program (CP) USER DIRECTORY for the CONNECT:Direct GCS virtual machine:

Parameter	Function
IUCV ALLOW	Allows the IUCV environment to be used.
MAXCONN NN	Set NN=(2 x the number of MAXUSERS), where MAXUSERS is the same value specified in the CONNECT:Direct MAXUSERS initialization parameter.

The following is a sample CP directory entry for the VM ID that will initial program load (IPL) the CONNECT:Direct GCS machine and run CONNECT:Direct for VM/ESA.

```
USER      CDVM      PASS      24M           64M       BG
NAMESAVE  GCS
IPL       CMS       PARM      AUTOCR
OPTION    MAXCONN   NN
CONSOLE   01F       3215
IUCV      ALLOW
SPOOL     00C       2540      READER *
SPOOL     00D       2540      PUNCH A
SPOOL     00E       1403      A
LINK      MAINT     190       190           RR
LINK      MAINT     19D       19D           RR
LINK      MAINT     19E       19E           RR
MDISK     191       3380      121   050     CDVM1   MR  RPASS     WPASS
```

If you choose not to add your data sets to an existing VSAM Master Catalog, then allocate the following minidisk in your GCS userid for VSAM:

```
 MDISK   195  3380  nnn  nnn  CDVSAM  MR  RPASS  WPASS  MPASS
```

Note: To determine the number of cylinders or blocks needed refer to *Planning the DASD Requirements for Distribution and Installation Files* chapter beginning on page 3-1.

Once CONNECT:Direct is installed, change the second line of this CP directory to automatically run the IPL for the CONNECT:Direct GCS machine.

Use the following change:

```
 IPL   GCS   PARM   AUTOLOG
```

This command causes the GCS virtual machine to be IPLed rather than the CMS virtual machine.

4. When you generate the GCS system and authorize the CONNECT:Direct GCS virtual machine so it can run in the supervisor state.

Note: To prevent system lock when a GCS dump is created, do not authorize the Common Dump Receiver during GCS GEN.

Step 2 — Configure the SFS Virtual Machine for CONNECT:Direct (Optional)

CONNECT:Direct for VM/ESA supports the Shared File System (SFS) through a SFS Server which runs in a separate virtual machine. You specify the name of this virtual machine with the SFS.SERVER.VMID= initialization parameter. You must configure SFS Server virtual machine if SFS files will participate in CONNECT:Direct file copies.

Complete the following steps to configure the CONNECT:Direct SFS Server virtual machine.

1. Assign the CONNECT:Direct SFS Server virtual machine a CP privilege class of BG for System Resource Operator and General User.

2. Verify that the CONNECT:Direct SFS Server virtual machine has at least 4 megabytes of memory available.

3. Add the following control statements and parameters to the Control Program (CP) USER DIRECTORY for the CONNECT:Direct SFS Server virtual machine:

Parameter	Function
IUCV ALLOW	Allows the IUCV environment to be used.
MAXCONN NN	Set NN=(2 x the number of MAXUSERS), where MAXUSERS is the same value specified in the CONNECT:Direct MAXUSERS initialization parameter.

The following is a sample CP directory entry for the VM ID that will run the CONNECT:Direct SFS Server.

```
USER      CDVMSFS  PASS     4M         4M      BG
IPL       CMS      PARM     AUTOCR
OPTION    QUICKDSP MAXCONN             nn
CONSOLE   01F      3215
IUCV      ALLOW
SPOOL     00C      2540     READER          *
SPOOL     00D      2540     PUNCH           A
SPOOL     00E      1403     A
LINK      MAINT    190      190        RR
LINK      MAINT    19D      19D        RR
LINK      MAINT    19E      19E        RR
MDISK     191      3380     121  030   CDVM1  MR  RPASS    WPASS
```

4. Log on to the CONNECT:Direct SFS Server virtual machine and edit
    PROFILE EXEC.

    The following example shows the format of the CMS commands
    that must be included in the PROFILE EXEC file.

```
/* Link to CONNECT:Direct for VM/ESA distribution disk        */
   "CP LINK userid ccuu1 ccuu2 RR password "
   "ACC ccuu2 M/A "
   "GLOBAL LOADLIB CDV3200 "
   "GLOBAL TXTLIB DMSAMT VMMTLIB "
   "GLOBAL CSLLIB VMLIB "
   "FILEDEF SYSOUT PRINT PERM "    /* Used for Tracing */
   "FILEDEF PERF PRINT PERM "    /* Used for Tracing */
   Queue "OSRUN DMFRSMAI "
   Exit 0
```

5. The SFS Server userid should be defined to the CMS Shared File
    System with SFS Administrator authority.

    Note: The CONNECT:Direct SFS Server virtual machine is really an
    extension of CONNECT:Direct for VM and as such should be
    given the same performance options (SHARE and so on),
    that the CONNECT:Direct DTF service machine has been
    given.

Step 3 — Download Tape and Configure VSAM MDISK

Use CDSETUP to quickly install CONNECT:Direct. The CDSETUP
utility accepts the installation defaults contained on the installation tape.

The CDSETUP utility performs the following functions:

▶   Copies all CONNECT:Direct for VM/ESA build and run source
    from the installation tape on to the program mdisk

▶ Formats a VSE/VSAM CONNECT:Direct for VM/ESA system mdisk

Processing time is usually two minutes for a successful installation.

Setting Up the CDSETUP Utility

You must complete these steps before running the CDSETUP utility:

1. Logon to the VM userid that is being used as the CONNECT:Direct for VM/ESA service machine. You must IPL CMS for the installation task.

   Note: You must IPL CMS, not GCS, for the installation task.

   ---

2. Verify that you have read/write capability to your VSAM mdisk. This mdisk generates a VSE/VSAM mdisk. Do not format the mdisk. The CDSETUP utility formats the mdisk for you.

3. Access the mdisk that will contain the downloaded tape material as your 'A' disk. This information is usually stored on a 191 mdisk from the CONNECT:Direct for VM/ESA userid.

   The 'A' mdisk must be in standard 4K CMS format with a minimum of 6000 4k blocks available.

   Note: Sterling Commerce suggests you keep back levels of CONNECT:Direct for VM/ESA on a different mdisk from the current installation 'A' mdisk.

   For example, if you save your existing Network Map to the "A" mdisk, you may erase it later in the installation.

   ---

4. Mount the install tape on a tape drive attached to your userid as 181.

5. Enter this command to extract CDSETUP from the install tape.

```
VMFPLC2 FSF 4 (181
```

Issue the following command to load the CDSETUP EXEC.

```
VMFPLC2 LOAD CDSETUP EXEC A (181
```

Using the CDSETUP Utility

Complete the steps described in the *Setting Up the CDSETUP Utility* section beginning on page 6-6 before you can use the CDSETUP utility.

Issue the following command to run the utility:

```
CDSETUP
```

CDSETUP takes you to the CONNECT:Direct for VM/ESA Installation menu. Select only one of the four options from this menu.

```
                    CONNECT:DIRECT FOR VM/ESA INSTALLATION


SELECT ONE OF THE FOLLOWING FOUR OPTIONS:

'B'  TAPE DOWNLOAD AND VSAM MDISK FORMAT

'T'  TAPE DOWNLOAD ONLY — NO VSAM FUNCTIONS

'V'  VSAM MINIDISK FORMAT — NO TAPE FUNCTIONS

'X'  STOP ALL PROCESSING AND ABORT PROGRAM







ENTER OPTION
```

Note: Most installations will use option 'B' to download the installation tape and update the VSAM disk.

CDSETUP Error Messages

Skip this section unless you have an error message while running the CDSETUP utility. The CDSETUP utility checks your 'A' mdisk to ensure the availability of 6000 blocks of free space to hold the installation tape material.

The utility ends without downloading the tape and issues an error message in the following situations:

▶ the mdisk does not have 6000 blocks of free space available
▶ the utility cannot access the 'A' mdisk

If you receive one of these error messages, review the steps in the *Setting Up the CDSETUP Utility* section beginning on page 6-6 to verify your hardware connections and the availability of free space on the 'A' mdisk.

If the 'A' mdisk already contains CONNECT:Direct files, CDSETUP issues a warning message withe the option to continue or abort the job. If you choose to continue the job, CDSETUP replaces all existing 'A' mdisk files with the same FN and FT that matches a install tape file.

Note: Sterling Commerce suggests you keep back levels of CONNECT:Direct for VM/ESA on a different mdisk from the current installation 'A' mdisk.

For example, if you save your existing Network Map to the "A" mdisk, you may erase it later in the installation.

An information message displays to inform you when the actual install tape download begins. Processing time is usually two minutes for a successful installation. A completion message displays after a successful download.

If you selected option 'T' from the CONNECT:Direct for VM/ESA Installation menu, the CDSETUP utility ends here.

If you selected option 'B' from the CONNECT:Direct for VM/ESA Installation menu, the CDSETUP utility continues with the VSAM mdisk function.

Configuring the VSAM mdisk

CDSETUP automates the VSAM 195 (or user-selected) mdisk configuration. The utility runs until it completes the VSAM 195 mdisk configuration, issues a Return Code, and displays a completion message after a successful configuration.

The CDSETUP utility overwrites the 195 mdisk into a VSE/VSAM format, allocates the VSAM master catalog on the 195 mdisk, and pre-allocates VSAM file space.

If the mdisk 195 is not large enough to support the CONNECT:Direct for VM/ESA VSAM requirements or if other related errors occur, a non-zero Return Code displays.

If you receive a non-0 Return Code, complete the following steps:

1. Run the CDSETUP utility again by entering CDSETUP (NOTAPE).

2. Select option 'V' on the CONNECT:Direct for VM/ESA Installation menu. You only need to restart using option 'V'.

Step 4 — Prepare VTAM Definitions

Defining local and remote CONNECT:Direct nodes is a four-part procedure:

1. Define the applid of the local CONNECT:Direct node and multiple applids for the interactive users on the same node.

   The CONNECT:Direct for VM/ESA file NDMAPPL NDMVTAM provides an example of the CONNECT:Direct applid definition to VTAM. The file NDMIAPPL NDMVTAM provides an example of an interactive userid definition.

2. Define the mode table entries to be used with each applid.

   The CONNECT:Direct for VM/ESA file NDMMODET NDMVTAM provides an example of how to define the mode table entries to be used with applids.

3. Define the cross-domain resource manager minor nodes.

   The CONNECT:Direct for VM/ESA file NDMCDRM NDMVTAM provides an example of how to define Cross-Domain Resource Manager minor nodes.

4. Define the cross-domain resources.

   The CONNECT:Direct for VM/ESA file NDMCDRSC NDMVTAM provides an example of how to define the cross-domain resources.

Step 5 — Run CDINSTAL

In this step, you execute CDINSTAL to install CONNECT:Direct on your system. Specific installation instructions for a 9335 DASD device and a 3380 or 3390 DASD device follow.

Installing on a 9335 DASD Device Only

Before running CDINSTAL, edit the NDMSKL1 AMSERV distribution file. Change the space allocation field on the cluster definition for the CONNECT:Direct checkpoint data set to the following:

```
FROM Records (40 6)
TO Records (1500 1500)
```

Note: Do not insert sequence numbers in the NDMSKL1 AMSERV file. Sequence numbers in this file cause AMSERV problems. To remove sequence numbers from a CMS file, use the following CMS XEDIT commands:

```
SET TRUNC 80
SET ZONE 1 80
SET SERIAL OFF
LOCATE :1
CLOCATE  :  73
CDELETE  8
REPEAT *
```

Installing on a 3380, 3390, or 9335 DASD Device

Use the following instructions to run the CDINSTAL EXEC:

1. Issue CP and CMS commands to link and access the CONNECT:Direct VSAM disk.

2. Define the location of the VSAM Master Catalog.

3. Define the CONNECT:Direct load library as the library to search for modules.

The following is an example of the required commands to issue. This information should be added to the PROFILE EXEC of the DTF userid.

```
CP LINK userid ccuu1 ccuu2 MW password
ACCESS 195 B
DLBL IJSYSCT B DSN mastcat (PERM
GLOBAL LOADLIB CDVnnnn
```

Using CDINSTAL to Create and Load the Network Map

The CDINSTAL EXEC installs CONNECT:Direct by displaying a series of panels that take you through steps for naming files, unit names for devices, volume serials, and so on.

All panels include tutorials, which can be accessed by pressing the **PF1/HELP KEY**, that explain the information requested on a panel.

Note: Select all options for base installations. Customize the file
NTMPLOAD EXEC to update existing Network Maps.

Creating a Network Map for a New Installation

The CDINSTAL EXEC builds the CONNECT:Direct Network Map. The
Network Map defines the local and adjacent node names and their
corresponding VTAM applids, as well as VTAM applids and TCP/IP
nodes for interactive users. CONNECT:Direct Processes reference the
logical node names in the Network Map.

Enter the following command to execute CDINSTAL:

```
CDINSTAL
```

The first installation panel allows you to specify which files will be
defined and loaded, as well as the naming conventions for
CONNECT:Direct VSAM files. Any existing VSAM files are deleted,
and new VSAM files are defined.

Note: The delete steps produce a condition code of 8 if there are no old
files to delete. This is a normal condition for a new installation.

To end the installation at any point, press the **PF3/END KEY**. Any
defined files are kept, but these defined files will be deleted when the
installation EXEC is rerun with the option to define the files selected.

```
                 CONNECT:Direct for VM/ESA
              INSTALLATION - RESOURCE DEFINITION

    DEFINE VSAM FILES      ===>    Y  (Y or N)
    DEFINE NETWORK MAP     ===>    Y  (Y or N)
    LOAD MESSAGES          ===>    Y  (Y or N)
    LOAD NETWORK MAP       ===>    Y  (Y or N)

    VSAM FILE NAME HIGH-LEVEL QUALIFIER
    ==>    HLVSAM

    VSAM CATALOG NAME FOR CONNECT:Direct FILES
    ==>    MASTCAT

    VSAM OWNER ID FOR CONNECT:Direct FILES
    ==>  CDVM32

    VOLUME FOR VSAM FILES
    ==>  CDVSAM

    FILEMODE LETTER FOR CONNECT:Direct VSAM FILES
    ==>  B
```

Enter **Y** at the DEFINE NETWORK MAP prompt to set up the
CONNECT:Direct Network Map for a new installation.

Use the second panel to define the local and adjacent nodes for your
Network Map.

```
              CONNECT:Direct for VM/ESA GENERATE LOCAL/ADJACENT NODE

 ENTER THE LOGICAL NAME FOR THE LOCAL NODE
      ==> _____

 ENTER TWO VTAM APPLIDS SEPARATED BY A BLANK
      ==> _____

 ENTER THE SUPERUSER PASSWORD FOR THE LOCAL NODE
      ==> ____

 ENTER THE INTERACTIVE VTAM APPLIDS FOR THE LOCAL NDM NODE:


         _____     _____     _____     _____     _____
         _____     _____     _____     _____     _____
         _____     _____     _____     _____     _____
         _____     _____     _____     _____     _____
```

The third panel defines which connections you want to define for your
Network Map. Insert a non-blank character beside the connections you
want to define.

Note: The CDINSTAL EXEC will only process the options you select
from this menu. Select all options for base installations.

```
                       CONNECT:Direct for VM/ESA
                       UPDATE NETWORK MAP PANEL

       PLACE A NON-BLANK CHARACTER BESIDE THE CONNECTIONS YOU WANT TO DEFINE


  =>  ALL VTAM NODES

  =>  ALL TCP/IP VTAM NODES

  =>  MS-DOS VTAM NODES

  =>  TANDEM VTAM NODES

  =>  OS/400 LU6.2 VTAM NODES

  =>  OpenVMS VTAM NODES
```

The fourth panel allows you to define all VTAM nodes.

Note: Do not enter you local node as an adjacent node. The
CDINSTAL EXEC does this for you.

```
                    CONNECT:Direct for VM/ESA
               UPDATE NETWORK MAP PANEL - ALL VTAM NODES
              (EXCEPT FOR MS-DOS, TANDEM, OS/400 AND OpenVMS)

           ADJACENT          VTAM       PARSESS     SESSION LOGMODE
           NODE NAME         APPL       MAX DEFAULT TYPE    NAME      NETID

   1 ==>  _____  _____  ___  ___   _____  _____  _____
   2 ==>  _____  _____  ___  ___   _____  _____  _____
   3 ==>  _____  _____  ___  ___   _____  _____  _____
   4 ==>  _____  _____  ___  ___   _____  _____  _____
   5 ==>  _____  _____  ___  ___   _____  _____  _____
   6 ==>  _____  _____  ___  ___   _____  _____  _____
   7 ==>  _____  _____  ___  ___   _____  _____  _____
   8 ==>  _____  _____  ___  ___   _____  _____  _____
   9 ==>  _____  _____  ___  ___   _____  _____  _____
  10 ==>  _____  _____  ___  ___   _____  _____  _____
  11 ==>  _____  _____  ___  ___   _____  _____  _____
  12 ==>  _____  _____  ___  ___   _____  _____  _____
  13 ==>  _____  _____  ___  ___   _____  _____  _____
  14 ==>  _____  _____  ___  ___   _____  _____  _____

   NOTE: DO NOT ENTER YOUR LOCAL NODE AS AN ADJACENT
         NODE, THIS IS DONE FOR YOU.
```

The fifth panel allows you to define all TCP/IP nodes.

```
                    CONNECT:Direct for VM/ESA
               UPDATE NETWORK MAP PANEL - ALL TCP/IP NODES

           ADJACENT          TCP/IP     PARSESS     SESSION TCP/IP
           NODE NAME         PORT       MAX DEFAULT TYPE    ADDRESS

   1 ==>  _____  _____  ___  ___   _____  _____
   2 ==>  _____  _____  ___  ___   _____  _____
   3 ==>  _____  _____  ___  ___   _____  _____
   4 ==>  _____  _____  ___  ___   _____  _____
   5 ==>  _____  _____  ___  ___   _____  _____
   6 ==>  _____  _____  ___  ___   _____  _____
   7 ==>  _____  _____  ___  ___   _____  _____
   8 ==>  _____  _____  ___  ___   _____  _____
   9 ==>  _____  _____  ___  ___   _____  _____
  10 ==>  _____  _____  ___  ___   _____  _____
  11 ==>  _____  _____  ___  ___   _____  _____
  12 ==>  _____  _____  ___  ___   _____  _____
  13 ==>  _____  _____  ___  ___   _____  _____
  14 ==>  _____  _____  ___  ___   _____  _____
```

The sixth panel allows you to define MS-DOS nodes.

```
                    CONNECT:Direct for VM/ESA
              UPDATE NETWORK MAP PANEL - MS-DOS NODES

          ADJACENT          LOGICAL
          NODE NAME         UNIT NAME

   1 ==>  _____  _____
   2 ==>  _____  _____
   3 ==>  _____  _____
   4 ==>  _____  _____
   5 ==>  _____  _____
   6 ==>  _____  _____
   7 ==>  _____  _____
   8 ==>  _____  _____
   9 ==>  _____  _____
  10 ==>  _____  _____
  11 ==>  _____  _____
  12 ==>  _____  _____
  13 ==>  _____  _____
  14 ==>  _____  _____
```

The seventh panel defines the Tandem nodes for the Network Map.

```
                    CONNECT:Direct for VM/ESA
              UPDATE NETWORK MAP PANEL - TANDEM NODES
                                             SEPARATE MAX + DEF VALUES BY A BLANK
ADJACENT NODE NAME  ==> _____  PARSESS VALUES ==> _____ (MAX DEF)

ENTER SNODE LU NAMES BELOW:


        _____   _____   _____   _____   _____
        _____   _____   _____   _____   _____
        _____   _____   _____   _____   _____


ADJACENT NODE NAME  ==> _____  PARSESS VALUES ==> _____ (MAX DEF)

ENTER SNODE LU NAMES BELOW:


        _____   _____   _____   _____   _____
        _____   _____   _____   _____   _____
        _____   _____   _____   _____   _____
```

The eighth panel defines the OS/400 LU62 nodes for the Network Map.

```
                       CONNECT:Direct for VM/ESA
                UPDATE NETWORK MAP PANEL - OS/400 NODES

ADJACENT NODE NAME  ==> _____     VTAM LU62 ==> _____
   MAX CLASS ==> ___                                  DEFAULT CLASS ==> ___
   OS400 LIBRARY NAME ==> _____           LOGMODE NAME ==> _____

ENTER SNODE LU NAMES BELOW:


        _____     _____     _____     _____     _____
        _____     _____     _____     _____     _____
        _____     _____     _____     _____     _____


ADJACENT NODE NAME  ==> _____     VTAM LU62 ==> _____
   MAX CLASS ==> ___                                  DEFAULT CLASS ==> ___
   OS400 LIBRARY NAME ==> _____           LOGMODE NAME ==> _____

ENTER SNODE LU NAMES BELOW:


        _____     _____     _____     _____     _____
        _____     _____     _____     _____     _____
        _____     _____     _____     _____     _____
```

The ninth panel defines the VAX nodes for the Network Map.

```
                       CONNECT:Direct for VM/ESA
                UPDATE NETWORK MAP PANEL - VAX NODES
                                                SEPARATE MAX = DEF VALUES BY A BLANK
ADJACENT NODE NAME  ==> _____   PARSESS ==> _____ (MAX DEF)

ENTER PNODE LU NAMES BELOW:
        _____     _____     _____     _____     _____
        _____     _____     _____     _____     _____
        _____     _____     _____     _____     _____
ENTER SNODE LU NAMES BELOW:
        _____     _____     _____     _____     _____
        _____     _____     _____     _____     _____
        _____     _____     _____     _____     _____


ADJACENT NODE NAME  ==> _____   PARSESS ==> _____ (MAX DEF)

ENTER PNODE LU NAMES BELOW:
        _____     _____     _____     _____     _____
        _____     _____     _____     _____     _____
        _____     _____     _____     _____     _____

ENTER SNODE LU NAMES BELOW:
        _____     _____     _____     _____     _____
        _____     _____     _____     _____     _____
        _____     _____     _____     _____     _____
```

A completion message appears to let you know that CONNECT:Direct
has loaded the Network Map. An example of the completion message
follows:

```
CREATING NETMAP LOAD INPUT FILE:  CDIRECT NETMAP A
```

If your site is going to use an existing Network Map, you must customize the file NTMPLOAD EXEC, which was loaded during *Step 3– Download Tape and Configure VSAM MDISK.*

Note: Use the NTMPLOAD EXEC for all changes and maintenance after the initial CONNECT:Direct Network Map has been created by CDINSTAL.

The following figure is the NTMPLOAD EXEC file as it is distributed.

```
/* EXEC TO LOAD NDM NETWORK MAP.                                 */
/* - CHANGE MODE TO THE MODE WHERE THE NETMAP  FILE IS LOCATED. */
/* - CHANGE HILQ TO THE HIGH LEVEL QUALIFIER.                    */
/* - BE SURE A DLBL HAS BEEN ISSUED FOR IJSYSCT, THE VSAM        */
/*   MASTER CATALOG.                                             */
/* - BE SURE A GLOBAL LOADLIB COMMAND FOR THE NDM LOAD LIBRARY   */
/*   HAS BEEN ISSUED.                                            */
FILEDEF SYSOUT    TERM
'ESTATE CDIRECT NETMAP *'                               /* P3707 */
If rc ^= 0 then Exit                                    /* P3707 */
FILEDEF INPUT    DISK CDIRECT NETMAP
DLBL     NETMAP   MODE DSN 'HILQ.NETMAP'
SET      DOS OFF
OSRUN    DMCNTMPL
EXECOS
```

For more information on the Network Map, refer to the *Maintaining the Network Map* chapter in the *CONNECT:Direct for VM/ESA Administration Guide*.

You will need to use XEDIT to perform the following customizations:

1.   The two references to CDIRECT NETMAP in the previous NTMPLOAD EXEC file are in bold. Change the two occurrences of CDIRECT NETMAP to the CMS file name and file type of the file where you plan to maintain the source for the Netmap, if different from CDIRECT NETMAP. CDIRECT NETMAP is the name under which the initial source file for the netmap was stored because of running the DEFINE NETWORK MAP option of the CDINSTAL EXEC.

2.   Change MODE to the letter you have chosen to represent the minidisk containing the CONNECT:Direct system files.

3.   Change HILQ to the high–level qualifier that you chose for the VSAM system files when you ran the CDINSTAL EXEC.

Step 6 — File Migration Procedures (Optional)

This section outlines the procedures for migrating files from previous releases of CONNECT:Direct for VM to CONNECT:Direct for VM/ESA.

Note: This optional step is for existing CONNECT:Direct for VM customers only. Skip this step if you do not have to migrate existing VM files.

Required Variables for File Migration

Review this section before you begin to use the file migration utilities. Your file migration is easier if you understand the required variables.

CONNECT:Direct has two utilities to migrate your existing VM files:

▶ DMSTCONV utility for migrating the statistics file
▶ DMCAUTH utility for migrating existing the Authorization file

These two file migration utilities have required variables that you must define. The following table defines each of the required variables for the file migration utilities.

Variable	Definition
oldmcat	Indicates the VSAM Master Catalog DSName for the older version of CONNECT:Direct for VM.
oldmuid	Indicates the VSAM userid name of the older version of CONNECT:Direct for VM.
oldmdev	Indicates the VSAM minidisk of the older version of CONNECT:Direct for VM.
oldhlq	Indicates the VSAM high-level qualifier of the older version of CONNECT:Direct for VM.
loadlib	Indicates the LOADLIB name of CONNECT:Direct for VM/ESA Version 3.2.00.
mastcat	Indicates the VSAM Master Catalog DSName for CONNECT:Direct for VM/ESA Version 3.2.00.
hlq3.200	Indicates the VSAM high-level qualifier for CONNECT:Direct for VM/ESA Version 3.2.00.

Migrating Existing VM Statistics File

This section provides a brief overview of how the DMSTCONV utility works and steps on how to use the DMSTCONV utility to migrate the existing VM statistics file.

Understanding the DMSTCONV Utility

The DMSTCONV utility reads the records from the old format relative record cluster in chronological order starting with the oldest. The DMSTCONV utility writes the records to the series of ESDS clusters in the statistics file pair list. As DMSTCONV writes each ESDS record, it also writes any necessary index information to the associated KSDS.

When a file pair fills, DMSTCONV closes it and begins writing records to the next pair in the list. If all the file pairs fill before exhausting the records from the old log, then the file pair list *wraps* as it does in normal processing. The program overlays old records in the first file pair with newer records.

When processing completes, the last file pair written is set up as the active pair when the DTF is brought up using the same file pair list.

Using the DMSTCONV Utility to Migrate Statistics File Pairs

Complete these steps for running the DMSTCONV utility.

1. Edit member STATCONV EXEC found on the GCS virtual machine minidisk.

    Note: The EXEC assumes two file pairs. Modify the EXEC to include additional file pairs and set any EXEC variable values appropriate to your environment.

    ---

2. Code the ESDSxx and KSDSxx DD statements to refer to the desired output file pairs.

3. Enter the following command:

    ```
    STATCONV
    ```

Migrating Existing Authorization Files

The section provides a brief overview of how the DMCAUTH utility works and steps on how to use the DMCAUTH utility to migrate existing VM Authorization files.

Understanding the DMCAUTH Utility

CONNECT:Direct for VM/ESA has changed the format for the Authorization file from previous versions of CONNECT:Direct for VM. Use the DMCAUTH utility to migrate your existing Authorization files. The DMCAUTH utility performs these tasks:

▶ Reads the old format Authorization file records
▶ Converts these files to the new format
▶ Writes the converted files to the new Authorization file

Use the AUTHCONV EXEC found on the GCS virtual machine minidisk to execute the DMCAUTH utility.

Using the DMCAUTH Utility to Migrate Existing Authorization Files

Perform the following steps to run the DMCAUTH utility:

1. Edit the AUTHCONV EXEC found on the GCS virtual machine minidisk to set any variable values appropriate to your environment.

2. Verify that the 3.2.00 Authorization file is an empty VSAM data set.

   Note: You must delete and redefine the 3.2.00 Authorization file if there are existing records in the data set.

3. Enter the following command:

```
AUTHCONV
```

Step 7 — Install the IUI (Optional)

The Interactive User Interface (IUI) provides a method of command entry through fill-in-the-blank screens for easier administrator control. The IUI allows you to communicate to CONNECT:Direct for VM/ESA using an ISPF/PDF or ISPF/DIALOG system.

Note: ISPF/DIALOG system does not have the browse and edit features of the ISPF/PDF system.

You must complete the following steps to install the IUI:

1. Use the CDISPF utility to modify the appropriate source member contained within the ISPF library. This table identifies the source member and the ISPF library that contains the source member for each ISPF systems.

System	Source Member	IPSF Library
ISPF/PDF	ISR@PRIM	ISRPLIB MACLIB
ISPF/DIALOG	ISP@PRIM	ISPPLIB MACLIB

2. Modify the installation EXEC used to bring up either the ISPF/PDF or ISPF/DIALOG system. This EXEC is usually the IBM-supplied ISPF EXEC or ISPSTART EXEC.

You may want to contact the systems programmer responsible for the ISPF environment to make these changes.

WARNING: If your Primary Options Menu contains local modifications which make it different from the IBM version, you must re-install the local modifications after installing the IUI. CONNECT:Direct for VM/ESA reserves the selection values for **M** and **N** on the Primary Options Menu.

Understanding the Changes to the Primary Options Menu

CONNECT:Direct provides you with the CDISPF utility so you can modify the IBM-supplied Primary Options Menu for ISPF. The modifications to this menu consists of adding four lines to allow users to gain access to CONNECT:Direct for VM/ESA.

You must make these modifications to the Primary Option Menu for ISPF:

▶ **M** lets you display messages in the CONNECT:Direct message file.

▶ **N** connects you to the CONNECT:Direct service machine and supports all CONNECT:Direct functions.

ISR@PRIM and ISP@PRIM are the source for the Primary Options Menu as provided by IBM. This source is provided as a maclib member within either ISRPLIB MACLIB or ISPPLIB MACLIB.

Running the CDISPF Utility

This section describes how to modify the appropriate source member contained within the ISPF library and how to modify the installation EXEC used to bring up either the ISPF/PDF or ISPF/DIALOG system.

Modifying the Source Member

You must modify the source member before you modify the installation EXEC as described in the *Modifying the Installation EXEC* section beginning on page 6-23. To use the CDISPF utility to modify the appropriate source member contained within the ISPF library, you must complete the following steps:

1. Logon to any CMS userid that is intended to have access to CONNECT:Direct for VM/ESA and IPL CMS. You can use the CONNECT:Direct server userid (with CMS IPL'ED) to logon.

2. Link and access the mdisk that contains the CONNECT:Direct installation files. This is the mdisk used as output for the CDSETUP utility which was used to download the installation tape.

3. Run the CDISPF utility.

   The CDISPF utility clears the screen and requests user information as it runs. A description of each CDISPF prompt follows.

   From the CMS command line, enter the following:

   ```
   CDISPF
   ```

   The CDISPF utility initially prompts you for the name/userid that owns the mdisk where the ISPF product has been installed. Enter either the VM/ESA userid of the ISPF service machine or press **Enter** to accept the default of ISPVM.

   The prompt appears as follows.

   ```
   ENTER THE NAME/USERID OF THE ISPF SERVICE MACHINE (DEFAULT: ISPVM)
   ```

   The CDISPF utility next prompts you for the minidisk owned by the ISPF service machine. This minidisk contains the installation and operational material as distributed by IBM. The default value is 192. In this example of the prompt, XXXX is the userid you supplied in the previous prompt.

```
ENTER THE BASE PROGRAM MDISK FOR XXXX (DEFAULT: 192)
```

The CDISPF utility next prompts you for the type of ISPF system to install. There is no default value. Enter either a **1** for ISPF/PDF or a **2** for ISPF/DIALOG.

Note: If there is no ISPF/PDF installed, you should enter a **2**.

The prompt appears as follows.

```
ENTER '1' FOR ISPF/PDF OR '2' FOR ISPF/DIALOG
```

The CDISPF utility next prompts you to enter the VM/ESA userid for the primary CONNECT:Direct DTF server machine. This value is the 'DTF' running under GCS VM/ESA userid. There is no default value. The prompt appears as follows.

```
ENTER THE VM USERID OF YOUR CONNECT:DIRECT SERVER
```

The final prompt from the CDISPF utility is the name of the high-level VSAM qualifier used for the XXXX.NETWORK file. A default value for the netmap name will appear as indicated by XXXX in the following display example.

```
THE DEFAULT NETMAP HIGH-LEVEL QUALIFIER NAME IS: XXXX
TO ACCEPT DEFAULT PRESS ENTER—TO OVERRIDE TYPE A NEW VALUE
```

Note: Input only the XXXX filename, not the complete file name. If the default filename is correct in the prompt, press enter to accept the default. If the default filename is not correct, you must type in a new value.

The CDISPF utility finishes a successful installation with a completion message and a return code of zero. The CDISPF utility writes a new CONNECT:Direct for VM/ESA version of either the ISR@PRIM PANEL or the ISPPRIM PANEL onto your 'A' mdisk.

To make the new ISR@PRIM PANEL or ISP@PRIM PANEL available to all ISPF users, you must do the following:

1. Move the file onto the ISPF program disk or a common mdisk.

2. Change the access FM on the file definition statement to reflect its new location.

You may want to validate your local ISPF configuration with the ISPF System Programmer.

Modifying the ISPF EXEC

After completing the steps listed in the *Modifying the Source Member* section beginning on page 6-21, your 'A' minidisk will contain a CMS file of the new CONNECT:Direct for VM/ESA version of either the ISR@PRIM PANEL or the ISPPRIM PANEL.

You must modify the installation EXEC used by your site to start ISPF. When you start the ISPF, the modified EXEC locates and uses the CMS file on the 'A' minidisk to display the modified version of the Primary Options Menu file.

For ISPF/DIALOG systems, you must add this file definition statement:

```
FILEDEF ISPPLIB DISK ISPNULL PANEL * (PERM CONCAT
```

For ISPF/PDF systems, you must add this file definition statement:

```
FILEDEF ISPPLIB DISK ISRNULL PANEL * (PERM CONCAT
```

Note: Most installations will position the file definition statement shown previously ahead of any other ISPPLIB file definition statements already in the file.

You must add the following file definition statements regardless of which ISPF system your site uses.

```
FILEDEF ISPPLIB DISK CDPLIB MACLIB * (PERM CONCAT
FILEDEF ISPMLIB DISK CDMLIB MACLIB * (PERM CONCAT
FILEDEF ISPLLIB DISK CDV3200 LOADLIB * (PERM CONCAT
```

Two additional links are needed for IUI access:

▶ A link to the VSAM 195 which is accessed as B.

▶ A link to the DTF 191 disk.

Note: If ISR@PRIM remains on this disk, the access FM will need to be before the ISPF program disk FM.

Step 8 — Select Initialization Parameters

CONNECT:Direct at start-up processes the initialization parameters which are customized by the user as required.

Using the SAMPLE INITPARM file

Use the DMGCSMAP STARTUP file to specify the SAMPLE INITPARM file to your CONNECT:Direct system.

```
DMGCSMFP
DMGCSAPI VSAM.FM=B
OSRUN DMINIT PARM='SAMPLE INITPARM A'
```

You must modify the SAMPLE INITPARM member, which contains an example of the initialization parameters, from the CONNECT:Direct distribution tape before starting CONNECT:Direct.

You must specify the NDM.NODE and NDM.KEY parameters for CONNECT:Direct to initialize properly. Find the values for these parameters in the documentation shipped with the CONNECT:Direct installation package.

Review the following table for a list of updates required within the SAMPLE INITPARM file:

Values	Definition
UUUUUU	Indicates the VM CONNECT:Direct Console Operator ID.
PPPPPP	Specifies the password (super password).
XXXXXX	Indicates the VSAM high-level qualifier.
ZZZZZZ	Specifies the production key (NDM.KEY).
TTTTTT	Indicates NDM.NODE values.

Overrides for initialization parameters can be specified on the OSRUN statement contained within the DMGCSMAP STARTUP file provided.

Note: You can use the overrides to initially test CONNECT:Direct and determine system problem instead of editing the initialization parameter file.

VM limits the value specified for a parameter to 100 characters. An example follows.

```
OSRUN DMINIT PARM='SAMPLE INITPARM A  QUIESCE=YES'
```

Specify the Initialization Parameters and the VSAM Initialization Parameters

Specify all initialization parameters for the CONNECT:Direct and VSAM initialization parameters.

Refer to the *Initialization Parameters* appendix in the CONNECT:Direct for VM/ESA Administration Guide for complete definitions of the CONNECT:Direct and VSAM initialization parameter.

Step 9 — Start CONNECT:Direct

The Master Application Program allows you to run the GCS API, the CONNECT:Direct Message File Program, and the CONNECT:Direct for VM/ESA DTF as subtasks in the same GCS virtual machine.

The GCS API and Message File Program provide VTAM and VSAM support, respectively, to the CMS user.

To set up the Master Application Program to run automatically, complete these steps.

1. Rename the PROFILE $GCS file to PROFILE GCS.

2. Log on to the DTF service machine, IPL CMS, and edit PROFILE GCS. The following example shows the format of the GCS commands that must be included in the PROFILE GCS file.

```
/* Profile for running C:D VMESA in the DTF server userid      */
'CP SET EMSG OFF'
'CP SET IMSG OFF'
/CP SPOOL CONS * CLASS T START'

/* ************************************************************ */
/* link and access the CD:VMESA program mdisk if it is NOT your */
/* DTF servers 'A' mdisk.                                       */
/* ************************************************************ */
/* 'CP LINK userid xxxx xxxx RR                                 */
/* 'ACCESS xxxx C/A'                                            */

/* ************************************************************ */
/* link and access the CD:VMESA VSAM mdisk (usually the 195)   */
/* ************************************************************ */
/* 'CP LINK userid xxxx xxxx RR (if mdisk not owned by server) */
 'ACCESS 195 B'

/* ************************************************************ */
/* VSE/VSAM commands required for C:D VMESA server             */
/* NOTE 1: you must specify the name of your VM master         */
/* catalog if it is not the default MASTCAT                    */
/* NOTE 2: you must specify the name of your VSAM high level   */
/* qualifier if it is not correct for your site.               */
/* ************************************************************ */
'DLBL IJSYSCT B DSN MASTCAT (PERM '
'DLBL DMNETMP B DSN CDVM32.NETMAP (VSAM PERM '
'DLBL DMMSGFL B DSN CDVM32.MSG (VSAM PERM '

/* ************************************************************ */
/* issue VMESA command to elect the CD:VMESA program library  */
/* ************************************************************ */
'GLOBAL LOADLIB CD3200'

/* ************************************************************ */
/* execute the supplied GCS exec to memory load CD:VMESA      */
/* selected modules for performance considerations            */
/* ************************************************************ */
'EXEC GCSCMD'

/* ************************************************************ */
'DMGCSMAP'    /*MASTER APPLICATION PROGRAM - ATTACHES OTHERS */
EXIT;
```

A description of the GCS commands follows.

SPOOL

starts the console log, which is a diagnostic tool. All output displayed on the console is also sent to a spool file. To close the spool file, issue the following command:

```
CP CLOSE CONS
```

LINK

(using the first LINK command) links the user to the disk containing the CONNECT:Direct for VM/ESA load library. Change *userid, ccuu1, ccuu2,* and *password* to the proper values.

The second LINK command links the user to the VSAM
disk containing the CONNECT:Direct VSAM files.
Change *userid, ccuu3, ccuu4,* and *password* to the proper
values.

Note: The following CCUUs must not be used for the
value of *ccuu2* or *ccuu4*. These CCUUs are
reserved for CONNECT:Direct for VM/ESA use.

060 – 0FF	(used for punches)
F01 – FF1	(used for tape)
4C1 – 4E9	(used for disks)

ACCESS

(using the first ACCESS command) allows access to the
disk containing the CONNECT:Direct for VM/ESA load
library. Change *ccuu2* and *mymode* to the proper values.

The second ACCESS command allows access to the
VSAM disk containing the CONNECT:Direct VSAM files.
Change *ccuu4* and *vsammode* to the proper value.

DLBL

defines the VSAM catalog containing CONNECT:Direct
VSAM files. The ddname for the catalog is IJSYSCT.
Change *vsammode* and *mastcat* to the proper values.

The second and third DLBL commands define the VSAM
files used by CONNECT:Direct. Change *vsammode,
mastcat, hilq.netmap,* and *hilq.msg* to the proper values.

GLOBAL

defines the load libraries to be searched for modules.

SET EMSG

specifies that both error codes and error text are
displayed at the terminal.

GCSCMD

is the Sterling Commerce GCS EXEC that performs
LOADCMD commands, which define CONNECT:Direct
for VM/ESA programs to be executed as commands.

DMGCSMAP

runs the Master Application Program (DMGCSMAP).
The Master Application Program runs the GCS API, the
CONNECT:Direct Message File Program, and the
CONNECT:Direct for VM/ESA DTF as subtasks.

3. Edit the DMGCSMAP STARTUP file, if necessary.

   To cause the Master Application Program to automatically start up the GCS API, the CONNECT:Direct Message File Program, and the CONNECT:Direct DTF, create a file, DMGCSMAP STARTUP A, that contains the following commands.

```
DMGCSMFP
DMGCSAPI VSAM.FM=vsammode
OSRUN DMINIT PARM='SAMPLE INITPARM A'
```

Note: Comments are not allowed in the DMGCSMAP STARTUP file. Blank spaces are not allowed on either side of the equal signs (=) in the DMGCSAPI command string or the OSRUN DMINIT command string.

The following list explains the commands and parameters used in the DMGCSMAP STARTUP file:

DMGCSMFP

   starts the CONNECT:Direct Message File Program.

DMGCSAPI

   starts the GCS API. Change *vsammode* to the filemode of the VSAM disk that contains the CONNECT:Direct VSAM files.

OSRUN

   starts the CONNECT:Direct DTF. To run the CONNECT:Direct DTF in a separate GCS machine, issue the OSRUN command at the GCS machine where the CONNECT:Direct DTF will run.

   Note: Specify overrides for initialization parameters in the PARM section of the OSRUN command instead of editing an initialization parameter file. OSRUN DMINIT will not execute properly if there are blank spaces on either side of the equal signs. The CONNECT:Direct DTF will not initialize and the user may not see any error messages.

The following list explains the commands and parameters used in the DMGCSMAP STARTUP file:
Each program in the DMGCSMAP STARTUP file automatically starts when the Master Application Program is started and will issue a write-to-operator with reply (WTOR) message when initialization completes.

The Master Application Program provides you with the ability to run multiple programs under one GCS virtual machine. The following run under the Master Application Program:

▶ CONNECT:Direct Message File Program
▶ GCS API
▶ CONNECT:Direct DTF

The following WTORs appear:

```
00 DMGCSMAP : S(tart a Command) D(isplay Command Status) E(ndDMGCSMAP)?
01 DMGCSMFP: S(ession status) Q(uiesce) U(nquiesce)  E(ndDMGCSMFP)?
02 DMGCSAPI:  S(ession status) Q(uiesce) U(nquiesce)  E(ndDMGCSAPI)?
03 SOIA000A — ENTER COMMAND:
```

Use the following command to validate WTORs.

```
Q REPLY
```

For the DMGCSMFP and DMGCSAPI WTORs, a reply of:

▶ S: Displays all userids with active IUCV sessions to the Message File Program and the GCS API
▶ Q: Quiesces the program
▶ U: Unquiesces or resumes the program
▶ E: Ends the program

For the DMGCSMAP WTOR, a reply of:

▶ S: Followed by any valid GCS command, will cause that command to be executed in a subtask attached by DMGCSMAP
▶ D: Displays each command status
▶ E: Ends the DMGCSMAP program

For the SOIA000A WTOR, reply with any CONNECT:Direct for VM/ESA Operator Command CLIST provided. A complete discussion of these CLISTs is in the *CONNECT:Direct Console Operator's Guide* in the *CONNECT:Direct for VM/ESA Operator Interface* chapter.

4. IPL GCS.

The following messages are issued for CONNECT:Direct for VM/ESA once you have IPLed GCS:

```
00 DMGCSMAP : S(tart a Command) D(isplay Command Status) E(ndDMGCSMAP)?
01 DMGCSMFP:  S(ession status)  Q(uiesce)  U(nquiesce)   E(ndDMGCSMFP)?
02 DMGCSAPI:  S(ession status)  Q(uiesce)  U(nquiesce)   E(ndDMGCSAPI)?
SITA001I CONNECT:Direct for VM/ESA INITIALIZATION BEGUN.
SITA002I CONNECT:Direct PARAMETER FILE ALLOCATED AND OPEN.
SITA022I LOADING CONNECT:Direct MODULES.
SITA523I SECURITY INITIALIZED.
SITA023I INITIALIZING CONNECT:Direct STORAGE.
SITA024I BUILDING THE TCA CHAIN.
SITA026I CREATING THE SYSTEM TASKS (MASTER AND TIMER).
SITA025I BUILDING THE ECB ADDRESS LIST.
SITA027I BUILDING THE DDN TABLE.
SITA067I MESSAGE FILE IS OPEN.
SITA068I AUTHORIZATION FILE IS OPEN.
SITA134I TYPE DEFAULTS FILE IS OPEN.
SITA069I NETWORK MAP FILE IS OPEN.
SITA074I CHECK POINT FILE IS OPEN.
SITA030I PCQ/TCQ BEING BUILT.
SITA029I STATISTICS FACILITY BEING INITIATED.
SSTL026I Statistics File Pair 01 is now active.
SSTL019I Statistics facility successfully initialized.
SITA034I VTAM SERVICES BEING INITIATED APPLID = xxxxxxx.
SITA036I CONNECT:DIRECT FOR VM/ESA 3.02.00 PUF 0 INITIALIZATION
         COMPLETE.
STCP015I IBM TCP/IP IUCV INTERFACE INITIALIZING.
STCP022I IBM TCP/IP INITIALIZING IUCV INTERFACE.
STCP023I IBM TCP/IP CONNECTING TO TCP THROUGH IUCV TO TCPIP
STCP024I IBM TCP/IP INITIALIZING TCP/IP INTERFACE
STCP021I IBM TCP/IP IUCV INTERFACE INITIALIZED
STCP001I IBM TCP/IP ISSUING SOCKET CALL
STCP002I IBM TCP/IP ISSUING BIND SOCKET
STCP019I IBM TCP/IP PORT NUMBER IS 1364
STCP003I IBM TCP/IP CONNECT:DIRECT NOW LISTENING FOR CONNECTIONS
03 SOIA000A —  ENTER COMMAND:
```

Note: In the previous example, the TCP initialization parameter was specified as TCP=YES.

The SOI000A (WTOR) message means that CONNECT:Direct is up.

5. Enter this command to disconnect the DTF service machine, but do not log off.

```
#CP DISC
```

6. Log on to a CMS machine and initialize the CONNECT:Direct authorization file and the CONNECT:Direct type defaults file.

To initialize these files, edit the BATCH EXEC and BATCH INPUT distribution files as shown in the comment section of each file. Once the files have been edited, issue the following command from CMS:

```
BATCH BATCH INPUT fm
```

In this example, *fm* is the mode of the disk where the BATCH INPUT file is located.

Note: You must be outside ISPF to issue this command.

Step 10 — Sign On to CONNECT:Direct

This step explains how to sign on to CONNECT:Direct using the following four methods:

▶ IUI
▶ Operator Interface
▶ Batch Interface
▶ CMS Command Line Interface

Signing on Through the IUI

Select the CONNECT:Direct option by means of the ISPF Primary Option menu. Sign on to CONNECT:Direct.

Note: You may need to verify with the system administrator your userid and password.

The ISPF Primary Options panel now displays the following options.

```
M  C:D.MSG        — Display C:D messages
N  C:D.SERVER     — CONNECT:Direct SERVER
```

Signing on Through the Operator Interface

You must specify the MCS.SIGNON and MCS.CLIST initialization parameters in the SAMPLE INTPARM file to use the operator interface. For more information, see the *CONNECT:Direct for VM/ESA Operator Interface* chapter in the *CONNECT:Direct Console Operator's Guide.*

Signing on Through the Batch Interface

For instructions on using the batch interface, refer to the *Issuing Commands Through the Batch Interface* chapter in the *CONNECT:Direct for VM/ESA User's Guide.*

Signing on Through the CMS Command Line Interface

> When using the CMS Command Line Interface, the user is automatically signed on to CONNECT:Direct when the first command is issued. For more information see the *Issuing Commands Through the CMS Command Line Interface* chapter of the *CONNECT:Direct for VM/ESA User's Guide.*

Step 11 — Run Sample Processes

To gain experience with CONNECT:Direct operation, you can run the sample Processes on the distribution tape, which have the filetype NDMPROC. These Processes require some customization before they can be submitted.

If you have the IUI, you can use the general purpose submit screen to modify any Process parameters used during Process submission. The following table describes the sample processes.

Process	Purpose
PROC01	VSAM to SAM example.
PROC02	SAM to SL TAPE example.
PROC03	SAM to VSAM example.
PROC04	Run Task and DMNOTIFY example.
PROC05	Submit within a Process example.
PROC06	VSAM to VSAM example.

Tuning CONNECT:Direct for VM/ESA for Performance

After you have CONNECT:Direct for VM/ESA installed and functional, you can tune it to enhance the performance and usage of resources. Actual resourse usage by CONNECT:Direct varies according to the implemented configuration.

You can tune CONNECT:Direct for VM/ESA based upon the guidelines provided to assist you in estimating the impact of CONNECT:Direct in your environment for:

▶ Statistics file pair usage
▶ Shared File System usage

Tuning the Statistics Files

This section describes how to determine the most efficient use of your Statistics file space. The amount of activity in the CONNECT:Direct system determines how frequently the statistics file pair list wraps around.

Estimate your activity rate and allocate enough space so that the system records several days of records before a wrap-around occurs. After running CONNECT:Direct for several days, you may need to adjust the number or sizes of the statistics files to accomplish this.

Using the SCCSTAT Utility to Determine File Usage

Use the SCCSTAT utility to find out the rate at which the system generates statistics records. SCCSTAT also performs an analysis of the contents of the statistics file showing what percentage of the records are of each record type. This utility is executed by the SCCSTAT EXEC or the SCCSTAT1 EXEC to analyze a single statistics entry-sequenced file.

To help with your estimates of statistics file usage, CONNECT:Direct provides the utility programs SCCSTAT1 and SCCSTAT.

▶ SCCSTAT1 reports on the contents of the statistics file for previous versions of CONNECT:Direct for VM.

▶ SCCSTAT reports on the statistics files for Version 3.2.00. It calculates the average number of CIs used per day at one DTF.

SCCSTAT1 Utility

Use the SCCSTAT1 EXEC to report on the contents of the statistics file for CONNECT:Direct versions prior to Version 3.2.00.

Note: Sterling Commerce recommends that you always run with an ESTAE file definition if you need to use the SCCSTAT1 utility.

Perform the following steps to use the SCCSTAT1 utility:

1. Set the required variables to run the EXEC. The following table defines each of the required variables for the SCCSTAT1 utility.

Variable	Definition
oldmcat	indicates the VSAM Master Catalog DSName of the prior version of CONNECT:Direct for VM
olduid	indicates the VSAM userid name of the prior version of CONNECT:Direct for VM
oldmdev	indicates the VSAM minidisk for the prior version of CONNECT:Direct for VM
oldhlq	indicates the VSAM qualifier for the prior version of CONNECT:Direct for VM
mastcat	indicates the VSAM Master Catalog of CONNECT:Direct for VM/ESA Version 3.2.00
loadlib	indicates the LOADLIB name of CONNECT:Direct for VM/ESA Version 3.2.00

2. Run an analysis of the pre-3.2 STATS file. To analysis of these files, perform these steps:

   a. Access the pre-3.2 CONNECT:Direct Master Catalog with the following command:

   ```
   'RELEASE V'
   'CP LINK' old userid oldmdev as 295 RR'
   'ACCESS 295 V'
   'DLBL IJSYSCT V DSN' msatcat (VSAM'
   ```

   b. Set LOADLIB name for the CONNECT:Direct for VM/ESA Version 3.2 program library with the following command:

   ```
   'GLOBAL LOADLIB' loadlib
   ```

   c. Execute the STATS file analysis program with this command:

   ```
   'OSRUN SCCSTAT1'
   ```

3. Reset VSAM Master Catalog DLBL. Use this command resets the Master Catalog.

   ```
   'RELEASE V(DET'
   'ACCESS 195 V'
   'DLBL IJSYSCT V DSN mastcat'(VSAM'
   ```

SCCSTAT Utility

Use the SCCSTAT EXEC to report on the contents of the statistics file for CONNECT:Direct versions prior to Version 3.2.00.

Note: Sterling Commerce recommends that you always run with an ESTAE file definition if you need to use the SCCSTAT utility.

The following command runs with an ESTAE file definition.

```
'FILEDEF ESTA DUMMY'
```

Set SYSOUT output to the terminal with the following command:

```
'FILEDEF SYSOUT TERMINAL'
```

Perform the following steps to use the SCCSTAT utility:

1. Set the required variables to run the EXEC. The following table
   defines each of the required variables for the SCCSTAT utility.

Variable	Definition
hlq3200	indicates the VSAM high-level qualifier for CONNECT:Direct for VM/ESA Version 3.2.00
loadlib	indicates the LOADLIB name of CONNECT:Direct for VM/ESA Version 3.2.00
mastcat	indicates the VSAM Master Catalog of CONNECT:Direct for VM/ESA Version 3.2.00

2. Perform the following steps to run analysis of the Version 3.2 STATS
   file ESDS.

   a. Access the Master Catalog with the following command:

   ```
   'DLBL IJSYSCT V DSN' mastcat'(VSAM'
   ```

   b. Set the DLBL for STATS file ESDS with the following command:

   ```
   'DLBL ESDS V DSN' hlq3200'.STATS.ESDS01 (VSAM '
   ```

   c. Set the LOADLIB name for the CONNECT:Direct for VM/ESA
   Version 3.2.00 program library with this command:

   ```
   'GLOBAL LOADLIB' loadlib
   ```

   d. Execute the STATS file analysis program with the following
   command:

   ```
   'OSRUN SCCSTAT'
   ```

Statistics Files Space Allocation Example

In this example, the CONNECT:Direct software was installed using the default statistics installation of two file pairs with a total capacity of 13,500 records. After running CONNECT:Direct for a time, the user determines that this allows the logging of records for about 2.5 days before the file pair list wraps. The administrator wants to provide space for 7 days worth of records to be available at any given time. The administrator does the following:

1. Use SCCSTAT to determine the number of records written daily.

   Run the SCCSTAT utility against the statistics entry-sequenced clusters to determine the rate at which the system generates the statistics records.

   For example, SCCSTAT shows that records per day is approximately equal to 5,400.

2. Determine the total capacity of the statistics file.

   ```
   capacity = (records per day) * days
   ```

   Determine the total capacity in this example by multiplying the 5,400 records per day by 7 days. In this case, the total capacity of the statistics file is 37,800 records.

3. Determine the number of records per file pair.

   In this example, the administrator decides to define four file pairs, so each will be given a capacity of 9,500 records, for a total capacity of 38,000 records.

4. Determine the RECORDS parameter value for the key-sequenced clusters.

   ```
   KSDS-records = 0.75 * (ESDS-records)
   ```

   Determine the number of KSDS records by multiplying 75% by 9,500, the number of records per file pair. The RECORDS parameter value for the key-sequenced clusters is 7,125.

Based on these calculations, the administrator allocates four file pairs. The entry-sequenced cluster (ESDS) of each pair is defined with RECORDS(9500). The key-sequenced clusters (KSDS) are defined with RECORDS(7125).

Refer to the *CONNECT:Direct for VM/ESA Administration Guide* for more information on statistics administration.

Tuning the SFS Server

This section describes how to tune your SFS server to enhance its performance and availability.

Guidelines for Tuning the SFS Server

The implementation of the SFS Server allows for the flexibility for up to 256 DTF regions to use the same SFS Server. The SFS Server is written using the CMS multi-tasking facilities. In addition, SFS Server can take advantage of multiple virtual CPUs for multiple threads. The SFS Server starts a new thread for each file that is copied.

Follow these guidelines to ensure efficient use of the SFS Server:

1. Use the OPTION QUICKDSP to ensure that the CPU resources are made available to the SFS Server.

2. Use the same performance options and SHARE that the DTF virtual machine is given because the SFS Server is an extension of the DTF.

Installation Worksheets

Use the worksheets in this appendix to help you plan and complete the installation set up your CONNECT:Direct for VM/ESA node.

Planning Your Installation

Read your *CONNECT:Direct for VM/ESA Release Notes* and complete the following information.

Verify Your Hardware and Software Requirements:

	Available	Required
Authorized Library	_____	_____
DASD	_____	_____
Tape Drive	_____	_____
VM/ESA Operating System		
VM/ESA	_____	_____
VTAM Release	_____	_____
VSAM	_____	_____
Dynamic Partition	_____	_____
GCS	_____	_____

SNA Network Requirements (LU6.2)

  VTAM   _____   _____

  IBM Advanced Communications Functions for Network Control Program (NCP)

     _____   _____

SNA Network Requirements (Non-LU6.2)

  VTAM   _____   _____

  IBM Advanced Communications Functions for Network Control Program (NCP)

     _____   _____

Plan Your DASD Requirements

VSAM Files Space Requirements

     _____   _____

Data Transmission Facility (DTF) Virtual Storage Requirements

     _____   _____

IUI Virtual Storage Space Requirement

     _____   _____

Shared File System Storage Requirement

     _____   _____

Prepare Your VTAM Definitions

APPLID of Local DTF   _____

APPLID for IUI and Batch Sessions   _____

APPLID for Loop-Back Processing   _____

APPLID for MS-DOS   _____

Logmode Table Entries   _____

Cross-Domain Resource Manager Minor Nodes   _____

Cross-Domain Resources   _____

Multiple DTFs on Same Machine or VTAM   _____

Network or Domain Name   _____

Prepare VTAM and NCP Parameters

Select RU Size for SNA Sessions   _____

Determine Effects of NCP Parameters   _____

Information Required During Installation

Compile the following information on this worksheet before you begin your installation. This information will be required during the installation:

Logical node name for local CONNECT:Direct node _____

VTAM DTF APPLID for local CONNECT:Direct node _____

Logical node names for adjacent nodes (remote nodes) _____

VTAM DTF APPLIDS for adjacent nodes (remote nodes) _____

Interactive APPLIDS (for all nodes) _____

Installation Steps Worksheet

This worksheet follows the installation process as explained in the *Installing CONNECT:Direct for VM/ESA* chapter beginning on page 6-1.

Step 1–Configure the GCS Virtual Machine for CONNECT:Direct

Use the information you gathered for the *Information Required During Installation* worksheet to complete this step of the installation.

Step 2–Configure the SFS Virtual Machine for CONNECT:Direct (Optional)

Proceed to the next step if you do not want to use the Shared File System (SFS). Use the information you gathered for the Information Required During Installation worksheet to complete this step of the installation.

Step 3–Download Tape and Configure VSAM mdisk

Use the CDSETUP utility for a quick installation.

Verify that you meet the following requirements for formatting a minidisk as a DOS VSAM disk:

A 3380 disk drive _____

The minidisk to be formatted is 20 cylinders _____

The ccuu of the minidisk is 195 _____

The volume label of the disk is CDVSAM _____

CDVSAM is accessed as the V disk _____

Step 4–Prepare VTAM Definitions

Prepare the following definitions:

APPLID (local node) _____

APPLID (PNODE=SNODE processing) _____

Example definitions (NDMAPPL and NDMIAPPL) are source members of type V.

APPLIDs (interactive and batch users on the same node)

_____ _____ _____ _____

_____ _____ _____ _____

_____ _____ _____ _____

Mode table entry (local node) _____

Example definitions (NDMMODET) are source members of type V.

Mode table entries (one for each type of connection)

_____ _____ _____ _____

_____ _____ _____ _____

_____ _____ _____ _____

Installation Steps Worksheet

Step 5–Run CDINSTAL

Which of the following DASD devices will you be using? (circle one)   3380   3390   9335

Refer to the instructions beginning on page 6-9 for details on how to install on a 3380, 3390, and 9335 DASD device.

Use the Network Map Definitions Worksheets in this appendix to prepare your adjacent node definitions.

Step 6–Migrating CONNECT:Direct for VM Files (Existing Users Only)

Skip this step if you are a new customer or you do not want to migrate your existing files.

What Version of CONNECT:Direct for VM are you currently running? _____

Step 7–Install the IUI (Optional)

Contact the system programmer responsible for the ISPF environment to make these changes.

Step 8–Select Initialization Parameters

The NDM.NODE and NDM.KEY parameters are required for proper initialization of the CONNECT:Direct for VM/ESA software.

NDM.NODE _____

NDM.KEY_____

Initialization parameter for VSAM files are also required. Refer to DEFVSAM.PROC which you loaded in Step 2.

For initial installation and testing, specify SECURITY=OFF until you install a security system.

Step 9–Start CONNECT:Direct

Use the start-up job stream NDM.PROC in the distribution library.

Step10–Sign On to CONNECT:Direct

Which method will you use to sign on to CONNECT:Direct?

     _____ Batch Interface

     _____ Interactive User Interface

     _____ CMS Command Line Interface

     _____ Operator Interface

Step 11–Run Sample Processes

Which of the following sample Processes do you need to see?

_____ VSAM to SAM

_____ SAM to SL TAPE

_____ SAM to VSAM

_____ Run Task and DMNOTIFY

_____ Submit within a Process

_____ VSAM to VSAM

Refer to the table on page 6-32 which describes where you can locate the examples.

Tuning CONNECT:Direct for VM/ESA (optional)

After you have CONNECT:Direct for VM/ESA installed and functional, you can enhance the performance and usage of resources by tuning the following:

_____ Statistics File Pairs

_____ CMS File Request Server

Refer to the *Tuning CONNECT:Direct for VM/ESA for Performance* chapter beginning on page 7-1 for detailed instructions on how to enhance the performance of statistics file pairs and the CMS File Request Server.

Running the TAPEEXIT (Optional)

Do you want to use the Sterling Software, Inc. tape management software product VMTAPE? _____

The *CONNECT:Direct for VM/ESA Tape Exit* appendix describes the necessary steps for using VMTAPE with CONNECT:Direct for VM/ESA.

Sample VTAM Definitions

Which of the following VTAM definitions do you need to see?

_____ VTAM definitions for a full networking CONNECT:Direct

_____ VTAM definitions for a cross-domain network

_____ VTAM definitions for a mainframe-to-microcomputer connection

Worksheet for Local Node and Generated
Adjacent Node Definitions

Local Node:

The following entries define the local CONNECT:Direct node in the Network Map. Refer to the *CONNECT:Direct for VM/ESA Administration Guide* for information about the syntax, valid parameters, and an example.

Local Node logical node name _____

Communications Name (VTAM APPLID) _____

SUPERUSR password (optional) _____

VTAM APPLIDS for IUI and Batch Sessions (adjacent node definition):

The following entry defines the VTAM APPLIDS for the local CONNECT:Direct to be used for IUI and batch sessions. Refer to page 4-5 for more information.

Interactive VTAM APPLIDS _____

_____

_____

_____

PNODE=SNODE Definitions (adjacent node definition):

The following entries define the adjacent node definition for PNODE=SNODE processing. Reference the *CONNECT:Direct for VM/ESA Administration Guide* for a discussion of PNODE=SNODE processing.

Communications Name (VTAM APPLID) for the Adjacent Node _____

Parallel Sessions Parameter: PARSESS=(max, default) _____

Network Map Adjacent Node Definitions for 370 SNA LU0 Nodes

Use this worksheet for adjacent node definitions for 370 SNA LU0 nodes. Refer to the *CONNECT:Direct for VM/ESA Administration Guide for a* description of adjacent node syntax, valid parameters, and examples. **See the worksheet on page A-12 for Tandem SNA nodes.**

Adjacent Node Name   _____

Communications Name (VTAM APPLID)   _____

Parallel Sessions Parameter: PARSESS=(max, default)   _____

Session Type   **SNA**

Adjacent Node Name   _____

Communications Name (VTAM APPLID)   _____

Parallel Sessions Parameter: PARSESS=(max, default)   _____

Session Type   **SNA**

Adjacent Node Name   _____

Communications Name (VTAM APPLID)   _____

Parallel Sessions Parameter: PARSESS=(max, default)   _____

Session Type   **SNA**

Adjacent Node Name   _____

Communications Name (VTAM APPLID)   _____

Parallel Sessions Parameter: PARSESS=(max, default)   _____

Session Type   **SNA**

Network Map Adjacent Node Definitions for
MS-DOS LU0 Nodes and OS/2

Use this worksheet for adjacent node definitions for MS-DOS and OS/2 LU0 nodes. Refer to the description of adjacent node definitions in the *CONNECT:Direct for VM/ESA Administration Guide.*

Adjacent Node Name

Communications Name (Logical Unit Name)

Adjacent Node Name

Communications Name (Logical Unit Name)

Adjacent Node Name

Communications Name (Logical Unit Name)

Adjacent Node Name

Communications Name (Logical Unit Name)

Adjacent Node Name

Communications Name (Logical Unit Name)

Adjacent Node Name

Communications Name (Logical Unit Name)

Adjacent Node Name

Communications Name (Logical Unit Name)

Adjacent Node Name

Communications Name (Logical Unit Name)

Adjacent Node Name

Communications Name (Logical Unit Name)

Adjacent Node Name

Communications Name (Logical Unit Name)

Network Map Adjacent Node Definitions for OpenVMS Nodes

Use this worksheet for OpenVMS Node Names and associated Logical Unit Pools.

Adjacent Node Name   _____

Parallel Sessions (PARSESS) Default Class   _____

PNODE LU Pool (PNODE.LUS)   _____

                                                          _____

                                                          _____

SNODE LU Pool (SNODE.LUS)   _____

                                                          _____

Adjacent Node Name   _____

Parallel Sessions (PARSESS) Default Class   _____

PNODE LU Pool (PNODE.LUS)   _____

                                                          _____

                                                          _____

SNODE LU Pool (SNODE.LUS)   _____

                                                          _____

Adjacent Node Name   _____

Parallel Sessions (PARSESS) Default Class   _____

PNODE LU Pool (PNODE.LUS)   _____

                                                          _____

                                                          _____

SNODE LU Pool (SNODE.LUS)   _____

                                                          _____

Network Map Adjacent Node Definitions for Tandem SNA Nodes

Use this worksheet for CONNECT:Direct for Tandem Node Names and associated Logical Unit Pools. Refer to the description of adjacent node definitions and examples in the *CONNECT:Direct for VM/ESA Administration Guide.*

Adjacent Node Name

Parallel Sessions (PARSESS) Default Class

LU Pool (SNODE.LUS)

Adjacent Node Name

LU Pool (SNODE.LUS)

Adjacent Node Name

LU Pool (SNODE.LUS)

Network Map Adjacent Node Definitions for OS/400 SNUF Nodes (LU0)

Use this worksheet for CONNECT:Direct for OS/400 Node Names and associated Logical Unit Pools. Refer to the description of adjacent node definitions beginning and examples in the *CONNECT:Direct for VM/ESA Administration Guide.*

Adjacent Node Name _____

Remote Library Name _____

Session Type **SNUF**

Parallel Sessions Parameter: PARSESS=(max, default) _____

LU Pool (SNODE.LUS) _____

_____

_____

Adjacent Node Name _____

Remote Library Name _____

Session Type **SNUF**

Parallel Sessions Parameter: PARSESS=(max, default) _____

LU Pool (SNODE.LUS) _____

_____

_____

Adjacent Node Name _____

Remote Library Name _____

Session Type **SNUF**

Parallel Sessions Parameter: PARSESS=(max, default) _____

LU Pool (SNODE.LUS) _____

_____

_____

Network Map Adjacent Node Definitions for OS/400 LU6.2 Nodes

Use this worksheet for CONNECT:Direct for OS/400 Node Names and associated Logical Unit Pools. Refer to the description of adjacent node definitions and examples in the *CONNECT:Direct for VM/ESA Administration Guide.*

Adjacent Node Name

Communications Name (not on installation panel)

Remote Library Name

Session Type   **LU62**

Parallel Sessions Parameter: PARSESS=(max, default)

Logmode Entry Name (LOGMODE) **- Required**

Operating Environment (ENVIRONMENT) **- Required**   **AS400**

LU Pool (SNODE.LUS)

Adjacent Node Name

Communications Name (not on installation panel)

Remote Library Name

Session Type   **LU62**

Parallel Sessions Parameter: PARSESS=(max, default)

Logmode Entry Name (LOGMODE) **- Required**

Operating Environment (ENVIRONMENT) **- Required**   **AS400**

LU Pool (SNODE.LUS)

Worksheet for Updating Network Map Adjacent Node Definitions: 370 SNA LU6.2 Nodes

Use this worksheet for adjacent node definitions for 370 SNA LU6.2 nodes. There is no installation panel for this type of adjacent node definition. You can define these nodes after your installation is complete. Refer to the description of node definitions and examples in the *CONNECT:Direct for VM/ESA Administration Guide*.

Adjacent Node Name   _____

Communications Name (VTAM APPLID)   _____

Session Type   **LU62**

Parallel Sessions Parameter: PARSESS=(max, default)   _____

Logmode Entry Name (LOGMODE)   _____

Operating Environment (ENVIRONMENT)   _____

APPLIDS   _____

_____

_____

Adjacent Node Name   _____

Communications Name (VTAM APPLID)   _____

Session Type   **LU62**

Parallel Sessions Parameter: PARSESS=(max, default)   _____

Logmode Entry Name (LOGMODE)   _____

Operating Environment (ENVIRONMENT)   _____

APPLIDS   _____

_____

_____

Network Map Adjacent Node Definitions for MVS TCP/IP Nodes

Use this worksheet for adjacent node definitions for MVS TCP/IP nodes. There is no installation panel for this type of adjacent node definition. You can define these nodes after your installation is complete.

Adjacent Node Name

Communications Name (TCP/IP Port Number)

Session Type   **TCP**

Parallel Sessions Parameter: PARSESS=(max, default)

Operating Environment (ENVIRONMENT)   **MVS**

Pacing Parameter: NDMPACE= (#sends, time)

APPLIDS

Adjacent Node Name

Communications Name (TCP/IP Port Number)

Session Type   **TCP**

Parallel Sessions Parameter: PARSESS=(max, default)

Operating Environment (ENVIRONMENT)   **MVS**

Pacing Parameter: NDMPACE= (#sends, time)

APPLIDS

Network Map Adjacent Node Definitions for MS-DOS and OS/2 LU6.2 Nodes

Use this worksheet for the definition of MS-DOS and OS/2 LU6.2 Adjacent Node parameters. There is no installation panel for this type of adjacent node definition. You can define these nodes after your installation is complete.

Adjacent Node Name _____

Communications Name (Logical Unit Name) _____

Remote Library Name **(for OS/2 only)** **CDOS2**

Session Type **LU62**

Logmode Entry Name (LOGMODE) - **Required** _____

Operating Environment (ENVIRONMENT) _____

Adjacent Node Name _____

Communications Name (Logical Unit Name) _____

Remote Library Name **(for OS/2 only)** **CDOS2**

Session Type **LU62**

Logmode Entry Name (LOGMODE) - **Required** _____

Operating Environment (ENVIRONMENT) _____

Adjacent Node Name _____

Communications Name (Logical Unit Name) _____

Remote Library Name **(for OS/2 only)** **CDOS2**

Session Type **LU62**

Logmode Entry Name (LOGMODE) - **Required** _____

Operating Environment (ENVIRONMENT) _____

Network Map Adjacent Node Definitions for UNIX LU6.2 Nodes

Use this worksheet for UNIX LU6.2 Adjacent Node information. There is no installation panel for this type of adjacent node definition. You can define these nodes after your installation is complete.

Adjacent Node Name

Communications Name

Session Type   **LU62**

Parallel Sessions Parameter: PARSESS=(max, default)

Logmode Entry Name (LOGMODE) **– Required**

Operating Environment (ENVIRONMENT)   **UNIX**

Adjacent Node Name

Communications Name

Session Type   **LU62**

Parallel Sessions Parameter: PARSESS=(max, default)

Logmode Entry Name (LOGMODE) **– Required**

Operating Environment (ENVIRONMENT)   **UNIX**

Adjacent Node Name

Communications Name

Session Type   **LU62**

Parallel Sessions Parameter: PARSESS=(max, default)

Logmode Entry Name (LOGMODE) **– Required**

Operating Environment (ENVIRONMENT)   **UNIX**

Network Map Adjacent Node Definitions for NetWare LU6.2 Nodes

Use this worksheet for NetWare LU6.2 Adjacent Node information. There is no installation panel for this type of adjacent node definition. You can define these nodes after your installation is complete.

Adjacent Node Name _____

Communications Name (when specifying a single LU name) _____

Parallel Sessions Parameter: PARSESS=(max, default) _____

Operating Environment Parameter: ENVIRONMENT   **NW**

Session Type   **LU62**

Logmode Entry Name (LOGMODE) **– Required** _____

PNODE LU Pool (PNODE.LUS) _____

_____

_____

SNODE LU Pool (SNODE.LUS) _____

_____

_____

Adjacent Node Name _____

Communications Name (when specifying a single LU name) _____

Parallel Sessions Parameter: PARSESS=(max, default) _____

Operating Environment Parameter: ENVIRONMENT   **NW**

Session Type   **LU62**

Logmode Entry Name (LOGMODE) **– Required** _____

PNODE LU Pool (PNODE.LUS) _____

_____

_____

SNODE LU Pool (SNODE.LUS) _____

_____

_____

Network Map Adjacent Node Definitions for UNIX , NetWare, or Windows NT TCP/IP Nodes

Use this worksheet for UNIX, NetWare, or Windows NT TCP/IP Adjacent Node information. There is no installation panel for this type of adjacent node definition. You can define these nodes after your installation is complete.

Adjacent Node Name

Communications Name

Session Type   **TCP**

Parallel Sessions Parameter: PARSESS=(max, default)

Operating Environment Parameter: ENVIRONMENT   **UNIX** / **NW** / **NT** (circle one)

Adjacent Node Name

Communications Name

Session Type   **TCP**

Parallel Sessions Parameter: PARSESS=(max, default)

Operating Environment Parameter: ENVIRONMENT   **UNIX** / **NW** / **NT** (circle one)

Adjacent Node Name

Communications Name

Session Type   **TCP**

Parallel Sessions Parameter: PARSESS=(max, default)

Operating Environment Parameter: ENVIRONMENT   **UNIX** / **NW** / **NT** (circle one)

Adjacent Node Name

Communications Name

Session Type   **TCP**

Parallel Sessions Parameter: PARSESS=(max, default)

Operating Environment Parameter: ENVIRONMENT   **UNIX** / **NW** / **NT** (circle one)

CONNECT:Direct for VM/ESA Tape Exit

This appendix describes the steps necessary for using VMTAPE. CONNECT:Direct for VM/ESA provides an exit that supports processing by the Sterling Software VM tape management software VMTAPE.

Step 1—Specify CONNECT:Direct Initialization Parameters

The VMTAPE exit with CONNECT:Direct for VM/ESA uses the following initialization parameters to support processing.

Required Parameter

The following CONNECT:Direct initialization parameter must be specified to use VMTAPE with CONNECT:Direct for VM/ESA.

TAPEMOUNT.EXIT=DMGVMTMX
specifies the name of the tape mount exit. For VMTAPE, this must be DMGVMTMX. DMGVMTMX is the name of the tape mount exit in the distribution load library that supports VMTAPE. DMGVMTMX is not modifiable as it is not distributed in source form.

Default: none

The following optional CONNECT:Direct initialization parameters can be used to support VMTAPE processing.

TAPE.RETPD=dddd

specifies the number of days the tape will be retained. *dddd* is the number of days. Values range from 0–9999.

Default: 0

VMTAPE.MACHINE.ID

identifies the VMTAPE service machine id that will receive tape services requests sent by VMTAPE exit (DMGVMTMX). This parameter is required if the VMTAPE service machine id is something other than VMTAPE.

Default: None

VMTAPE.TIMEOUT

is a timeout value used by VMTAPE to cancel the outstanding tape mount request if the tape is not mounted within this time frame. The parameter is specified in minutes. The valid range of 1–99.

Default: 00:30:00

Step 2 — Update VMTAPE CONFIG File

The VMTAPE CONFIG file contains the configuration specifications of VMTAPE. To use VMTAPE with CONNECT:Direct for VM/ESA, the following must be included in VMTAPE CONFIG.

▶ CONNECT:Direct for VM/ESA needs to be authorized for ANYTAPE, OWNEDBY, and SAVEFOR.

▶ CONNECT:Direct for VM/ESA needs to be in the userid list in the TAPEOPER record in order to receive tape operator messages.

CONNECT:Direct for VM/ESA Tape Mount/Demount Exit

This exit point has been added to provide more flexibility in the handling of tapes by CONNECT:Direct for VM/ESA and to allow a user or a CONNECT:Direct for VM/ESA module, to be called whenever a

tape is to be mounted or demounted. It is the responsibility of the tape mount exit, if one has been provided, to perform the following mount tasks:

▶ Acquire and attach a tape drive
▶ Ensure the proper tape has been mounted
▶ Perform any necessary tape label manipulations
▶ Position the tape for I/O

The responsibility of the tape mount exit for demounts includes the following:

▶ Perform any necessary tape label manipulations
▶ Perform tape cataloging
▶ Rewind and unload the specified tape

Note: It is not necessary for the tape mount exit to detach the specified tape drive when called for a demount.

The responsibility of the tape mount exit for queries includes the following:

▶ Set the volume serial (volser) count
▶ Set the VOLSER list

The presence of a tape mount exit is determined by the new initialization parameter TAPEMOUNT.EXIT. When a tape mount exit is to be used to handle tape mounts or demounts, this initialization parameter must be specified. The syntax for the initialization parameter follows:

```
TAPEMOUNT.EXIT=module name
```

The tape mount exit is called at four different points in:

▶ During DTF initialization to allow the tape mount exit to perform any of its own initialization

▶ Before a tape mount to obtain a list of volsers to use

▶ When a tape must be mounted

▶ When a tape must be rewound and unloaded

When the tape mount exit is called, it is passed the address of the mount exit control block MEXCB in Register 1. The macro defining MEXCB is

named DMFMEXCB and can be found in the CONNECT:Direct distribution macro library. The following figure shows the MEXCB control block. The fields that can be examined or modified by the tape mount exit are described after the figure.

```
MEXCB           DS 0D                TAPEMOUNT EXIT CONTROL BLOCK
*
*      CONTROL BLOCK INPUT/OUTPUT FIELDS
*
MEBLKSIZ        DS CL5               TAPE BLOCK SIZE
MELRECL         DS CL5               TAPE LOGICAL RECORD LENGTH
MERECFM         DS CL1               TAPE RECORD FORMAT - F/V/U
MECNTLCH        DS CL1               CONTROL CHARS - BLANK/A/M
MEBLKATR        DS CL1               BLOCK ATTRIBUTE - BLANK/B/S/R
MERTNCD         DS F                 MOUNT EXIT RETURN CODE
MEMSGID         DS CL8               MOUNT EXIT MESSAGE ID
MEVOLSER        DS CL6               TAPE VOLSER
MEVOLCNT        DS H                 VOLSER COUNT
MEVOLLST        DS 255CL6            VOLSER LIST
*
*      CONTROL BLOCK INPUT ONLY FIELDS
*
MEFUNC          DS CL8               EXIT FUNCTION
*                                       INIT     => INITIALIZATION
*                                       MOUNT    => MOUNT TAPE
*                                       DEMOUNT  => DEMOUNT TAPE
*                                       QUERY    => RETURN VOLSER LIST
MEVOLSEQ        DS CL4               TAPE VOLUME SEQUENCE NUMBER
MEFILSEQ        DS CL4               TAPE FILE SEQUENCE NUMBER
MECUU           DS CL4               VIRTUAL ADDR OF TAPE DRIVE
MEDSN#          DS H                 LENGTH OF DATASET NAME
MEDSN           DS CL44              TAPE DATASET NAME
MEACCT#         DS H                 LENGTH OF ACCOUNTING INFO
MEACCT          DS CL256             ACCOUNTING INFORMATION
MEDEN           DS CL1               TAPE DENSITY
*                                       0 => 200   BPI
*                                       1 => 556   BPI
*                                       2 => 800   BPI
*                                       3 => 1600  BPI
*                                       4 => 6250  BPI
*                                       5 => 38K   BPI
METRTCH         DS CL2               7-TRACK TAPE RECORDING MODE
*                                    XF=COMPRESS        \
*                                                         IDRC
*                                    NF=NOCOMPRESS     /
MEEXPDT         DS CL5               TAPE EXPIRATION DATE - YYDDD
MERETPD         DS CL4               TAPE RETENTION PERIOD IN DAYS
MELABEL         DS CL3               TAPE LABEL TYPE
*                                       SL => STANDARD-LABELED
*                                       NL => NON-LABELED
MEGDG           DS CL8               GDG NUMBER - BLANK/0/+NN/-NN
MEPASSWD        DS CL8               PASSWORD
MEPOOL          DS CL16              TAPE POOL NAME
MEMODE          DS CL3               TAPE ACCESS MODE
*                                       IN  => INPUT ONLY
*                                       OUT => OUTPUT AND INPUT
MEUSERID        DS CL8               TAPE OWNERS USERID
MEDENXF         DS CL1               Y=SYSTEM DEFAULT COMPRESS
*                                    N=SYSTEM DEFAULT NOCOMPRESS
                DS CL3               FILLER
MERSTRT         DS CL1               NON-BLANK IF IN RESTART
MESCRTCH        DS CL1               NON-BLANK MEANS DON'T CATALOG
MEBLKCNT        DS CL6               BLOCK COUNT FOR TRAILER LABEL
MEEOF           DS CL1               NON-BLANK IF DEMOUNT AND AT EOF
MEUNIT          DS CL8               3480, 3480X, TAPE UNIT TYPE
                DS 3A                RESERVED FOR FUTURE USE
                DS 0D
MEXCB#          EQU *-MEXCB          SIZE OF MEXCB
```

MEBLKSIZ

is a five-character representation of a numeric field, ranging from 00000–32760, indicating the block length in bytes. If the record format of the file is fixed, then the block length must be a multiple of the logical record length for the file. If the record format of the file is variable, then the number represents the maximum block length (including a four-byte length indicator in the blocks). If the record format of the file is undefined, then the number represents the maximum block length.

If this is an input file, MEMODE=IN, and this field is blank, then the exit must set this field before returning to its caller.

MELRECL

is a five-character representation of a numeric field, ranging from 00000–32760, indicating the record length in bytes. If the record format of the file is fixed, then the number represents the logical record length. If the record format of the file is variable, then the number represents the maximum logical record length (including a four-byte length indicator in the records). If the record format of the file is undefined, then the number will be zero (00000).

If this is an input file, MEMODE=IN, and this field is blank, then the exit is expected to set this field before returning to its caller.

MERECFM

is a one-character field indicating the format of the records in the file. **F** is for records of fixed length, **V** is for variable length records, and **U** is for records with undefined lengths.

If this is an input file, MEMODE=IN, and this field is blank, then the exit must set this field, and the fields MECNTLCH and MEBLKATR, before returning to its caller.

MECNTLCH

is a one-character indicating whether a control character set was used to create the file and the type of control characters used. If this field is an **A**, then the file contains ASCII control characters. If this field is an **M**, then the file contains machine control characters. If this field is blank, then the file contains no control characters, unless this is an input file, in which case the exit is expected to set this field before returning to its caller.

MEBLKATR

is a one-character indicating the block attribute used to create the file. If this field is a **B**, then the file has blocked records. If this field is an **S**, then the file has spanned records. If this field is an **R**, then the file has both blocked and spanned records. If this field is blank, then the file has no blocked or spanned records, unless this is an input file, in which case the exit is expected to set this field before returning to its caller.

MERTNCD

is a fullword binary field indicating the success or failure of the tape mount exit. A nonzero value returned by the exit indicates some kind of failure occurred during its processing.

MEMSGID

is an eight-byte alphanumeric field indicating the success or failure of the tape mount exit. This field must be set when the return code set by the tape mount exit, MERTNCD, is nonzero before returning to its caller.

Note: All possible message IDs returned should be inserted into the CONNECT:Direct message file with the CONNECT:Direct message load program.

MEVOLSER

is a six-byte alphanumeric field identifying the tape volume with the file name specified. It may be from 1–6 characters, but if it is fewer than six characters, then it must be left-justified in the field. If the value of this field is SCRTCH upon entry to the tape mount exit, then the exit must set this field to the volume serial number of the tape mounted for the file requested before returning to its caller.

MEVOLCNT

is a halfword binary field containing the number of volsers in the field MEVOLLST. If MEFUNC=QUERY, then this field must be set by the exit before returning to its caller.

MEVOLLST

is a character field with room for up to 255 volsers containing a left-justified list of volsers with no separators between them. If MEFUNC=QUERY, then this field must be set by the exit before returning to its caller. MEVOLCNT volsers must be put in this field.

MEFUNC

is an eight-byte alphabetic, input-only field used to control the function of the tape mount exit. If this field is INIT, then the exit is expected to perform its initialization logic. If this field is MOUNT, then the exit is expected to perform its mount logic. If this field is DEMOUNT, then the exit is expected to perform its demount logic. If this field is QUERY, then the exit is expected to return a volser list in MEVOLLST and the number of volsers in the list in MEVOLCNT. The only fields set in the MEXCB that are passed to the exit for QUERYs are MEFUNC, MEDSN#, MEDSN, MEGDG, and MEUSERID.

MEVOLSEQ

is a four-character representation of a numeric field indicating the number of the tape volume within the multivolume group containing the specified file. Values range from 0001–9999.

MEFILSEQ

is a four-character representation of a numeric field indicating the relative position of the file specified within a multifile tape. Values range from 0001–9999, where 0001 is the first file on the tape.

MECUU

is a four-character representation of a numeric field indicating the virtual address of the tape drive to be used for the function requested by the tape mount exit caller. If MEFUNC=MOUNT, then a tape drive must be attached at this address. If MEFUNC=DEMOUNT, then the tape on the tape drive attached at this address must be rewound and unloaded. Values range 0F01–0FF1.

MEDSN#

is a halfword binary field containing the length of the file name in the MEDSN field.

MEDSN

is a four-byte alphanumeric field containing the name of the file on a tape to be mounted or demounted. The generation data group value, if any, is not part of this field. Only the rightmost 17 bytes of the file name in this field will be contained in any labels on the tape.

MEACCT#

is a halfword binary field containing the length of the accounting information in the MEACCT field.

MEACCT

is a 256-byte alphanumeric field containing process accounting information.

MEDEN

is a one-character representation of a numeric field indicating the recording density of the tape, according to the following table.

Value	Recording Density of Tape
0	200 bpi
1	556 bpi
2	800 bpi
3	1600 bpi
4	6250 bpi
5	38K bpi (3480 tape drives)

METRTCH

is a two-byte field indicating the tape recording technique used to create the file on the tape. This field is set only for 7-track tapes and for IDRC cartridges.

METRTCH Code	Tape Recording Technique
T	Odd parity with translation
C	Odd parity with conversion
E	Indicates even parity with no translation
ET	Indicates even parity with translation
blank	Odd parity with no translation or conversion (a 9-track tape)
XF	Indicates compression for IDRC cartridges
NF	Indicates no compression for IDRC cartridges

MEEXPDT

is a five-character representation of a numeric field indicating the year and date that the file may be scratched or overwritten. The format of this field is YYDDD, where YY is the year (00–99) and DDD is the day (001–366). This field will be blanks if no expiration date has been specified.

MERETPD

is a four-character representation of a numeric field indicating the number of days from the create date to pass before the file may be scratched or overwritten. The range is 0000–9999. This field will be blanks if an expiration date has been specified.

MELABEL

is a three-character field indicating whether the tape is standard-labeled (SL) or non-labeled (NL).

MEGDG

is an eight-character representation of a numeric field indicating whether the file is part of a generation data group (GDG). If this field is blank, the file is not part of a GDG. If this field is not blank, then it will either be 0 (zero) or will begin with a + or – followed by a relative GDG number.

MEPASSWD

is an eight-byte alphanumeric field indicating the password to use when accessing the file for input and the password to use for write-protection when accessing the file for output. This field will be blank if no password has been specified.

MEPOOL

is a 16-byte alphanumeric field indicating the name of the tape pool for scratch tape mounts. This field will be blank except when a scratch tape is requested (MEVOLSER=SCRTCH) for output (MEMODE=OUT).

MEMODE

is a three-character field indicating the file access mode. If this field is IN, then the file is to be opened read-only. If this field is OUT, the file is to be opened for update.

MEUSERID

is an eight-byte alphanumeric field containing the userid of the user making the request.

MEDENXF

is a one-character field indicating the system default for IDRC compression. Valid values are Y or N.

MERSTRT

is a one-character field and is non-blank if in restart mode. If in restart mode, then the mount exit must be sure not to rewrite or modify the volume and header tape labels even if the tape is to be mounted for output.

MESCRTCH

is a one-character field and is set only when MEFUNC is DEMOUNT. If this field is non-blank, then the tape to be demounted should be scratched instead of being cataloged.

MEBLKCNT

is a six-character representation of a numeric field and is set only when MEFUNC is DEMOUNT. This value can be used for the tape trailer label. It is the number of blocks on the tape.

MEEOF

is a one-character field indicating whether EOV or EOF tape trailer labels are written on the tape. This field is non-blank only when MEFUNC is DEMOUNT and the end of the file has been reached.

MEUNIT

is an eight-character field indicating the type of tape drive needed. If this field is 3480, then a model 3480 cartridge drive is required. If this field is 3480X, then a model 3480 cartridge drive with IDRC capability is required. Otherwise, this field will be TAPE, and any tape drive that can handle the recording density and tape recording technique specified will satisfy the request.

VTAM Definitions

This appendix includes the following sample VTAM definitions to assist
you in planning for the installation:

▶   VTAM definitions for a full networking CONNECT:Direct
▶   VTAM definitions for cross-domain network
▶   VTAM definitions for a mainframe-to-microcomputer connection

VTAM Definitions for Full Networking CONNECT:Direct

This section shows a sample CONNECT:Direct network with an SNA
connection.

```
          CD.NODE.A                                    CD.NODE.B

 ┌──────────────────────────────┐           ┌──────────────────────────────┐
 │ LOCAL.NODE=CD.NODE.A         │           │ LOCAL.NODE=CD.NODE.B         │
 │  DTF APPLID=CDAPPL1          │           │  DTF APPLID=CDAPPL2          │
 │ SUBAREA 20                   │           │ SUBAREA 30                   │
 │                              │           │                              │
 │                              │           │                              │
 │                              │           │                              │
 │ ADJACENT.NODES      APPLIDS  │           │ ADJACENT.NODES      APPLIDS  │
 │ CD.NODE.B           CDAPPL2  │           │ CD.NODE.A           CDAPPL1  │
 └──────────────────────────────┘           └──────────────────────────────┘
```
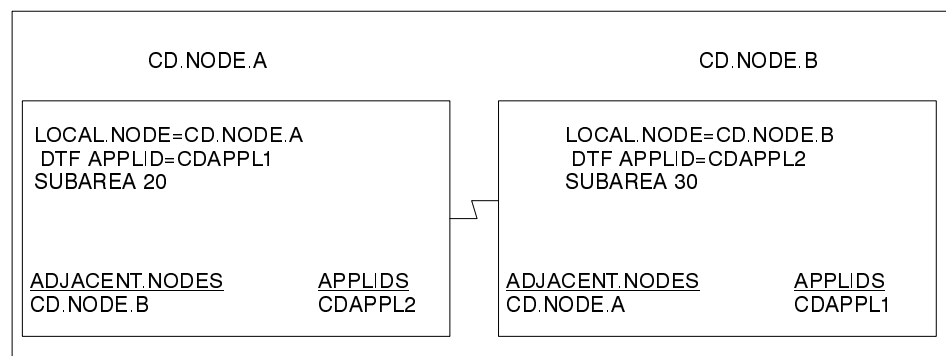
The previous example corresponds to the VTAM definitions presented
in the following sections.

Node A Definitions

The following is the DTF APPL definition for CD.NODE.A.

```
CDAPPL1 APPL   ACBNAME=CDAPPL1                                     X
               AUTH=(ACQ,NOCNM,NOPASS,NOPO,VPACE),                 X
               EAS=N,             APPROXIMATE # CONCURRENT SESS     X
               MODETAB=NDMTAB,    MODE TABLE NAME                  X
               SONSCIP=NO,        NO UNBIND IN SCIP EXIT           X
               SRBEXIT=NO,        NO SRB PROCESSING                X
               VPACING=7,         RECEIVE PACING OF 7              X
               DLOGMOD=NDMLOGM,   MODE TABLE ENTRY                 X
               PARSESS=YES,       PARALLEL SESSIONS CAN BE USED    X
               VTAMFRR=NO
```

The following are the APPL definitions for the API for CD.NODE.A.

```
NAI01    APPL  ACBNAME=NAI01,                                      x
               DLOGMOD=NDMLOGM,                                    X
               MODETAB=NDMTAB

NAI02    APPL  ACBNAME=NAI02,                                      x
               DLOGMOD=NDMLOGM,                                    X
               MODETAB=NDMTAB
```

The following is the Cross-Domain Manager definition for CD.NODE.A.

```
CDRMA    VBUILD TYPE=CDRM
SA30     CDRM   SUBAREA=30,ELEMENT=1,ISTATUS=ACTIVE
SA20     CDRM   SUBAREA=20,ELEMENT=1,ISTATUS=ACTIVE
```

The following is the Cross-Domain definition for CD.NODE.A.

```
CDRSCA   VBUILD TYPE=CDRSC
CDAPPL2 CDRSC  CDRM=SA30
```

The following is the Network Map definition for CD.NODE.A.

```
LOCAL.NODE=((CD.NODE.A   CDAPPL1 ,, $PW)   —
       TCQ=(TCX.FILE   TCQ.FILE))
* PNODE=SNODE DEFINITION                    *
ADJACENT.NODE=((CD.NODE.A   CDAPPL1A)   —
       PARSESS=(12 2) —
       APPLIDS=(NAI01,NAI02))
* SNA CONNECTIONS                           *
ADJACENT.NODE=((CD.NODE.B   CDAPPL2)   —
       PARSESS=(6 3) —
       APPLIDS=(NAI01,NAI02))
```

Node B Definitions

The following is the DTF APPL definition for CD.NODE.B.

```
CDAPPL2 APPL   ACBNAME=CDAPPL2                                     X
               AUTH=(ACQ,NOCNM,NOPASS,NOPO,VPACE),                 X
               EAS=N,              APPROXIMATE # CONCURRENT SESS    X
               MODETAB=NDMTAB,     MODE TABLE NAME                 X
               SONSCIP=NO,         NO UNBIND IN SCIP EXIT          X
               SRBEXIT=NO,         NO SRB PROCESSING               X
               VPACING=7,          RECEIVE PACING OF 7             X
               DLOGMOD=NDMLOGM,    MODE TABLE ENTRY                X
               PARSESS=YES,        PARALLEL SESSIONS CAN BE USED   X
               VTAMFRR=NO
```

The following are the APPL definitions for the API for CD.NODE.B.

```
NBI01     APPL  ACBNAME=NBI01,                                    x
                DLOGMOD=NDMLOGM,                                  X
                MODETAB=NDMTAB

NBI02     APPL  ACBNAME=NBI02,                                    x
                DLOGMOD=NDMLOGM,                                  X
                MODETAB=NDMTAB
```

The following is the Cross-Domain Manager definition for CD.NODE.B.

```
CDRMB     VBUILD TYPE=CDRM
SA30      CDRM   SUBAREA=30,ELEMENT=1,ISTATUS=ACTIVE
SA20      CDRM   SUBAREA=20,ELEMENT=1,ISTATUS=ACTIVE
```

The following is the Cross-Domain definition for CD.NODE.B.

```
CDRSCA    VBUILD TYPE=CDRSC
CDAPPL1 CDRSC  CDRM=SA20
```

The following is the Network Map definition for CD.NODE.B.

```
LOCAL.NODE=((CD.NODE.B   CDAPPL2 ,, $PW)    —
       TCQ=(TCX.FILE    TCQ.FILE))
* PNODE=SNODE DEFINITION  *
ADJACENT.NODE=((CD.NODE.B   CDAPPL2B)   —
       APPLIDS=(NBI01,NBI02)
* SNA CONNECTIONS         *
ADJACENT.NODE=((CD.NODE.A   CDAPPL1)   —
       APPLIDS=(NBI01,NBI02)
```

Network or Domain Name in Cross Domain Network

The VTAM APPL definitions for CONNECT:Direct nodes are shown in the supplied examples (in $CD.CNTL) with the name field value (column 1) equal to the ACBNAME value. However, these names may be different. If they are different, the network name value contained in the name field must be unique within the network, and the value in the ACBNAME field must be unique within the domain.

If the names are different, follow these rules when defining the Network Map:

▶ The name specified for the local and adjacent nodes DTF APPLIDs must be the **network** name.

▶ The names specified in the APPLIDS keyword of the adjacent node must be the **domain** names.

The following sample illustrates definitions for two nodes with unique network and domain names.

```
         CD.NODE.A                              CD.NODE.B

   LOCAL.NODE=CD.NODE.A                  LOCAL.NODE=CD.NODE.B
    DTF  APPLID=CDDAPP1                   DTF APPLID=CDDAPP2
   SUBAREA 20                            SUBAREA 30
   API APPLIDS=(NAN01,NAN02)             API APPLIDS=(NBN01,NBN02)


   ADJACENT.NODES        APPLIDS        ADJACENT.NODES        APPLIDS
   CD.NODE.B             CDDAPP2        CD.NODE.A             CDDAPP1
```
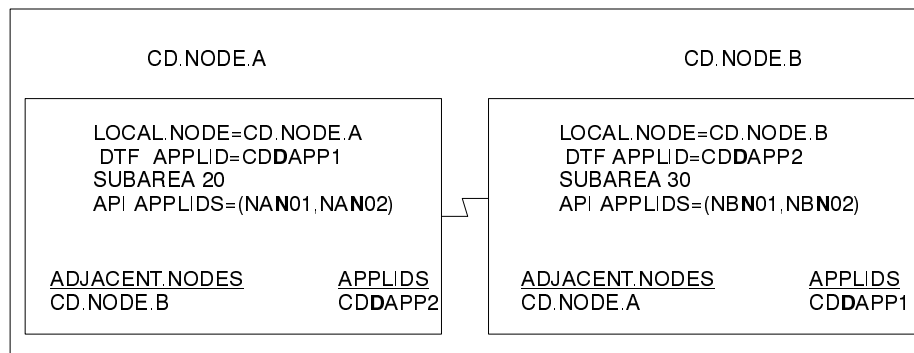
The above example corresponds to the VTAM definitions presented in the following figures.

In the examples which follow, domain names contain a **D** as part of their name and network names contain an **N**.

The following is the DTF APPL definition for CD.NODE.A.

```
CDNAPP1 APPL    ACBNAME=CDDAPP1
                AUTH=(ACQ,NOCNM,NOPASS,NOPO,VPACE),              X
                EAS=N,              APPROXIMATE # CONCURRENT SESS  X
                MODETAB=NDMTAB,     MODE TABLE NAME                X
                SONSCIP=NO,         NO UNBIND IN SCIP EXIT         X
                SRBEXIT=NO,         NO SRB PROCESSING              X
                VPACING=7,          RECEIVE PACING OF 7            X
                DLOGMOD=NDMLOGM,    MODE TABLE ENTRY               X
                PARSESS=YES,        PARALLEL SESSIONS CAN BE USED  X
                VTAMFRR=NO
```

The following are the APPL definitions for the API for CD.NODE.A.

```
NAN01    APPL  ACBNAME=NAID01,                                   X
               DLOGMOD=NDMLOGM,                                  X
               MODETAB=NDMTAB

NAN02    APPL  ACBNAME=NAID02,                                   X
               DLOGMOD=NDMLOGM,                                  X
               MODETAB=NDMTAB
```

The following is the Cross-Domain Manager definition for CD.NODE.A.

```
CDRMA     VBUILD TYPE=CDRM
SA30      CDRM   SUBAREA=30,ELEMENT=1,ISTATUS=ACTIVE
SA40      CDRM   SUBAREA=40,ELEMENT=1,ISTATUS=ACTIVE
```

The following is the Cross-Domain definition for CD.NODE.A.

```
CDRSCA    VBUILD TYPE=CDRSC
CDNAPP2   CDRSC  CDRM=SA30
```

The following is the Network Map definition for CD.NODE.A.

```
LOCAL.NODE=((CD.NODE.A   CDNAPP1 ,, $PW)   —
        TCQ=(CD.NODEA.TCX CD.NODEA.TCQ))
*                        *
ADJACENT.NODE=((CD.NODE.A   CDNAPP1)   —
        APPLIDS=(NAID01,NAID02)
*                        *
ADJACENT.NODE=((CD.NODE.B   CDNAPP2)   —
        APPLIDS=(NBID01,NBID02)
```

The following is the DTF APPL definition for CD.NODE.B.

```
CDNAPP2 APPL    ACBNAME=CDDAPP2
                AUTH=(ACQ,NOCNM,NOPASS,NOPO,VPACE),              X
                EAS=N,              APPROXIMATE # CONCURRENT SESS  X
                MODETAB=NDMTAB,     MODE TABLE NAME               X
                SONSCIP=NO,         NO UNBIND IN SCIP EXIT        X
                SRBEXIT=NO,         NO SRB PROCESSING             X
                VPACING=7,          RECEIVE PACING OF 7           X
                DLOGMOD=NDMLOGM,    MODE TABLE ENTRY              X
                PARSESS=YES,        PARALLEL SESSIONS CAN BE USED X
                VTAMFRR=NO
```

The following are the APPL definitions for the API for CD.NODE.B.

```
NBN01    APPL  ACBNAME=NBID01,                                   X
               DLOGMOD=NDMLOGM,                                  X
               MODETAB=NDMTAB

NBN02    APPL  ACBNAME=NBID02,                                   X
               DLOGMOD=NDMLOGM,                                  X
               MODETAB=NDMTAB
```

The following is the Cross-Domain Manager definition for CD.NODE.B.

```
CDRMB    VBUILD TYPE=CDRM
SA30     CDRM   SUBAREA=30,ELEMENT=1,ISTATUS=ACTIVE
SA20     CDRM   SUBAREA=20,ELEMENT=1,ISTATUS=ACTIVE
```

The following is the Cross-Domain definition for CD.NODE.B.

```
CDRSCA   VBUILD TYPE=CDRSC
CDNAPP1  CDRSC  CDRM=SA20
```

The following is the Network Map definition for CD.NODE.B.

```
LOCAL.NODE=((CD.NODE.B   CDNAPP2 ,, $PW)    —
       TCQ=(CD.NODEB.TCX CD.NODEB.TCQ))
*                       *
ADJACENT.NODE=((CD.NODE.B   CDNAPP2)    —
       APPLIDS=(NBID01,NBID02)
*                       *
ADJACENT.NODE=((CD.NODE.A   CDNAPP1)    —
       APPLIDS=(NAID01,NAID02)
```

VTAM Definitions for Mainframe-to-Microcomputer Connection

If you use CONNECT:Direct exclusively for mainframe-to-microcomputer communication, please note that instructions in this manual generally apply to the communication between two or more mainframe CONNECT:Direct nodes, not mainframe-to-PC connections.

Users setting up VTAM definitions for mainframe-to-microcomputer only connections should disregard information about the following:

▶ Multiple MVS sites within the CONNECT:Direct network
▶ VTAM cross-domain definitions, except as applicable to CONNECT:Direct for MS-DOS, OpenVMS, TANDEM, and OS/400.

Examples on the following pages pertain to the use of CONNECT:Direct for VM/ESA and CONNECT:Direct for MS-DOS in a mainframe-to-microcomputer network only.

The following figure illustrates a sample CONNECT:Direct network. This sample network corresponds to the VTAM definitions presented in the sections that follow. Note that the APPLIDS defined in the host Network Map must be the PC's VTAM LU.


```
                    CONNECT:Direct for MVS

              LOCAL.NODE=CD.NODE.A
               DTF APPLID=CDDAPP1
              SUBAREA 20

              ADJACENT.NODES        APPLIDS
              CD.PC.A               L2A1LU02
              CD.PC.B               T3A1LU01

    Switched                                    Leased
    Connection                                  Connection

    LOCAL.NODE=CD.PC.A              LOCAL.NODE=CD.PC.B
    LU=L2A1LU02                     LU=T3A1LU02

    CONNECT:Direct for MS-DOS      CONNECT:Direct for MS-DOS
```
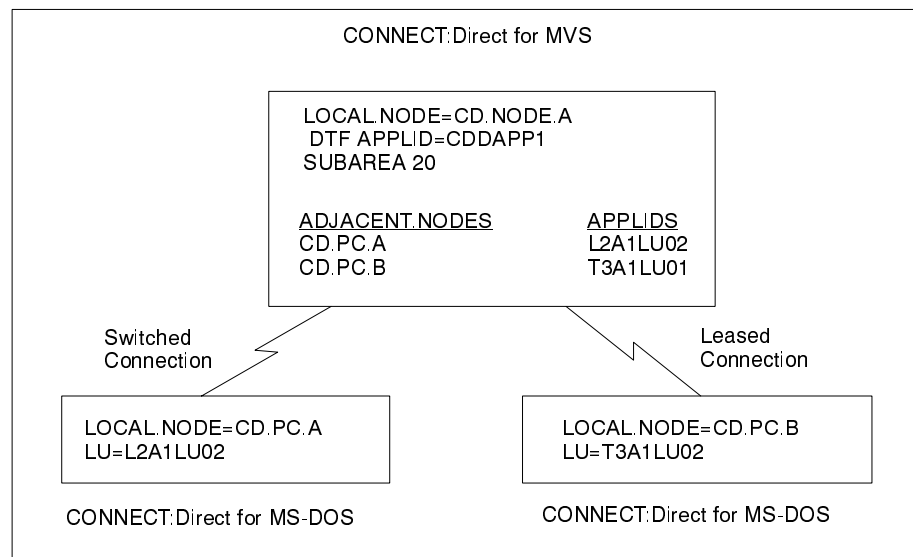
For each CONNECT:Direct for MS-DOS node that communicates with CONNECT:Direct for VM/ESA, a Network Map must exist on the CONNECT:Direct for MS-DOS node. Entries in the CONNECT:Direct for MS-DOS Network Map correspond to entries in the Network Map.

Example Definitions for CD.NODE.A

The following shows VTAM APPL definition for the DTF.

```
CDAPPL  APPL   ACBNAME=CDAPPL,        VTAM APPLICATION ID           X
               AUTH=(ACQ,NOCNM,NOPASS,NOPO,VPACE),                  X
               DLOGMOD=NDMLOGM,       LOGON MODE ENTRY FOR LU0      X
               EAS=20,                APPROX.  # OF CONCURRENT SESS. X
               MODETAB=NDMTAB,        MODE TABLE NAME               X
               SONSCIP=NO,            NO UNBIND IN SCIP EXIT        X
               SRBEXIT=NO,            NO SRB PROCESSING             X
               VPACING=6,             PACE COUNT                    X
               VTAMFRR=NO             NO FUNCTION RECOVERY ROUTINE
```

The following shows VTAM APPL definitions for the API and the DTF.

```
 NAI01    APPL  ACBNAME=NAI01,         VTAM APPLICATION ID           X
               MODETAB=NDMTAB,        LOGON MODE TABLE              X
               DLOGMODE=NDMLOGM,      LOGON MODE ENTRY FOR LU0      X
               VPACING=6              PACE COUNT
```

The following shows a mode table entry used with 3270-type connections, non-structured fields.

```
D4C32782 MODEENT LOGMODE=D4C32782,    ENTRY NAME                    X
               FMPROF=X'03',          FUNCTION MGMT PROF.  3        X
               TSPROF=X'03'           TRANSMISSION SERVICES PROTOCOL X
               PRIPROT=X'B1',         PRIMARY PROTOCOL              X
               SECPROT=X'90',         SECONDARY PROTOCOL            X
               COMPROT=X'3080',       COMMON PROTOCOL               X
               RUSIZES=X'87F8',       1K MAX.  SEC 3.8K MAX.  PRI   X
               PSERVIC=X'02000000000185000007E00'
                         0 1 2 3 4 5 6 7 8 9 1011
         MODEEND
         END
```

The following shows a leased network connection for a 3270-type connection.

```
 L3APU1   PU     ADDR=05,               (3274-51C) SNA POLLING ADDRESS  X
               PUTYPE=2,              PHYSICAL UNIT TYPE 2          X
               ISTATUS=ACTIVE,                                     X
               PASSLIM=7,             CONTIG SDLC FRAMES           X
               DLOGMOD=D4C32782,      DEFAULT BIND IMAGE           X
               MODETAB=NDMTAB,        LOG-ON MODE TABLE            X
               SSCPFM=USSSCS,         SUPPORT LOGON AND INITSELF   X
               USSTAB=USSTABS,        UNFORMATTED SYS SERV TAB     X
               PACING=1,              PACE COUNT                   X
               VPACING=1,             PACE COUNT                   X
               MAXDATA=265,           MAX SDLC FRAME SIZE          X
               MAXOUT=7               FRAMES BEFORE LINK RESP
 ****************************************************************
 T3A1LU02 LU     LOCADDR=2,ISTATUS=ACTIVE
```

The following shows a sample NCP definition for an autodial line.

```
L2AGRP1   GROUP LNCTL=SDLC,            SDLC LINE                    X
                DIAL=YES,              AUTODIAL                     X
                RETRIES=(4,5,2),       RE-DIAL OPTIONS              X
                REPLYTO=3.0            TIMEOUT VALUE
*
L2ALNE1   LINE  ADDRESS=(071,HALF),   LINE 71                      X
                CALL=OUT,              OUT CALL ONLY                X
                AUTO=70               AUTODIAL UNIT
*
L2APU1    PU    MAXLU=8                MAXIMUM NUMBER OF LUs
```

The following shows a VTAM major node switched definition for CONNECT:Direct for MS-DOS with autodial.

```
SWNDMPC   VBUILD TYPE=SWNET,MAXGRP=1,MAXNO=3
*
L2APU1    PU    ADDR=C1,               SNA POLLING ADDRESS          X
                IDBLK=017,             XID BLOCK                    X
                IDNUM=000E7,           XID NUMBER                   X
                MAXDATA=265,           MAX SDLC FRAME SIZE          X
                MAXOUT=7,              FRAMES BEFORE LINK RESP      X
                MODETAB=NDMTAB,        MODE TABLE NAME              X
                DLOGMODE=D4C32782,     MODE TABLE ENTRY             X
                PACING=6,              PACE COUNT                   X
                PASSLIM=7,             CONTIG SDLC FRAMES           X
                USSTAB=USSTABS,        USS TABLE NAME               X
                SSCPFM=USSSCS,         SUPPORT LOGON AND INITSELF   X
                VPACING=6,             PACE COUNT                   X
                MAXPATH=1,             MAXIMUM PATHS                X
                DISCNT=(YES,F)         DISCONNECT OPTION
*
L2APTH1   PATH  DIALNO=2920104,        TELEPHONE NUMBER             X
                GID=1,                 GROUP OF PATHS ACROSS ALL PUs X
                GRPNM=L2AGRP1,         AUTODIAL LINE GROUP NAME     X
                PID=1,                 IDENTIFIER FOR PATH DEFINED  X
                REDIAL=1               NUMBER OF DIALING RETRIES
*
L2A1LU02 LU     LOCADDR=2
```

The following shows the Network Map definition.

```
LOCAL.NODE = ((CD.NODE.A CDAPP1,, XYZZ1) -
        TCQ = (CD.NODEA.TCX CD.NODEA.TCQ))
ADJACENT.NODE = (CD.NODE.A CDAPP1) -
        APPLIDS = (NAI01 NAI02 NAI03)
ADJACENT.NODE = (CD.PC.A L2A1LU02)
ADJACENT.NODE = (CD.PC.B T3A1LU02)
```

Glossary

A

ADJACENT.NODE

Entry in the Network Map. Adjacent node entries define nodes in the network with which the local CONNECT:Direct may communicate. Each entry specifies a locally used CONNECT:Direct name, its associated network communications name, and session control parameters for these nodes.

API (Application Program Interface)

This CONNECT:Direct component accepts commands from the Interactive User Interface (IUI), Batch Interface, the Operator Interface, or user-written program and places them in a format so that the user's request can be executed by the DTF. If there are errors, the API returns a message to the user. If there are no errors, the API sends the command to the DTF using a VTAM session.

APPLID

The name specified in the ACB macro that identifies the application program to VTAM. For CONNECT:Direct, these applids correspond to a DTF node name or an API applid.

API POOL

Identifies the applids to be used for API communication with the DTF.

API POOL Segregation

To separate the pools of APPLIDs for use by the individual API types (BATCH, CICS, and TSO).

API System ID(s)

The System Identifier (SYSIDs or SMF IDs) of the CPUs (up to sixteen) that will be sharing a copy of the CONNECT:Direct DTF utilizing the SDF (Shared DASD Facility) option of CONNECT:Direct.

Authorization Facility

CONNECT:Direct facility that grants access to CONNECT:Direct commands.

Authorization File

CONNECT:Direct authorization file used to control access to CONNECT:Direct and identify commands that can be executed by userid. This file can also be used in conjunction with security exit interfaces with the secured point of entry feature.

B

Batch Interface

CONNECT:Direct interface that allows users to request CONNECT:Direct services from a batch job stream through control statements passed to a CONNECT:Direct-supplied program, DMBATCH.

C

Checkpoint/Restart

Eliminates the need to retransmit an entire file in the event of a transmission failure. A value on the COPY statement or in the initialization parameter, CKPT.MODE and CKPT, specifies the checkpoint interval. If a copy procedure is interrupted, CONNECT:Direct will restart that copy at the last checkpoint.

Command Line Interface

> CONNECT:Direct interface that allows users to submit
> CONNECT:Direct Processes and commands from their native command
> line environment.

Commands

> Are used to initiate and monitor activity within the CONNECT:Direct
> system and can be issued from the TSO/ISPF IUI, the operator console,
> a batch job, or a user application program.

D

DTF (Data Transmission Facility)

> The nucleus component of CONNECT:Direct. The DTF controls
> information distribution to other CONNECT:Direct nodes in the
> network. Start-up parameters that govern the overall activity of the
> DTF are defined within the initialization parameters.

E

ESF (Extended Submit Facility)

> An optional CONNECT:Direct for MVS feature. The ESF allows users to
> queue data transfer requests to a CONNECT:Direct node that is not
> active. This allows users to submit work to CONNECT:Direct, even if
> the CONNECT:Direct DTF is down.

I

InterCONNECT Option (ICO)

> The InterCONNECT Option is an optional feature of CONNECT:Direct
> and CONNECT:Mailbox which provides an automatic, secure way to
> route application-produced distribution files from a Direct supported
> node to a Mailbox node for distribution, automatically distribute
> Mailbox batches to a Direct node upon arrival, and provide complete
> notification of success or failure at each step of the process.

IUI (Interactive User Interface)

> The IUI interface is an ISPF screen and dialog component that allows
> users to define and submit CONNECT:Direct Processes as well as issue

CONNECT:Direct commands that monitor and control administrative and operations activity. An IUI is also available for a CICS environment with the optional product, CONNECT:Direct for CICS.

L

LOCAL.NODE

Entry in the Network Map. The local node entry defines the logical CONNECT:Direct name of the local CONNECT:Direct DTF and its associated communications name. The local node entry also contains the name of the transmission queue and the SUPERUSR ID password, if specified.

M

Modal Statements

CONNECT:Direct modal statements (IF THEN, EIF, ELSE, EXIT, and GOTO) allow you to alter the sequence of CONNECT:Direct Process execution based on completion of a previous Process step.

N

Network Map

VSAM file identifying all valid CONNECT:Direct nodes and applids in the network. There is one Network Map (Netmap) associated with each CONNECT:Direct node. There is one entry in that netmap for each of the other CONNECT:Direct nodes to which the local CONNECT:Direct node can initiate a session. The netmap entries also contain the rules or protocol to which the nodes will adhere when communicating.

Node

Any site in a network from which information distribution can be initiated.

O

Online Messages

Completion and error messages that are displayed online.

Operator Interface

Allows CONNECT:Direct commands to be issued from the MVS operator console. This interface also allows tailoring of CONNECT:Direct commands through a command list (CLIST) facility.

P

Parallel Sessions

Capability of having two or more concurrently active sessions between the same set of two LUs. With parallel session support, CONNECT:Direct allows multiple, concurrent file transfers between two CONNECT:Direct nodes.

PNODE (Primary Node)

CONNECT:Direct node on which the Process is being submitted. The primary node may also be referred to as the controlling or source node, but should not necessarily be interpreted as the sending node since PNODE can be the receiver. In every Process, there is one PNODE and one SNODE specified. The submitter of a Process is always the PNODE.

Process

A series of statements (which can be predefined and stored in a library) submitted through the API to initiate CONNECT:Direct activity, such as copying files, running jobs, and so on.

Process Statements

Are used to build a CONNECT:Direct Process. They contain instructions for transferring files, running operating system jobs, executing programs, or submitting other CONNECT:Direct Processes. Process statements include COPY, RUN JOB, RUN TASK, SUBMIT, SYMBOL, and Modals (conditional logic).

R

Retry Interval

Installation parameter that specifies the interval, in minutes, that the retries mentioned in the Max Retries parameter will be performed.

S

SDF (Shared DASD Facility)

An optional CONNECT:Direct for MVS product. Allows up to sixteen CPUs in a shared DASD complex to use a single copy of CONNECT:Direct. This allows applications and operators at each of the CPUs to initiate CONNECT:Direct activity. If the CPU containing the single copy of CONNECT:Direct also contains ESF, ESF is available to all other SDF nodes in the complex.

Segregation

To separate CONNECT:Direct APPLIDs by type. Also see API APPLID POOL Segregation.

Session Classes

Installation parameter that specifies the Process class groupings, priorities and number of Processes that can be concurrently executed on this CONNECT:Direct node.

SNA (Systems Network Architecture)

A network architecture designed to provide compatibility among a wide variety of hardware and software products so that they can be used to build complex networks. It defines protocols, standards, and message formats to which different hardware and software products must conform.

SNODE (Secondary Node)

The CONNECT:Direct node that interacts with the primary node (PNODE) during process execution. The secondary node (SNODE) may also be referred to as the participating, target, or destination node. In every Process, there is one PNODE and one SNODE.

SOLVE Option

The CONNECT:Direct for MVS SOLVE Option is an optional component of CONNECT:Direct. It is a callable application programming interface allowing SOLVE:Netmaster users to submit commands to and retrieve responses from CONNECT:Direct. The interface integrates automated data transfer with production applications on a variety of computer platforms running CONNECT:Direct.

SOLVE Option IUI

> The SOLVE Option IUI provides an interactive interface to CONNECT:Direct for SOLVE:Netmaster users.

Statistics Files

> A pair of VSAM data sets that hold CONNECT:Direct statistics records to document the history of a CONNECT:Direct Process.

Statistics Facility

> CONNECT:Direct facility that records all CONNECT:Direct activities.

SYMBOL Statement

> CONNECT:Direct Process statement that allows you to build symbolic substitution values.

Symbolics

> Symbolic parameters are supported within CONNECT:Direct Processes. This allows one predefined Process to be used for multiple applications. For example, the file names for a COPY operation could be passed to the Process by the user submitting the Process.

T

TCP/IP

> TCP/IP, Transmission Control Protocol/Internet Protocol, is a set of network standards that specify the details of how computers communicate, as well as a set of conventions for interconnecting networks and routing traffic.

TCQ (Transmission Control Queue)

> A VSAM relative record data set (RRDS) used to hold all Processes that have been submitted to CONNECT:Direct.

Index

T

Tandem, mode table example, 4-9

Tandem SNA nodes, worksheet, A-12

TCP/IP parameters, IBM implementation, 2-4

TCQ, general description, 1-5

TCQ file, space requirement, 3-3

TCX file, space requirement, 3-3

TRANSFR parameter, 5-5

TRANSFR, NCP parameter, 5-5

Transmission Control Index, VSAM definition, 3-5

transmission control index, space requirement, 3-3

Transmission Control Queue, VSAM definition, 3-5

transmission control queue, space requirement, 3-3

Transmission Control Queue (TCQ), general description, 1-5

TSO, IUI under, 1-3

tuning statistics file, 7-1

Type file
    space requirement, VSAM file, 3-3
    VSAM definition, 3-6

U

underlined letters, notational conventions, xiii

Unformatted Systems Services table, 4-14

UNIX LU6.2 nodes, worksheet, A-18

UNIX TCP/IP nodes, worksheet, A-20

uppercase and lowercase letters, notational conventions, xiii

uppercase letters, notational conventions, xii

USS table, 4-14

V

vertical bars, notational conventions, xiii

Virtual storage, size requirements, 2-2

VMTAPE exit, B-1
    MEXCB control block, B-3
    mount/demount, B-3

VSAM Definitions, Type file, 3-6

VSAM definitions
    authorization file, 3-3
    checkpoint file, 3-6
    LU1 script file, 3-7
    message file, 3-3
    Network Map, 3-6
    statistics files, 3-4
    Transmission Control Index, 3-5
    Transmission Control Queue, 3-5

VSAM files, 3-2
    organization and attributes, 3-2
    space requirement, 3-2

VTAM, network name vs domain name, C-4

VTAM and NCP critical parameters, 5-4

VTAM definitions
    batch interface, 4-1, 4-5
    cross domain resource definition (NDMCDRSC), 4-2
    cross domain resource manager node (NDMCDRM), 4-2
    DTF with LU0 (NDMAPPL), 4-2
    DTF with LU6.2 (NDMAPPL), 4-4
    examples, 4-1
    for SDF connection, 4-2
    IUI, 4-1, 4-5
    mode table, 4-2
    NDMAPPL, 4-1
    NDMCDRM, 4-2
    NDMCDRSC, 4-2
    NDMIAPPL, 4-1
    NDMMODET, 4-2
    preparation install step, 4-1
    Program Operator Application interface, 4-6

VTAM definitions, mainframe–to–microcomputer, C-7

VTAM definitions example, full networking SNA and NETEX, C-1

VTAM MODEENT macro, 5-1

VTAM parameters, RUSIZES, 5-2

W

Windows NT TCP/IP nodes, worksheet, A-20

Worksheets
    Installation Information, A-4
    Installation Steps, A-5
    Network Map definitions
        370 SNA LU0, A-9
        370 SNA LU6.2, A-15

X