

DECnet-Plus for OpenVMS

Introduction and User's Guide

Order Number: AA-Q191F-TE

November 1996

This manual introduces DECnet-Plus for OpenVMS. It provides an overview of the product's features and a conceptual overview of DECnet-Plus.

Revision/Update Information:	This manual supersedes the <i>DECnet/OSI for OpenVMS Introduction and User's Guide</i> .
Operating Systems:	OpenVMS VAX Version 7.1 OpenVMS Alpha Version 7.1
Software Versions:	DECnet-Plus for OpenVMS Version 7.1

Digital Equipment Corporation
Maynard, Massachusetts

November 1996

Digital Equipment Corporation makes no representations that the use of its products in the manner described in this publication will not infringe on existing or future patent rights, nor do the descriptions contained in this publication imply the granting of licenses to make, use, or sell equipment or software in accordance with the description.

Possession, use, or copying of the software described in this publication is authorized only pursuant to a valid written license from Digital or an authorized sublicensor.

Digital conducts its business in a manner that conserves the environment and protects the safety and health of its employees, customers, and the community.

© Digital Equipment Corporation 1996. All rights reserved.

The following are trademarks of Digital Equipment Corporation: Bookreader, DDCMP, DEC, DECdirect, DECnet, DECNIS, DECserver, DECsystem, DECwindows, Digital, DNA, InfoServer, OpenVMS, OpenVMS Cluster, PATHWORKS, ULTRIX, VAX, VAX DOCUMENT, VAXcluster, VAXstation, VMS, RSX, and the DIGITAL logo.

The following are third-party trademarks:

Macintosh is a registered trademark of Apple Computer, Inc.

MS-DOS is a registered trademark of Microsoft Corporation.

OS/2 is a registered trademark of International Business Machines Corporation.

OSF/1 is a registered trademark of Open Software Foundation, Inc.

OSI is a registered trademark of CA Management, Inc.

SCO is a trademark of Santa Cruz Operations, Inc.

UNIX is a registered trademark in the United States and other countries, licensed exclusively through X/Open Company Ltd.

All other trademarks and registered trademarks are the property of their respective holders.

Contents

Preface	ix
----------------------	----

Part I DECnet-Plus for OpenVMS Overview

1 Introducing DECnet-Plus for OpenVMS

1.1	Preparing for a Migration to DECnet-Plus	1-2
1.1.1	Differences Between DECnet Phase IV and DECnet-Plus Phase V ...	1-2
1.2	OSI and IETF Standards Supported	1-4
1.3	Dependencies and Licenses	1-6
1.3.1	DECnet Licenses	1-6
1.3.2	Name Services	1-7
1.3.2.1	Choosing the Local Namespace	1-7
1.3.2.2	Choosing DECdns	1-8
1.3.2.3	Choosing DNS/BIND	1-8
1.3.3	DECdts Time Server	1-8
1.3.4	Intermediate System	1-8
1.3.5	X.25	1-8
1.3.6	OSI Application Development	1-9
1.4	Distributed Networking	1-9
1.4.1	Distributed System Services	1-9
1.4.2	Distributed Network Applications	1-10
1.5	OpenVMS Cluster Systems	1-10
1.5.1	DECnet-Plus OpenVMS Cluster Alias	1-11
1.5.2	OpenVMS Cluster Configuration Procedure	1-11

2 DECnet-Plus for OpenVMS Components

2.1	Features of DECnet-Plus for OpenVMS	2-1
2.1.1	OpenVMS Software Components	2-2
2.2	DECnet-Plus for OpenVMS Base System	2-3
2.2.1	DECnet Over TCP/IP	2-5
2.2.2	Digital-Supplied Session Control Applications	2-5
2.2.3	Programming Interfaces	2-6
2.2.4	OSI Transport Service	2-7
2.2.5	NSP Transport Service	2-8
2.2.5.1	End-System-to-Intermediate-System (ES-IS) Routing	2-8
2.2.6	Network Management	2-8
2.2.7	Name Service	2-8
2.2.7.1	Local Namespace	2-9
2.2.8	Time Service	2-10
2.3	X.25	2-10
2.3.1	X.25 Native Software	2-11
2.3.2	X.25 Access Software	2-12

2.4	Wide Area Network Device Drivers	2-12
2.5	OSAK	2-12
2.5.1	OSAK API	2-13
2.5.2	OSAK Software and Management Facilities	2-14
2.5.2.1	Protocol Machine	2-14
2.5.2.2	Network Control and Management Facilities	2-14
2.5.2.3	OSAK Trace Utility	2-14
2.5.3	Active and Passive Addresses	2-15
2.6	FTAM	2-15
2.6.1	FTAM Gateway	2-16
2.7	Virtual Terminal (VT)	2-16
2.7.1	VT Gateways	2-16
2.7.1.1	LAT/VT and VT/LAT Gateways	2-17
2.7.1.2	VT/Telnet and Telnet/VT Gateways	2-17
2.8	Management Tools and Utilities	2-17
2.8.1	Network Control Language (NCL)	2-17
2.8.1.1	Network Management Graphical User Interface (NET\$MGMT)	2-18
2.8.2	Network Control Program (NCP)	2-18
2.8.3	Common Trace Facility (CTF)	2-18
2.8.4	CMISE API	2-19
2.8.5	Network Management Support Tools	2-19
2.8.6	DECnet-Plus Event Dispatcher (EVD)	2-19
2.8.7	CMIP Management Listener (CML)	2-20

Part II Conceptual Information

3 DECnet-Plus for OpenVMS Concepts

3.1	Communications Architecture: The OSI Reference Model	3-1
3.1.1	Comparison of DNA Phases	3-2
3.2	Data Flow	3-3
3.3	OSI Terminology	3-4
3.3.1	Protocols	3-4
3.3.2	Protocol Stacks (Towers)	3-5
3.3.3	Dialogues	3-5
3.3.4	Entities	3-6
3.3.5	Services	3-6
3.3.5.1	Service Primitives	3-6
3.3.5.2	Service Access Points (SAPs)	3-6
3.4	The Upper Three OSI Layers	3-8
3.4.1	Presentation and Session Entities	3-8
3.5	DNA Session Control Layer	3-9
3.5.1	Objects and Applications	3-9
3.5.2	Protocol Towers	3-9
3.5.3	Address Resolution	3-10
3.5.4	Address Selection	3-10
3.5.5	Connection Control	3-10
3.5.6	Digital-Supplied Session Control Layer Applications	3-10
3.6	Transport Layer	3-11
3.6.1	Transport Layer Ports and Session Layer Ports	3-11
3.6.2	Supported Transport Protocols	3-11
3.6.2.1	OSI Transport Protocol	3-12
3.6.2.2	Network Services Protocol (NSP)	3-13

3.6.3	OSI Transport Service	3-13
3.6.3.1	OSI Transport Service Functions	3-13
3.6.3.2	OSI Transport Templates	3-14
3.6.3.3	OSI Transport Connections	3-14
3.7	Network Layer	3-15
3.7.1	Network Services	3-15
3.7.1.1	Connectionless-Mode Network Service (CLNS)	3-16
3.7.1.2	Connection-Oriented Network Service (CONS)	3-17
3.7.2	Broadcast Network Connections: The Routing Process	3-18
3.7.2.1	Compliance with OSI Packet Formats and Addressing	3-19
3.8	Data Link Layer	3-20
3.8.1	Support for CSMA-CD Protocol on a LAN	3-21
3.8.1.1	Extended LANs	3-22
3.8.1.2	Intermediate System	3-23
3.8.1.3	Multicircuit End Systems	3-23
3.8.1.4	Areas and Multihomed Systems	3-23
3.8.2	Support for the HDLC Protocol	3-23
3.8.2.1	LAPB Support	3-24
3.8.3	Support for the DDCMP Protocol	3-24
3.8.3.1	Synchronous DDCMP	3-25
3.8.3.2	Asynchronous DDCMP	3-25
3.8.3.3	Converting DDCMP Links to HDLC Links	3-26
3.9	Physical Layer	3-26
3.9.1	CSMA-CD LAN Interface	3-26
3.9.2	Modem Connect Module	3-26
3.10	Network Management	3-27
3.10.1	Network Management Components	3-27
3.10.2	How the Director Works	3-28
3.10.2.1	Management Access Relationship	3-28
3.10.2.2	Management of Remote DECnet Phase IV Nodes	3-29
3.10.2.3	Support for Phase IV NCP and the NICE Protocol	3-29
3.10.2.4	Maintenance Operation Protocol (MOP) Support	3-29
3.10.3	Configuration Procedure for DECnet-Plus for OpenVMS	3-29

4 X.25 Networking Concepts

4.1	Introduction	4-1
4.1.1	Packet Switching Data Networks (PSDN)	4-1
4.1.2	PADs and Character-Mode Terminals	4-3
4.2	X.25 Protocols	4-5
4.2.1	Packet Level	4-6
4.2.2	Frame Level	4-6
4.2.3	Physical Level	4-7
4.3	Implemented Standards	4-7
4.3.1	Related CCITT Standards	4-7
4.3.2	Related OSI Standards	4-8
4.4	How Connections Are Made and Data Is Transferred	4-9
4.4.1	Physical Connections	4-9
4.4.2	Logical Channels	4-9
4.4.2.1	Virtual Circuits	4-10
4.4.2.2	Logical Channel Numbers	4-10
4.4.3	DTE Addresses	4-11
4.4.4	Controlling the Flow of Packets	4-11
4.5	Optional Facilities of Public and Private PSDNs	4-12

4.5.1	DTE Parameters	4-12
4.5.2	PAD Parameters	4-12
4.6	Network Profiles	4-13
4.7	PSDNs Supported by Digital's X.25 Products	4-13

Part III Using DECnet-Plus Utilities

5 DECnet Network Operations

5.1	How You Can Use DECnet	5-1
5.1.1	Performing Operations using DCL Commands	5-1
5.2	Specifying Node Names	5-2
5.2.1	Phase IV-Style Node Synonyms	5-2
5.2.2	Name Abbreviation Using Local Roots	5-2
5.2.3	Logging In to Other DECnet Nodes	5-2
5.3	Accessing Remote Files	5-3
5.3.1	Specifying Files	5-3
5.3.2	Specifying Files on a Non-OpenVMS System	5-3
5.3.3	Protecting Files	5-4
5.4	File Operations	5-4
5.4.1	Displaying Remote Directories and Files	5-5
5.4.2	Public Directories	5-5
5.4.3	Copying and Printing Remote Files	5-5
5.4.3.1	The COPY Command	5-5
5.4.3.2	The APPEND Command	5-6
5.4.3.3	The PRINT/REMOTE Command	5-6
5.4.4	Other Remote File Operations	5-6
5.4.4.1	Edit	5-7
5.4.4.2	Delete	5-7
5.4.4.3	Purge	5-7
5.4.4.4	Search	5-7
5.4.4.5	Compare	5-7
5.4.4.6	Sort	5-8
5.4.4.7	Merge	5-8
5.4.4.8	Examine	5-8
5.4.4.9	Back Up	5-8
5.5	Using the Mail and Phone Utilities	5-9
5.5.1	The MAIL Command	5-9
5.5.1.1	Sending Files and Long Messages	5-9
5.5.2	The Phone Utility	5-10

6 File Operations to and from Other DECnet and DECnet-Plus Nodes

6.1	General DECnet Restrictions	6-1
6.2	OpenVMS to Digital UNIX or ULTRIX Network Operations	6-2
6.2.1	File System Constraints	6-2
6.2.1.1	File Formats and Access Modes	6-3
6.2.1.2	OpenVMS RMS Interface	6-4
6.2.1.3	File Specifications	6-4
6.2.2	DCL Considerations	6-4
6.2.2.1	COPY	6-5
6.2.2.2	DIRECTORY	6-5
6.3	OpenVMS to MS-DOS Network Operations	6-5

6.3.1	File System Constraints	6-5
6.3.1.1	File Formats and Access Modes	6-5
6.3.1.2	OpenVMS RMS Interface	6-6
6.3.1.3	File Specifications	6-7
6.3.2	DCL Considerations	6-7
6.3.2.1	COPY	6-7
6.3.2.2	DIRECTORY	6-7
6.4	OpenVMS to RSX Network Operation Using RMS-based FAL	6-8
6.4.1	File Formats and Access Modes	6-8
6.4.2	OpenVMS RMS Interface	6-8
6.4.3	File Specifications	6-8
6.4.4	DCL Considerations	6-9
6.4.4.1	COPY	6-9

Part IV Glossary

DECnet-Plus Glossary

G.1	Definitions	Glossary-2
G.2	Acronyms	Glossary-91

Index

Figures

2-1	DNA Phase V DECnet-Plus for OpenVMS	2-1
2-2	Component Relationships	2-11
2-3	OSAK Software on DECnet-Plus for OpenVMS Systems	2-13
3-1	OSI Reference Model	3-1
3-2	OSI Reference Model: Data Flow	3-3
3-3	The Relationship of PDUs, User Data, and SDUs	3-5
3-4	Service User, Service Provider, and SAP Interaction	3-7
3-5	SAPs at Different OSI Layers	3-7
3-6	ES-IS and IS-IS Protocols	3-19
4-1	Packet Switching Equipment	4-2
4-2	Packets Traveling Over a PSDN	4-3
4-3	Connecting Character-Mode DTEs to a PSDN	4-4
4-4	X.25 Protocol Levels	4-5
4-5	Structure of a Packet	4-6
4-6	Structure of a Frame	4-7
4-7	CCITT Packet-Switching Recommendations	4-8
4-8	Virtual Circuits Versus Logical Channels	4-10
4-9	DTE Address Example	4-11

Tables

1	DECnet-Plus for OpenVMS Documentation	x
1-1	DNA Phases	1-1
1-2	OSI Standards Supported by DECnet-Plus for OpenVMS	1-5
1-3	DECnet Phase IV and DECnet-Plus Licensing	1-7
2-1	Functions and Associated Software	2-3
2-2	OSAK Software: Summary of ISO Standards Implemented	2-14
3-1	Layers of the OSI Reference Model and Their Functions	3-2
3-2	Layer Names: DNA Phase IV and DNA Phase V	3-2
3-3	Functions of the OSI Transport Protocol Classes	3-12
1	Acronyms	Glossary-91
2	ISO and IEC Acronyms in Transition	Glossary-97

Preface

This book introduces the DECnet-Plus for OpenVMS (formerly DECnet/OSI) product, providing an overview of the product's features and a conceptual overview of DECnet-Plus software. Also described are common end-user tasks such as how to use the remote file, remote login, and mail utilities. A complete glossary of DECnet-Plus terminology is also included.

See your Software Product Description (SPD) for detailed information about new features and product requirements.

Intended Audience

This book is written for:

- Network planners and managers, both DECnet for OpenVMS (Phase IV) users and new DECnet-Plus (Phase V) users
- OpenVMS system managers
- Installers of OpenVMS
- Namespace planners and managers
- DECdts planners and managers
- Managers of these Open Systems Interconnection (OSI) features:
 - OSI Transport connections
 - X.25 communications
 - Wide area network device drivers (WANDD)
 - Remote OSI file operations
 - DECnet-Plus virtual terminal
 - Writing and running additional OSI applications

Document Structure

This book has four parts:

- Part I provides an overview of:
 - Transitioning to DECnet-Plus
 - Features and components of the DECnet-Plus for OpenVMS product

- Part II provides conceptual information about:
 - DECnet-Plus for OpenVMS
 - X.25 networking
- Part III provides user information on:
 - Using remote files with DECnet-Plus
 - Using the DECnet-Plus login utility
 - Sending mail to DECnet-Plus nodes
- Part IV is a glossary of DECnet-Plus terminology.

Guide to Documentation

DECnet-Plus for OpenVMS documentation is available in three sets:

- Documentation set for OpenVMS Alpha systems
- Documentation set for OpenVMS VAX systems
- Supplemental X.25 documentation set for OpenVMS Alpha systems

Table 1 lists the documentation that supports this version of the DECnet-Plus for OpenVMS software.

Table 1 DECnet-Plus for OpenVMS Documentation

Document	Contents
Documentation Sets: OpenVMS Alpha and VAX Systems	
<i>DECnet-Plus for OpenVMS Introduction and User's Guide</i>	This manual. Describes the manuals in the documentation sets, outlines the DECnet-Plus for OpenVMS features and tools, explains how to use and manage an end system, and provides a comprehensive glossary of DECnet terminology.
<i>DECnet-Plus for OpenVMS Release Notes</i>	Print this text file at the beginning of the installation procedure and read it before you install DECnet-Plus for OpenVMS. Volume 1 of the DECnet-Plus for OpenVMS distribution kit contains the Release Notes. Describes changes to the software; installation, upgrade, and compatibility information; new and existing software problems and restrictions; and software and documentation corrections.
<i>DECnet-Plus for OpenVMS Installation and Basic Configuration</i>	Explains how to install and configure the DECnet-Plus for OpenVMS software and how to perform postinstallation tasks.
<i>DECnet-Plus for OpenVMS Installation Quick Reference Card</i>	Provides quick-reference information to help you install DECnet-Plus software on an OpenVMS node. Use this card with the <i>DECnet-Plus for OpenVMS Installation and Basic Configuration</i> manual.

(continued on next page)

Table 1 (Cont.) DECnet-Plus for OpenVMS Documentation

Document	Contents
Documentation Sets: OpenVMS Alpha and VAX Systems	
<i>DECnet-Plus for OpenVMS Applications Installation and Advanced Configuration</i>	Explains how to install X.25 for OpenVMS Alpha, X.25 Access and X.25 Native Mode for OpenVMS VAX (formerly VAX P.S.I. Access and VAX P.S.I.), FTAM, VT, and OSAK software. Includes information about how to configure DECnet-Plus for OpenVMS using the Advanced configuration option and how to modify an existing configuration.
<i>DECnet-Plus for OpenVMS Network Management</i>	Provides in-depth information about how to monitor and manage DECnet-Plus for OpenVMS systems using various tools and Network Control Language (NCL) commands. Explains how to set up and use event dispatching and how to perform all day-to-day management tasks for the local DECnet-Plus for OpenVMS node, including setting up OpenVMS clusters, managing security, downline loading, and monitoring the network.
<i>DECnet-Plus for OpenVMS Network Management Quick Reference Guide</i>	Provides quick-reference information about the tools that help you manage and monitor a DECnet-Plus network. Use this guide with the <i>DECnet-Plus for OpenVMS Network Management</i> manual.
<i>DECnet-Plus Network Control Language Reference</i>	Outlines command descriptions and examples for all Network Control Language (NCL) commands that you execute to manage, monitor, and troubleshoot the network. Begins with an orientation chapter that contains information about how to execute NCL commands, followed by a command chapter for each module in the DECnet Phase V layered model.
<i>DECnet-Plus Planning Guide</i>	Provides configuration and planning guidelines, including namespace planning information, to help you transition a network from the DECnet Phase IV to DECnet Phase V architecture.
<i>DECnet-Plus Problem Solving</i>	Explains how to isolate and solve DECnet problems in an OpenVMS environment that can occur while the network is in operation. Includes information about how to perform loopback tests and how to use the DTS/DTR utility to solve problems.
<i>DECnet-Plus DECdns Management</i>	Explains DECdns concepts and how to manage a DECdns distributed namespace. Use this manual with the <i>DECnet-Plus Planning Guide</i> .
<i>DECnet-Plus DECdts Management</i>	Introduces Digital Distributed Time Service (DECdts) concepts and describes how to manage the software and system clocks.

(continued on next page)

Table 1 (Cont.) DECnet-Plus for OpenVMS Documentation

Document	Contents
Documentation Sets: OpenVMS Alpha and VAX Systems	
<i>DECnet-Plus DECdts Programming</i>	Contains DECdts time routine reference information and describes the time-provider interface (TPI).
<i>DECnet-Plus OSAK Programming</i>	Explains how to use the OSAK (OSI Applications Kernel) interface to create OSI (Open Systems Interconnection) applications for any supported operating system.
<i>DECnet-Plus OSAK Programming Reference</i>	Provides reference information on using the OSAK interface to create OSI applications on any supported operating system.
<i>DECnet-Plus OSAK SPI Programming Reference</i>	Provides reference information about using the OSAK session programming interface (SPI) to create OSI applications on any supported operating system.
<i>DECnet-Plus FTAM and Virtual Terminal Use and Management</i>	Explains how to use and manage FTAM (File Transfer, Access, and Management) software for remote file transfer and management and VT (Virtual Terminal) for remote login to OSI-compliant systems.
<i>DECnet-Plus FTAM Programming</i>	Explains how to access the FTAM protocol through Digital's FTAM API (application programming interface).
<i>DECnet-Plus for OpenVMS Programming</i>	A three-part manual. Contains information about how to design and write an application that follows a client/server model and uses the OpenVMS Interprocess Communication (\$IPC) system service and the transparent and nontransparent communication with the queue Input/Output (\$QIO) system service. Explains how to write programs using the OpenVMS system services to communicate with OSI transport services. Provides information about the Common Management Information Service (CMISE) API.
<i>DECnet/OSI for VMS CTF Use</i>	Explains how use the Common Trace Facility (CTF) troubleshooting tool to collect and analyze protocol data from networking software.
<i>DECnet/OSI for VMS X.25 Management</i>	For OpenVMS VAX systems only. Provides X.25 and X.29 information for X.25 Access and X.25 Native Mode (formerly VAX P.S.I. Access and VAX P.S.I.). Provides information about wide area network device driver (WANDD) software.
<i>VAX X.25 Problem Solving</i>	
<i>VAX P.S.I. Programming</i>	
<i>VAX P.S.I. Programming Reference</i>	
<i>VAX WANDD Programming</i>	
<i>VAX P.S.I. Accounting</i>	
<i>VAX P.S.I. X.25 Use</i>	
<i>VAX P.S.I. X.29 Management</i>	
<i>VAX X.25 Security</i>	

(continued on next page)

Table 1 (Cont.) DECnet-Plus for OpenVMS Documentation

Document	Contents
Documentation Sets: OpenVMS Alpha and VAX Systems	
<i>X.25 for OpenVMS Management Guide</i>	For OpenVMS Alpha systems only. Explains how to manage and monitor an X.25 system using network tools.
<i>X.25 for OpenVMS Security Guide</i>	For OpenVMS Alpha systems only. Explains the X.25 security model and the tasks required to set up and manage X25 security.
<i>X.25 for OpenVMS Problem Solving</i>	For OpenVMS Alpha systems only. Provides guidance on how to solve problems that can occur while using an X.25 system.
Supplemental X.25 Documentation Set (OpenVMS Alpha Systems)	
<i>X.25 for OpenVMS Accounting</i>	Explains how to use X.25 accounting to obtain performance records and information about how X.25 is being used.
<i>X.25 for OpenVMS Configuration</i>	Discusses how to configure X.25 on an OpenVMS Alpha system.
<i>X.25 for OpenVMS Programming</i>	Explains how to write X.25 and X.29 programs to perform network operations.
<i>X.25 for OpenVMS Programming Reference</i>	Provides reference information for X.25 and X.29 programmers.
<i>X.25 for OpenVMS Utilities</i>	Explains how to use and manage X.25 Mail and X.29 communications.

For additional information on the DECnet-Plus products and services, access the Digital OpenVMS World Wide Web site. Use the following URL:

<http://www.openvms.digital.com>

Reader's Comments

Digital welcomes your comments on this manual or any of the DECnet-Plus documents. Send us your comments through any of the following channels:

Internet	openvmsdoc@zko.mts.dec.com
Fax	603 881-0120, Attention: OSSG Documentation, ZKO3-4/U08
Mail	OSSG Documentation Group, ZKO3-4/U08 110 Spit Brook Rd. Nashua, NH 03062-2698

How To Order Additional Documentation

Use the following table to order additional documentation or information. If you need help deciding which documentation best meets your needs, call 800-DIGITAL (800-344-4825).

Telephone and Direct Mail Orders

Location	Call	Fax	Write
U.S.A.	DECdirect 800-DIGITAL 800-344-4825	Fax: 800-234-2298	Digital Equipment Corporation P.O. Box CS2008 Nashua, NH 03061
Puerto Rico	809-781-0505	Fax: 809-749-8300	Digital Equipment Caribbean, Inc. 3 Digital Plaza, 1st Street, Suite 200 P.O. Box 11038 Metro Office Park San Juan, Puerto Rico 00910-2138
Canada	800-267-6215	Fax: 613-592-1946	Digital Equipment of Canada, Ltd. Box 13000 100 Herzberg Road Kanata, Ontario, Canada K2K 2A6 Attn: DECdirect Sales
International	—	—	Local Digital subsidiary or approved distributor
Internal Orders	DTN: 264-4446 603-884-4446	Fax: 603-884-3960	U.S. Software Supply Business Digital Equipment Corporation 8 Cotton Road Nashua, NH 03063-1260

ZK-7654A-GE

Terminology

The following terms are used interchangeably:

- Transition and migration
- Phase IV and DECnet Phase IV
- Phase V and DECnet Phase V
- System and node
- End system and end node
- Intermediate system and router
- Multivendor and non-Digital-specific
- Link state and:
 - Link state routing algorithm
 - Link state protocol
 - DECnet-Plus routing algorithm
 - DECnet-Plus routing
- Name service and directory service

Conventions

The following conventions apply to this book.

Convention	Meaning
special type	Indicates a literal example of system output or user input. In text, indicates command names, keywords, node names, file names, directories, utilities, and tools. On a DECnet-Plus for ULTRIX system, enter the word or phrase in the exact case shown. You can abbreviate command keywords to the smallest number of characters that OpenVMS, DEC OSF/1, NCL, DECdns, DECdts, and the other utilities accept, usually three characters.
<i>italic</i>	Indicates a variable.
text style	Indicates a new term defined either in the text or in the glossary.
Return	Indicates that you press the return key.
Ctrl/x	Indicates that you press the control key while you press the key noted by <i>x</i> .
[]	In command format descriptions, indicates optional elements. You can enter as many as you want.
{ }	In command format descriptions, indicates you must enter at least one listed element.

Part I

DECnet-Plus for OpenVMS Overview

Part I provides an overview of the transition from DECnet Phase IV to DECnet-Plus (Phase V). It also includes an overview of what is included with the DECnet-Plus for OpenVMS base system. This section includes the following chapters:

- Chapter 1 — Introducing DECnet-Plus for OpenVMS
- Chapter 2 — DECnet-Plus for OpenVMS Components

Introducing DECnet-Plus for OpenVMS

Digital's DECnet-Plus network is a family of hardware and software products that allows Digital operating systems to communicate with each other and with systems produced by other vendors.

The DECnet-Plus network supports remote system communication, resource sharing, and distributed processing. Network users can access resources on any system in the network as well as the resources of other vendors' systems on multivendor networks.

DECnet-Plus networking software provides true network independence. You get the full functionality of DECnet Phase IV, as well as DECnet enhancements plus full TCP/IP compatibility and OSI functionality.

DECnet-Plus for OpenVMS is Digital's OpenVMS implementation of:

- The Open Systems Interconnection (OSI) communications specifications, as defined by the International Organization for Standardization (ISO).
- Digital's communications architecture, Digital Network Architecture (DNA) Phase V, which is also backward compatible with the Phase IV architecture.

Phase V integrates the DNA and OSI layers. The DNA Phase V reference model is the architectural model on which DECnet-Plus networking implementations are based. DECnet-Plus also includes support for the internet standards RFC 1006 and RFC 1859, which allow you to run OSI and DECnet Phase IV applications over TCP/IP.

Table 1-1 shows the changes that have evolved with each new phase.

Table 1-1 DNA Phases

Phase I	Limited to two nodes
Phase II	Up to 32 nodes: file transfer, remote file access, task-to-task programming interfaces, network management
Phase III	Up to 255 nodes: adaptive routing, downline loading, record access
Phase IV	Up to 64,449 nodes: Ethernet local area networks, area routing, host services, OpenVMS Cluster support
Phase V	Virtually unlimited number of systems: OSI protocol support, transparent transport level links to TCP/IP, multivendor networking, local or distributed name service, distributed network management

DECnet-Plus for OpenVMS integrates DECnet and OSI network protocols which provides continued support for DECnet applications and enables support for OSI applications on OpenVMS. With separate TCP/IP software running on the same system, DECnet-Plus supports a multivendor, multiprotocol network

Introducing DECnet-Plus for OpenVMS

environment. DECnet applications can be run over NSP, CLNP, or TCP/IP transports. OSI applications can be run over CLNP or TCP/IP transports.

A full implementation of the Digital Network Architecture (DNA) Phase V for OpenVMS systems, the OSI component of the DECnet-Plus software, is implemented in accordance with the current U.S. and UK GOSIP requirements. GOSIP is the Government OSI Profile that defines OSI capabilities required by government procurement.

Note

Chapter 3 contains more conceptual information on the OSI Reference Model, OSI terminology (protocols, stacks, dialogues, entities, services), and how each specific layer relates to DECnet-Plus for OpenVMS.

1.1 Preparing for a Migration to DECnet-Plus

A number of automated tools (DECnet migration utilities and the NCP Emulator) and simplified configuration procedures are available to help you migrate to DECnet-Plus.

The complexity of the transition from Phase IV to Phase V architecture varies depending on the complexity of your existing network. In general, the smaller the network, the easier the transition. For larger networks, more planning is required. Nevertheless, whether your network is large or small, you can accomplish the transition without disrupting day-to-day operations.

DECnet-Plus now supports the fast configuration option, which a system or network manager can use to configure DECnet-Plus quickly on an OpenVMS system by invoking the `net$configure.com` procedure.

With the Fast Configuration option, you can configure a Phase IV system for network connectivity by answering a few questions. This feature is useful when you upgrade a DECnet Phase IV node to a DECnet-Plus node at some time in the future.

For information about how to configure your network using the Fast Configuration option, see the *DECnet-Plus for OpenVMS Installation and Basic Configuration* manual.

1.1.1 Differences Between DECnet Phase IV and DECnet-Plus Phase V

DECnet-Plus is the implementation of the fifth phase of the Digital Network Architecture (DNA). Phase V integrates the Open System Interconnection (OSI) protocols with DECnet protocols. In addition, Phase V includes support for the Internet standard RFC 1006 and the Internet draft RFC 1859, allowing OSI and DECnet applications to run over TCP/IP. Thus, applications that use DECnet-Plus can communicate with peer OSI and DECnet applications on any DECnet Phase IV-based system or OSI-based system, whether from Digital or from other vendors.

The addition of the OSI protocols and the TCP/IP communications capability is, therefore, the primary difference between DECnet Phase IV and DECnet-Plus Phase V. There are, however, other differences. DECnet-Plus contains many features designed to enhance networking capabilities. These new features include:

Introducing DECnet-Plus for OpenVMS

1.1 Preparing for a Migration to DECnet-Plus

- Global naming/directory services — In large networks, the number of nodes and users makes it difficult to manage any form of local directory service. Global (or distributed) name services allow large networks to easily store, manage, and access addressing information for (potentially) millions of network objects, such as end systems, users, printers, files, and disks. With the explosion of PCs as the desktop device of choice, networks containing millions of nodes have become a reality.
- Optional local naming/directory services — Smaller networks do not have as critical a need for global directory services.
- Expanded network management capabilities via the Network Control Language (NCL) — As networks become more complex, the management of networks has also become more complex. NCL enables network managers to access a wide range of manageable entities in the network for a wide range of management tasks.
- An improved routing algorithm (link state routing) — The larger a network becomes, the more overhead is generated by passing routing information between routers and between routers and end systems. Link state routing provides a more efficient algorithm for passing routing information while keeping the overhead traffic to a minimum. Link state routing is supported only on dedicated routers.
- Host-based routing — You can configure your network to enable host-based routing using the routing vector protocol. Host-based routing allows an OpenVMS system to operate as a DECnet-Plus intermediate system (IS). This feature is useful for those configurations where you need to route from a local area network (LAN) to a wide area network (WAN) and want to use an existing system to do the routing rather than investing in a dedicated router.
- Increased addressing — Networks can now grow to include potentially millions of nodes. DECnet-Plus uses the OSI standard address format. This format is designed to allow each node in a universal network to have a unique address. In other words, networks can now grow well beyond the bounds of any other addressing format currently in use. Existing Phase IV addresses can continue to be used for upgraded systems. The Phase IV address is moved into the OSI address format.
- Address autoconfiguration — DECnet-Plus nodes can take advantage of the address autoconfiguration feature where the adjacent router configures the node address for you.
- Single installation/configuration for OSI components — X.25, the Wide Area Device Drivers (WANDD), FTAM and Virtual Terminal (VT) applications are included in the DECnet-Plus H-Kit.

To fully benefit from these enhancements, you may need to make changes to your network. Prior to upgrading from DECnet Phase IV to DECnet-Plus, you must make certain decisions:

- Network addressing — Will users on the DECnet-Plus network have the ability to communicate with users on other OSI networks, either through electronic mail, EDI, FTAM, VTP, or other internetwork utilities? If yes, you must obtain a unique network identifier (IDP) from an authorized authority such as ANSI. If not, a default IDP is provided with DECnet-Plus that you can use at installation/configuration time.

Introducing DECnet-Plus for OpenVMS

1.1 Preparing for a Migration to DECnet-Plus

- Name services — Will name services be provided locally to each node or distributed throughout the network? Larger networks can benefit greatly from a distributed name service. The local service option is similar to Phase IV.
- DNA Phase V architecture has an overall management architecture called the **Enterprise Management Architecture (EMA)**. EMA defines a framework for the management of heterogeneous, multivendor distributed computing environments and the communications facilities that link them. EMA covers the entire distributed system, not just the communications aspects. The enterprise network comprises communication networks, computing systems, databases, and applications.

In conformance with EMA, DECnet-Plus provides distributed network management facilities that allow you to manage the network both in a local and distributed manner. The network management design is based on the director-entity framework and models defined by EMA. For an introduction to DECnet-Plus network management, see Section 3.10.

- The DECnet-Plus network management protocol is based on DNA **Common Management Information Protocol (DNA CMIP)** draft standard for network management operations. CMIP is used for encoding network management operations that can be performed on an entity. CMIP permits the exchange of information between a director and an agent. CMIP supersedes the Phase IV Network Information and Control Exchange (NICE) protocol.

DECnet-Plus for OpenVMS provides a callable interface for management operations. This interface, the CMIP Management Listener (CML), consists of CML\$ run-time routines. CML\$ routines perform the encoding of management data into CMIP and the decoding of data from CMIP, as well as interfacing to the entities.

DECnet Phase IV applications that have been written to create logical links to the Phase IV Network Management Listener (NML) and then parse the returned NICE protocol messages are not supported for managing DECnet-Plus for OpenVMS systems. To run on a network composed of DECnet-Plus systems, those applications must be rewritten to use DECnet-Plus network management software and protocols, such as CML and CMIP.

1.2 OSI and IETF Standards Supported

DECnet-Plus for OpenVMS implements standards approved by:

- The International Telegraph and Telephone Consultative Committee (CCITT)
- The Institute for Electrical and Electronic Engineers (IEEE)
- The Internet Engineering Task Force (IETF)

Table 1-2 lists the OSI standards that DECnet-Plus for OpenVMS supports. Note that for LANs at the Data Link and Physical layers, ISO standards are equivalent to IEEE standards.

Introducing DECnet-Plus for OpenVMS 1.2 OSI and IETF Standards Supported

Table 1–2 OSI Standards Supported by DECnet-Plus for OpenVMS

Layer	Standard	Description
	ISO 7498	Basic Reference Model
Application	ISO 8571	File Transfer, Access and Management (FTAM)
	ISO 8649	Service Definition — Association Control
	ISO 8650	Association Control Service Element (ACSE)
	ISO 9041	Virtual Terminal Protocol
	ISO 9072	Remote Operations Service Element
Presentation	ISO 8822	Presentation Service
	ISO 8823	Presentation (Kernel)
Session	ISO 8326	Connection-Oriented Network Service (CONS)
	ISO 8327	Connection-Oriented Network Service Protocol
	ISO 9595	Common Management Information Service Element (CMISE)
Transport	ISO 8072	Connection-Oriented Transport Service (COTS) Definition
	ISO 8073	Connection-Oriented Transport Service (COTS) Protocol
	RFC 1006	Defines how to implement ISO 8073 Transport Class 0 on top of TCP
	RFC 1006 Extension (Internet Draft)	Defines how to implement ISO 8073 Transport Class 2 non-use of Explicit Flow Control on top of TCP
Network	ISO 7498 Addendum 1	Connectionless-Mode Transmission
	ISO 8208	X.25 Packet Level Protocol (PLP)
	ISO 8348	Service definition: Connection-Oriented Network Service (CONS)
	ISO 8348 Addendum 2	OSI addressing formats
	ISO 8473, 8348 Addendum 1	Connectionless-Mode Network Service (CLNS)

(continued on next page)

Introducing DECnet-Plus for OpenVMS

1.2 OSI and IETF Standards Supported

Table 1–2 (Cont.) OSI Standards Supported by DECnet-Plus for OpenVMS

Layer	Standard	Description
	ISO 8878	X.25/Connection-Oriented Network Service (CONS)
	ISO 8881	X.25 Packet Level Protocol in local area networks
	ISO 9542	End-system to intermediate-system routing exchange protocol
Data Link	ISO 3309, 4335, 7809, 8471, 8885	Point-to-point data links (HDLC)
	ISO 7776	Point-to-point X.25 data links (only with an X.25 license; LAPB, one possible user)
	ISO 8802-1 (IEEE 802.1) ¹	Ethernet support (CSMA-CD)
	ISO 8802-2 (IEEE 802.2)	Frame formats for 8802-3 LANs (CSMA-CD logical link control (LLC1))
		X.25 logical link control (LLC2) (one possible user)
	ISO 8802-3 (CSMA/CD)	LAN support (CSMA-CD)
	ISO 9314	Fiber Distributed Data Interface (FDDI)
Physical	ISO 8802-3 (IEEE 802.3)	CSMA-CD devices
	ISO 9314	Fiber Distributed Data Interface (FDDI)
	EIA RS-232-C	HDLC point-to-point devices
	EIA RS-422	HDLC point-to-point devices
	EIA RS-423	HDLC point-to-point devices
	CCITT V.35	HDLC point-to-point devices

¹DECnet-Plus supports only the addressing specifications in this standard.

1.3 Dependencies and Licenses

The DECnet-Plus for OpenVMS software has several related dependencies outlined in the following sections.

1.3.1 DECnet Licenses

To use DECnet-Plus for OpenVMS or DECnet Phase IV software, you need the appropriate software licenses. Table 1–3 lists the two basic licenses and three license keys for OpenVMS VAX and Alpha systems, for DECnet Phase IV and DECnet-Plus, respectively.

Introducing DECnet-Plus for OpenVMS

1.3 Dependencies and Licenses

Table 1–3 DECnet Phase IV and DECnet-Plus Licensing

<i>VAX Phase IV</i>	<i>VAX DECnet-Plus</i>	<i>Alpha Phase IV</i>	<i>Alpha DECnet-Plus</i>
End System License			
DVNETEND	DVNETEND	DVNETEND	DVNETEND
All DECnet Phase IV functionality except cluster alias and routing	All DECnet Phase IV functionality except cluster alias, OSI API ¹ , OSI application gateways, DECdns server, and routing	All DECnet Phase IV functionality except cluster alias	All DECnet-Plus functionality except cluster alias, OSI API ¹ , OSI application gateways, and routing
Extended Function License			
DVNETRTG	DVNETRTG	DVNETEXT	DVNETEXT
All DECnet Phase IV functionality including cluster alias ² and routing	All DECnet-Plus functionality including cluster alias ² , OSI API, OSI application gateways, DECdns server, and host-based routing ³	All DECnet Phase IV functionality including cluster alias ² but excluding routing ⁴	All DECnet-Plus functionality including cluster alias ² , OSI API, OSI application gateways, and host-based routing ³
¹ Application programming interface			
² The DVNETRTG license is required on one node in the cluster to enable cluster alias			
³ Host-based routing is available on DECnet-Plus (Version 7.1); it was not available on DECnet Phase V (DECnet/OSI) systems before Version 7.1			
⁴ Routing is not supported on DECnet Phase IV OpenVMS Alpha systems			

1.3.2 Name Services

For mapping between node names and node addresses, you need at least one of the following three name services:

- Local namespace
- DECdns
- DNS/BIND

A DECnet-Plus network can use one name service exclusively, or it can have a mixture of systems using one or more of the name services. While configuring DECnet-Plus, you specify one or more of the three available name services to use on the node. To determine which name service(s) to use, check which name services are already being used by other nodes in your network. For example, if the other nodes in your network are already using DECdns, you will most likely want to use DECdns and join the existing namespace. The following sections include additional criteria and dependencies on the name services.

1.3.2.1 Choosing the Local Namespace

Choose the Local namespace if you have a small network and do not wish to use a distributed namespace. The Local namespace is similar to the permanent node database (NETNODE_REMOTE.DAT), used on DECnet Phase IV systems. With the Local namespace, name-to-address mapping information has to be administered separately on each node. To use the Local namespace, no additional software is required.

Introducing DECnet-Plus for OpenVMS

1.3 Dependencies and Licenses

1.3.2.2 Choosing DECdns

With DECdns, all node names in the network can be administered from one location. The mapping information is stored on two or more DECdns servers and kept up-to-date networkwide automatically. DECnet-Plus requires at least two DECdns servers in the network. DECdns server software must be installed and configured on these systems (the server software is optional software included with the DECnet-Plus for OpenVMS software kit). The *DECnet-Plus Planning Guide* describes planning considerations and the *DECnet-Plus for OpenVMS Applications Installation and Advanced Configuration* and *DECnet-Plus DECdns Management* guides include installation and configuration instructions.

1.3.2.3 Choosing DNS/BIND

DNS/BIND, the distributed name service for TCP/IP, supports the storage of IP addresses and the use of node synonyms. Node synonyms allow for backward compatibility with older applications that cannot use long domain names. (Note that DECnet-Plus also allows for node synonyms to provide backward compatibility with DECnet Phase IV node names.) DNS/BIND is needed if you want DECnet-Plus to run applications over TCP/IP. To use the DNS/BIND name service, DECnet-Plus requires one or more DNS/BIND servers in the network. DNS/BIND must be selected as one of the name services if you plan to use the DECnet over TCP/IP or OSI over TCP/IP features. See the appropriate TCP/IP documentation for more information on DNS/BIND.

1.3.3 DECdts Time Server

The Digital Distributed Time Service (DECdts) synchronizes the system clocks in computers connected by a network. The DECnet-Plus for OpenVMS configuration procedure autoconfigures the DECdts clerk. If your network uses multiple DECdns servers, or if you need network clock synchronization, Digital recommends that you install at least three DECdts servers on each LAN. See the *DECnet-Plus DECdts Management* guide for more information.

1.3.4 Intermediate System

In large networks and networks requiring high throughput, one or more dedicated routers are recommended for the network. Digital recommends using host-based routers to replace DECnet Phase IV host-based routers or in environments not requiring high throughput.

1.3.5 X.25

The DECnet-Plus for OpenVMS VAX systems license includes the right to use X.25 Access software (formerly known as VAX P.S.I. Access) or X.25 Native mode software (formerly known as VAX P.S.I.), which requires an additional license.

The X.25 software in DECnet-Plus for OpenVMS is backwards compatible with systems running the older VAX P.S.I. products. For further information on X.25, refer to Chapters 2 and 4.

On DECnet-Plus for OpenVMS Alpha systems, the following licenses are required:

- To run CONS over LLC2 or CLNS over Digital HDLC, a DECnet-Plus for OpenVMS Alpha license is required.
- To use LAPB to a WAN and to use any of the X.25 APIs or utilities over either LAN or WAN, an X.25 for OpenVMS Alpha systems license is required.

1.3.6 OSI Application Development

To develop OSI applications, you need to use the OSI application development interfaces (API) installed with the base system. These tools allow you to build network applications that adhere to the OSI standards defined by the International Organization for Standardization (ISO).

You can use the following components in building OSI applications:

- An application programming interface (API) to the File Transfer, Access and Management (FTAM) services within the Application layer
- The OSI Applications Kernel (OSAK) API, which provides access to:
 - Presentation layer services
 - The Association Control Service Element (ACSE) of the Application layer
- The Remote Operations Service Element (ROSE) of the Application layer

Where ISO standards exist, the APIs conform to these standards.

The following table shows the components available on Digital UNIX and OpenVMS platforms.

Operating System	Components
Digital UNIX	FTAM, OSAK (Presentation and ACSE)
OpenVMS	FTAM, OSAK (Presentation and ACSE), ROSE (VAX only)

1.4 Distributed Networking

A DECnet-Plus network can be viewed as a distributed processing system. The major functions of the network, such as network management and routing, are not centralized in a single system. Each system can manage both itself and remote systems. Adaptive routing eliminates the need to set up network data paths.

Some of the primary features of the DECnet-Plus distributed network are:

- Optional use of distributed system services for networkwide names and synchronized time
- Support for distributed network applications

1.4.1 Distributed System Services

Networkwide capabilities include the optional use of distributed system services designed to make the network as transparent as possible to users and applications. The DECnet-Plus distributed computing environment provides the following services:

- Global naming (see the *DECnet-Plus Planning Guide*)
- Time synchronization (see the *DECnet-Plus Planning Guide*)

Introducing DECnet-Plus for OpenVMS

1.4 Distributed Networking

1.4.2 Distributed Network Applications

In the DECnet-Plus distributed processing environment, you can physically distribute multiple resources or tasks that perform various functions between systems on the network.

A **distributed application** is a collection of processes that use resources, such as processing elements, databases, and physical devices, located on other systems in the network. As a single logical application, the elements or tasks are physically divided. A **task** is a modular component of work that the application programmer defines within an application. The work of a distributed application is divided among tasks that can communicate with each other.

In an OpenVMS operating system, a task is executed within the context of a process. The **process context** defines the environment in which the task executes. OpenVMS software controls the access and allocates the resources required by the task, based on the process context.

In a distributed application, each task is distinct and can be placed in different locations in the network. The system interface for the application allows you to run the application locally or remotely without any apparent difference.

You can distribute an application so that each task is assigned to a system with appropriate resources. For instance, one task computes on a powerful processor while another stores the information in a database on a system with extensive disk storage capabilities. A common example of a distributed application is an implementation of the client/server model, such as DECdns, in which the client task (on the DECdns clerk system) requests service from the server task on a different system (the DECdns server).

Interprocess communication is the movement of data and control from one task running within a process to another task in the network. Interprocess communication allows the various tasks on the different processes to cooperate and communicate with each other, exchanging message packets over the connection established between the tasks.

Distributed applications can increase availability. You can design a distributed application to avoid a single point of failure by moving tasks to other processors if a processor fails. Other benefits of distributed applications include:

- Load and resource sharing
- Reduced communication costs
- Modular, incremental growth capabilities
- Configuration flexibility capabilities

1.5 OpenVMS Cluster Systems

An OpenVMS cluster configuration is an organization of OpenVMS operating systems that communicate over a high-speed communications path and share processor resources as well as disk storage. DECnet connections are required for certain OpenVMS cluster tools and configurations. DECnet-Plus for OpenVMS software provides support for OpenVMS cluster systems.

1.5.1 DECnet-Plus OpenVMS Cluster Alias

DECnet-Plus for OpenVMS supports the use of multiple cluster aliases. The alias node identifier (a node name or node address) is common to some or all nodes in the cluster and permits users to address it as though it were one node.

For management purposes, the cluster alias is viewed by the DECnet-Plus software as a separate Node entity that is manageable through NCL commands. The Alias entity differs from a regular Node entity in some characteristics; for example, the Alias entity does not support a Circuit entity. The cluster alias appears to the network as a multicircuit end node, which is an end node with several active circuits. In an OpenVMS cluster system that consists of DECnet-Plus for OpenVMS systems on a LAN, the alias node appears as an end node with multiple points for attachment to the LAN.

DECnet-Plus for OpenVMS supports the ability to access nodes in an OpenVMS cluster using a separate alias node address, while retaining the ability to address each node in the cluster individually. Not all network objects may be accessed using this mechanism. The maximum number of nodes supported for a cluster alias is 96. The maximum number of cluster aliases for a single node is three.

1.5.2 OpenVMS Cluster Configuration Procedure

The `cluster_config.com` command procedure for performing a OpenVMS cluster configuration invokes the DECnet-Plus for OpenVMS `net$configure.com` command procedure to perform any required modifications to NCL initialization scripts. Use `cluster_config.com` to create a configuration for each satellite node in the cluster.

DECnet-Plus for OpenVMS Components

2.1 Features of DECnet-Plus for OpenVMS

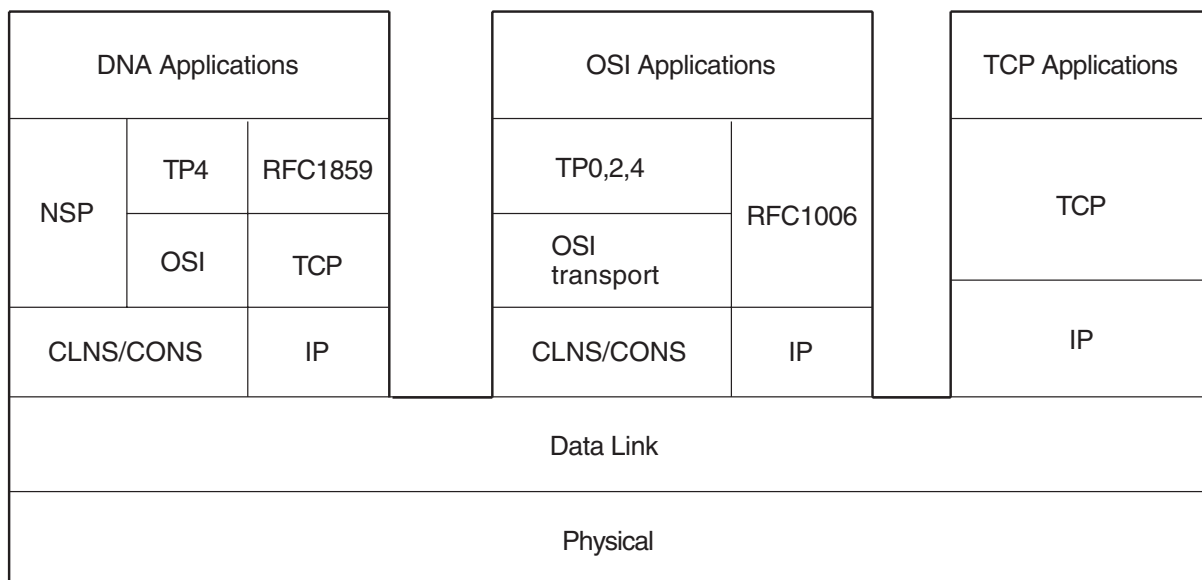
DECnet-Plus for OpenVMS is an implementation of Phase V of the Digital Network Architecture (DNA) for the OpenVMS operating system.

DECnet-Plus integrates DECnet and OSI network protocols allowing both stacks to share integrated network functions up to the Transport layer. Upper layers have been implemented as separate "towers," allowing existing DECnet and OSI applications to share the integrated Transport layer. Existing DECnet Phase IV and new DECnet and OSI applications are supported by DECnet-Plus. In combination with TCP/IP protocol stacks, OpenVMS systems can participate in multivendor, multiprotocol networks adhering to Open Networking standards.

For details on specific upgrades and enhancements, refer to the *DECnet-Plus for OpenVMS Release Notes*.

Refer to Figure 2–1, which illustrates the integrated DNA Phase V Reference Model.

Figure 2–1 DNA Phase V DECnet-Plus for OpenVMS



ZK-8978A-GE

DECnet-Plus for OpenVMS Components

2.1 Features of DECnet-Plus for OpenVMS

2.1.1 OpenVMS Software Components

DECnet-Plus for OpenVMS offers task-to-task communications, file management, and downline system and task loading. Network WAN connectivity is provided by WAN device drivers supporting host-based synchronous communications options for wide area networking.

DECnet-Plus for OpenVMS is available in two forms: End System and Extended Function. Extended Function provides all the features of End System plus the OSI application gateways (FTAM–DAP Gateway, VT-Telnet, LAT/VT and VT/LAT), the DECdns server (on VAX platforms), and host-based routing, and cluster alias.

The functions offered by OpenVMS are split into separate components. The entire base system installs automatically when you start the installation procedure, while the additional components are optionally installable. To support open, standards-based, multiprotocol communications, DECnet-Plus for OpenVMS is comprised of the DECnet-Plus base system (including DECdns and DECdts clerk software), and the following optional software:

- DECdts server—Software that synchronizes the system clocks in computers connected by a network.
- WAN device drivers—Software that links a hardware device and layered software for synchronous communications over wide area networks.
- X.25—Allows suitably configured DECnet-Plus systems to connect to Packet Switching Data Networks (PSDNs) conforming to CCITT recommendation X.25 (1980 or 1984) and/or to International Standards (ISO) 7776 and 8208. This allows a DECnet-Plus system to function as data terminal equipment (DTE) or be addressed as data circuit terminating equipment (DCE). On Alpha systems, X.25 is obtained separately from the DECnet-Plus software kit.
- Open System Application Kernel (OSAK) software—Digital’s implementation the OSI network architecture components that provide services for OSI applications.
- File Transfer, Access, and Management (FTAM) software—Digital’s OSI application that allows you to perform file operations between OSI-compliant systems.
 - DAP–FTAM gateway—Allows your DECnet-Plus node to participate in an OSI network in a simplified manner; for example, you can issue DCL commands to communicate with remote OSI systems through the gateway. Communication with a remote FTAM application is handled the same as any DECnet dialogue.
- Virtual Terminal (VT)—Digital’s OSI application that enables application and systems supporting different types of terminals to interoperate with each other. Using VT, you can use your terminal to access any other system running VT, regardless of the type of system. You can also use the VT gateways for access to and from non-OSI systems.
 - LAT/VT and VT/LAT gateways—Allows connections between a LAT terminal and OSI systems.
 - VT/Telnet and Telnet/VT gateways—Allows communications between OSI and Internet systems.

DECnet-Plus for OpenVMS Components

2.1 Features of DECnet-Plus for OpenVMS

Table 2–1 lists the functions of DECnet-Plus for OpenVMS matched with the specific software components that you need to install and/or configure during the DECnet-Plus for OpenVMS installation procedure. Other dependencies are also noted.

Table 2–1 Functions and Associated Software

Function	Software (and other dependencies)
OSI Transport Service classes 0, 2, 4 (TP 0, TP 2, TP 4)	Base system (configuration of OSI Transport software)
RFC 1006 RFC 1006 Extension (Internet Draft)	Base system (configuration of OSI Transport software) TCP/IP software
RFC 1859	ISO Transport Class 2 Non-use of Explicit Flow Control over TCP RFC 1006 extension
Routing (ES-IS)	Base system (and, to communicate beyond the local subnetwork, a DECnet-Plus-compliant intermediate system)
File operations to and from remote OSI systems	FTAM, OSAK software
VT to OSI systems	VT, OSAK software
Function as OSI end system	FTAM, VT, Digital-provided OSI application, or customer-written applications
Run any OSI application <i>other than FTAM, VT</i>	OSAK software
ACSE Presentation and Session	OSAK software
OSAK server (inbound connection handler)	OSAK software
WAN X.25 communications on local system	X.25 software
WAN X.25 communications via a connector node†	X.25 software
LAN X.25 communications (LLC2)	X.25 software
DECnet-Plus for OpenVMS network management	Base system
Network management support tools	Base system
Node-name management support tools	Base system
FTAM and VT gateways	Gateways software (included in FTAM), Extended Function license; for Telnet/VT gateway, also Digital TCP/IP Services for OpenVMS
DECdns clerk, DECDts clerk	Base system
DECDts server	Base system

†The connector node can be a DECNIS router 500 or 600, or X.25 configured in multihost mode.

2.2 DECnet-Plus for OpenVMS Base System

The DECnet-Plus for OpenVMS base system software allows an OpenVMS system to perform as both a DECnet end node and an OSI end system. It can communicate using ISO 8802-3 (CSMA-CD) or ISO 8802-5 (FDDI) broadcast lines, either Digital Data Communications Message Protocol (DDCMP) or point-to-point through HDLC lines.

DECnet-Plus for OpenVMS Components

2.2 DECnet-Plus for OpenVMS Base System

The DECnet-Plus base software offers the following features:

1. End system implementation of the first four layers of the OSI Reference Model and all the layers of DNA
2. Host-based routing (that replaces DECnet Phase IV routing vector host-based environments not requiring high throughput)
3. Application, Presentation, and Session layers
 - File Transfer, Access, and Management (FTAM) application
 - Virtual Terminal (VT) application
 - Application Service Elements (ASEs), including ACSE (Association Control Service Element)
4. DNA Session Control layer (see Section 3.5)
 - Interprocess Communication (\$IPC) programming interface
 - Queue Input/Output (\$QIO) programming interface
 - Common Management Information Service Element (CMISE) programming interface
 - Support for DNA-based, Digital and user applications
5. Transport layer, classes 0, 2, 4 (see Section 3.6)
 - OSI Transport Service
 - Digital Network Services Protocol (NSP) Transport Service
 - RFC 1006 and RFC 1006 Extension (Internet Draft)
6. Lower layers
 - OSI addressing formats, supporting very large network topologies
 - End-system-to-intermediate-system (ES-IS) routing
 - Connectionless-mode Network Service (CLNS) over local area network (LAN), wide area network (WAN), and X.25
 - Logical link control type 2 (LLC2) for Connection-Oriented Network Service (CONS) over LAN
7. Network layer (see Section 3.7)
 - OSI-compliant addressing formats; virtually unlimited network size
 - ES-IS routing
 - CLNS which supports OSI Transport class 4 over ISO 8802-3 and X.25 connections
 - Internetwork Protocol to support CLNS
 - CONS which supports OSI Transport classes 0, 2, and 4 over ISO 8802-3 and X.25 connections
8. Data Link layer (see Section 3.8)
 - Supports High-level Data Link Control (HDLC) for wide area communications, ISO 8802-3 (Ethernet CSMA-CD) and FDDI LANs, and Digital Data Communications Message Protocol (DDCMP) for backwards

DECnet-Plus for OpenVMS Components

2.2 DECnet-Plus for OpenVMS Base System

compatibility. HDLC support includes the Link Access Protocol Balanced (LAPB) protocol for X.25 communications.

- ISO 8802-2 logical link control type 1 (LLC1)
 - FDDI driver support
9. Physical layer (see Section 3.9)
 - CSMA-CD, HDLC, and FDDI devices supported
 10. Network management (see Section 3.10)
 11. Network management support tools (see Section 2.8)
 12. Node-name management support tools (see Section 2.8)

2.2.1 DECnet Over TCP/IP

The DECnet over TCP/IP feature allows users to run standard DECnet applications in an Internet Protocol (IP) routing environment.

When you are running DECnet over TCP/IP, you can connect using an IP address, for example:

```
$ set host node.site.company.com
```

The connection is made using the DNA CTERM application instead of the TCP application FTP. Note that both the source and target nodes must support DECnet over TCP/IP for this connection to work. The system can be configured to allow you to use synonyms (Phase IV-style names) instead of the IP host full name. If a system does not run DECnet and support DECnet over TCP/IP, you need to use COPY/FTP to access that system. For example:

```
$ COPY/FTP myfile *
```

For more information on running DECnet over TCP/IP, refer to the *DECnet-Plus Network Management* and the installation and configuration guides. This feature requires any valid DECnet license and a licensed and installed TCP/IP product that supports the PATHWORKS Internet Protocol (PWIP) interface.

2.2.2 Digital-Supplied Session Control Applications

A DECnet Phase IV "object" in DECnet-Plus is called an **application**. The DECnet-Plus for OpenVMS base system supplies the following session control applications, which are defined automatically at network startup:

- Task — Image that provides full Phase-IV compatibility.
- File Access Listener (FAL) — Image that provides authorized access to the file system of a DECnet-Plus system on behalf of processes executing on any network system. FAL communicates with the initiating system by means of the Data Access Protocol (DAP).
- CMIP Management Listener (CML) — Image that allows remote management of the local system.
- Loopback mirror (MIRROR) — Image used for some loopback testing.
- OpenVMS Mail — Image that provides mail service for DECnet-Plus for OpenVMS systems.
- OpenVMS Phone — Image that allows you to communicate with other users on your system or with any other connected DECnet-Plus system.

DECnet-Plus for OpenVMS Components

2.2 DECnet-Plus for OpenVMS Base System

- OpenVMS Cluster Performance Monitor (VPM) Server — Image used to run Open VMS Cluster monitoring features of the Monitor utility (MONITOR).

2.2.3 Programming Interfaces

The DECnet-Plus for OpenVMS base system offers the following programming interfaces.

- Common Management Information Element Service (CMISE) interface
DECnet-Plus for OpenVMS CMISE application programming interface provides the mechanism for user-written applications to perform OSI network management, and implements the ISO 9595 Common Management Information Service specification on OpenVMS. Two cooperating management applications in an OSI network are known as CMISE service users. These service users establish an association and exchange management information by means of the CMISE interface.

The CMISE interface provides three types of management services:

- Management Association Services for the establishment and release of associations between service users
- Management Operation Services for the transfer of management information between service users
- Management Notification Services for the reporting of events between service users

- Interprocess Communication (\$IPC) interface

DECnet-Plus for OpenVMS \$IPC system service is an OpenVMS operating system interface to the Session Control layer that can be used to perform interprocess communication. This interface is a standard OpenVMS system service call.

With the \$IPC system service, distributed applications can run in a variety of environments without modification. \$IPC operates over both local area and wide area connections. \$IPC provides for efficient communication within a single DECnet-Plus for OpenVMS system, and between a DECnet-Plus for OpenVMS system and any other system running DECnet-Plus software in either a DECnet-Plus or multivendor network environment.

\$IPC provides basic connection, data transfer, and information management services. It permits an association to be opened between the application and the DNA Session Control layer. The \$IPC interface can then create or accept connections over the opened association. \$IPC can receive and process multiple inbound connection requests destined for a single application.

Connection to the remote target can be made through specification of:

- The DECdns object name of the target application, if your network uses the distributed DECdns namespace. For this connection, Session Control transparently selects the protocol tower information to be used based on the protocol and address information supplied by DECdns for the application service.
- Protocol towers for the target application (as stored in the DNA\$Towers attribute) that specify which address and protocol is to be used to communicate with the application.

DECnet-Plus for OpenVMS Components

2.2 DECnet-Plus for OpenVMS Base System

- A Phase IV or DECnet-Plus node name and application identification. (The target application can be identified by the internal version of a DECnet-Plus full name properly formatted for the Local namespace on DECdns or by a Phase IV object number or task name.)

The primary functions of \$IPC include:

- Connection setup services — Opens an association between an application and Session Control and declares an application name for incoming connections. (Also performs the function of shutting or closing associations.)
 - Connection services — Requests a logical connection to a target task, identifying the target application in the connect initiate request.
 - Control services — Receives incoming connect initiate requests, associates the requests with the calling process, and accepts or rejects the connection. (Also disconnects or aborts connections.)
 - Data transfer services — Transmits and receives messages, including expedited data messages (messages that bypass flow control mechanisms). Receives unsolicited network events, such as third-party disconnects.
 - General services for managing information about the system and connections — Resolves names, obtains information about a current connection, enumerates local towers, and supports backward translation of an address.
- Queue Input/Output (\$QIO) interface

DECnet-Plus for OpenVMS continues to support transparent and nontransparent task-to-task communications applications that use \$QIO system service calls in the same way as for Phase IV. If a 6-character limit on node names is encoded in the Phase IV application, use a Phase IV-style node synonym for both incoming and outgoing connections.

In addition, DECnet-Plus for OpenVMS supports \$QIO calls that are specifically required for applications coded to the OSI Transport Service.

- Remote Access Interfaces

With DECnet-Plus for OpenVMS, you can perform network operations as a natural extension of the input/output operations on the local system. In addition, a DECnet-Plus for OpenVMS programmer can write command procedures and programs that make the network transparent to users.

You can access remote files and tasks using DCL commands and command procedures, higher-level language input/output statements, or OpenVMS Record Management Services (RMS). For example, programs written in COBOL, C, Ada, MACRO, Fortran, BASIC, Pascal, or PL/1 can access remote files.

2.2.4 OSI Transport Service

A DECnet-Plus for OpenVMS system functions as an OSI end node if, during configuration, you configure OSI Transport. An application can set up a transport connection to another application on any other system, either Digital or multivendor, running software that implements the OSI Transport Protocol.

DECnet-Plus for OpenVMS Components

2.2 DECnet-Plus for OpenVMS Base System

2.2.5 NSP Transport Service

DECnet-Plus for OpenVMS software includes the Network Services Protocol (NSP) for DECnet Phase IV compatibility. You can use the proprietary NSP and the open OSI Transport Protocols simultaneously.

2.2.5.1 End-System-to-Intermediate-System (ES-IS) Routing

The ISO End-System-to-Intermediate-System (ES-IS) Routing Exchange Protocol provides the process by which end systems communicate with intermediate systems (routers), or with each other, to exchange configuration information.

You can configure a LAN with all end systems. In end-system-only configurations, the DECnet-Plus systems use ES-IS routing to communicate directly with each other. They use a specific multicast address to which all end systems listen. With this routing process, the concept of adjacency does not exist.

2.2.6 Network Management

DECnet-Plus network management offers the following benefits:

- Consistency across the Digital Network Architecture (DNA), which in turn allows products developed by different sources to be managed in a consistent way.
- A modular design, which allows systems to be as simple or as complex as appropriate to the services they provide their users.
- Extensibility, which allows new functions to be added to DNA and managed consistently with existing functions.

You can perform the following network management functions:

- Controlling the network
- Providing host services to other DECnet-Plus systems
- Providing host services to DECnet Phase IV nodes
- Establishing DECnet-Plus configurations

Section 3.10 introduces the concepts of DECnet-Plus network management.

2.2.7 Name Service

The DECnet-Plus Session Control layer requires a name service to map node names to addresses. A DECnet-Plus network can use one namespace exclusively, or it can have a mixture of systems using the Local namespace, the DECdns distributed namespace, and/or DNS/BIND.

DECnet-Plus provides access to the node name and addressing information stored in one or more of the following name services:

- The **Local namespace** is a discrete, nondistributed namespace that stores name and address information locally in database files. It maintains a database of names and addresses on the local node. Use a Local namespace if you decide not to use a distributed namespace, or if you decide to delay full planning and implementation of a distributed namespace. The Local namespace is designed to scale up to at least 100,000 nodes depending on the number of address towers stored.

DECnet-Plus for OpenVMS Components

2.2 DECnet-Plus for OpenVMS Base System

- The **Digital Distributed Name Service** (DECdns) software is a distributed, global name service that allows users and applications to assign unique names to resources and then use these resources without knowing their physical location.

With DECdns, each DECnet-Plus for OpenVMS system that does not use a Local namespace will be configured as a DECdns clerk. DECdns clerks request servers to provide address mapping for them. Because DECnet-Plus for OpenVMS does not contain DECdns server software, servers running on Digital UNIX or OpenVMS systems must handle clerk requests.

The DECnet-Plus for OpenVMS configuration procedure autoconfigures the DECdns clerk software.

- The **Domain Name System** (DNS/BIND) is the distributed name service for TCP/IP and supports the storage of IP addresses. In addition, support for DNS/BIND provides for the use of node synonyms. This allows for backward compatibility with older applications that cannot use long domain names.

There are two ways to configure node synonyms for use with DNS/BIND:

- By constructing an appropriate set of naming search path templates.
- By defining local aliases.

While configuring DECnet-Plus, the system administrator specifies one or more of the following name services to use on the node: the Local namespace, DECdns, or Domain (for DNS/BIND).

DECnet-Plus includes a new in-memory naming cache to improve performance of name and address resolution for all supported name services.

DECnet-Plus includes features to ease namespace management including `decnet_register` (a namespace management tool), numerous NCL commands, and Common Trace Facility (CTF) support for monitoring node name and address resolution. In some instances, this simplified access to multiple name services is referred to as CDI or the common directory interface.

In addition to the Local namespace and DECdns, for node-name-to-address mapping your network can use, if it has one, an existing VAX Distributed Name Service (DNS) Version 1 namespace.

2.2.7.1 Local Namespace

The Local namespace is a discrete, nondistributed namespace that exists on a single node and provides that node with a local database of name and addressing information. Depending on the number of address towers stored, the Local names are designed to scale to at least 100,000 nodes.

The DECdns distributed namespace is not a requirement of DECnet-Plus, and is not dependent on DECdns. However, the DECdns clerk software is still required on each node. Use `decnet_register` to manage the node name and address information stored in your namespace. The `decnet_register` tool is described in the *DECnet-Plus for OpenVMS Network Management* guide.

DECnet-Plus for OpenVMS Components

2.2 DECnet-Plus for OpenVMS Base System

2.2.8 Time Service

The Digital Distributed Time Service (DECdts) synchronizes the system clocks in computers connected by a network. DECDts enables distributed applications to execute in the proper sequence even though they run on different systems.

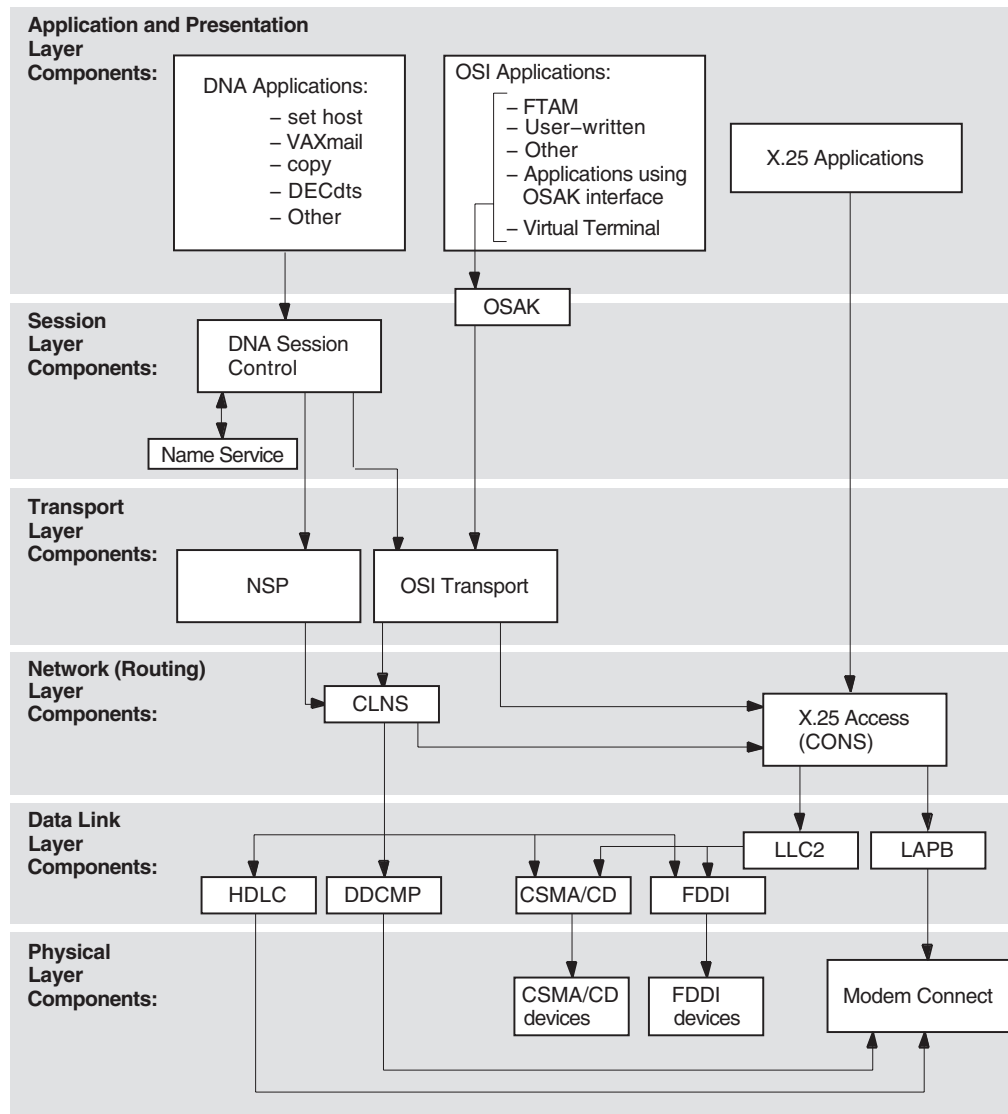
The DECnet-Plus for OpenVMS configuration procedure autoconfigures the DECDts clerk software. If your network uses multiple DECdns servers, or if you need network clock synchronization, Digital recommends that you install at least three DECDts servers on each LAN.

2.3 X.25

X.25 is a CCITT recommendation that specifies the interface to packet switching data networks (PSDNs). It implements the lower three layers of the OSI Reference Model. Conceptual information about X.25 is provided in Chapter 4, and platform-specific information is provided in the *X.25 for OpenVMS Management Guide*.

Figure 2-2 shows the relationship between the individual components in the DECnet-Plus environment on an OpenVMS system.

Figure 2-2 Component Relationships



LKG-09679-94R

2.3.1 X.25 Native Software

The X.25 native software allows suitably configured DECnet-Plus systems to connect to packet switching data networks (PSDNs) conforming to CCITT recommendation X.25 (1980 or 1984) and/or to international standards ISO 7776 and 8208. This allows a DECnet-Plus system to function as data terminal equipment (DTE) or be addressed as data circuit-terminating equipment (DCE) as follows:

- A packet-mode DTE connected to a supported PSDN conforming to CCITT Recommendation X.25 (1980, 1984)
- A packet-mode DTE connected to a CSMA-CD LAN conforming to ISO 8802-3 using ISO logical link control type 2 (LLC2) as specified in ISO 8881
- A packet-mode DTE/DCE connected to a DCE/DTE conforming to ISO 7776 and 8208

DECnet-Plus for OpenVMS Components

2.3 X.25

X.25 can be configured for native operation to support direct connections from a VAX system to one or more PSDNs, each of which may have one or more DTEs. It also allows communication with any non-Digital X.25 system on the same LAN that supports the ISO logical link control type 2 (LLC2) protocol.

2.3.2 X.25 Access Software

X.25 Access uses a connector node to provide the physical connection to a PSDN. A connector node can be a DECNIS router 500 or 600 or X.25 configured in multihost mode.

DECnet-Plus logical links are established by OpenVMS to connect the X.25 Access system to the X.25 Connector system. These links may use any supported DECnet-Plus communications path between the X.25 Access system and the Connector system, provided they themselves do not use an X.25 connection. X.25 Access uses these links to transmit X.25 or X.29 messages between the Connector system and the X.25 Access system.

X.25 software provides Connection-Oriented Network Service (CONS) to allow mapping between a destination NSAP address and a destination DTE address according to ISO 8348.

The X.25 software can be used in the following ways:

- Process-to-process (X.25) communications
- Process-to-terminal (X.29) communications
- Terminal-to-process (X.29) communications

2.4 Wide Area Network Device Drivers

The WAN device drivers included in DECnet-Plus for OpenVMS support synchronous communications. The device drivers all offer a supported user (\$QIO) interface.

The device drivers all support full-duplex and half-duplex operation, where appropriate to the protocol. See the Software Product Description (SPD) for supported device drivers.

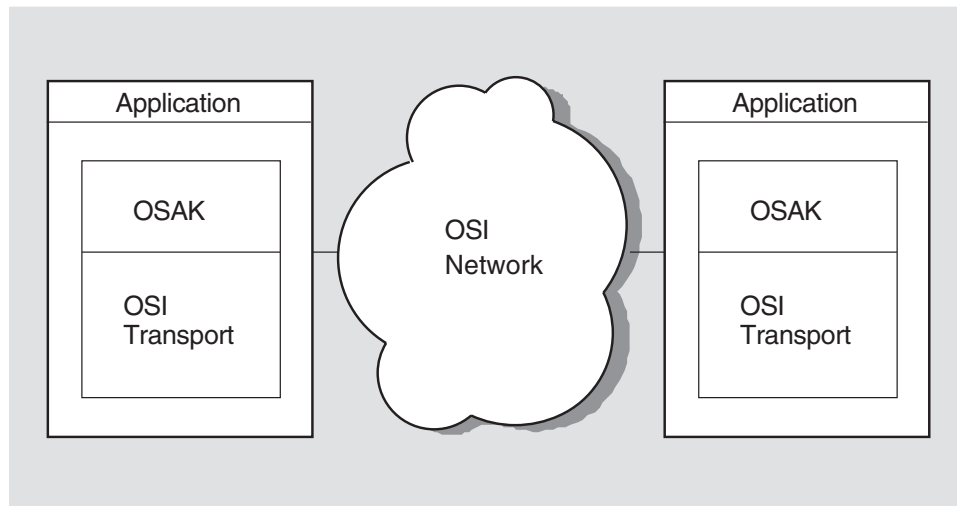
WAN device drivers include a pseudodriver (WANDRIVER) that provides a programming interface to the data link level for the LAPB, Digital HDLC, and DDCMP protocols.

2.5 OSAK

The OSI Applications Kernel (OSAK) software allows you to run applications that can communicate with other applications running in an OSI environment. These applications can be Digital products, or they can be applications that you have written specifically for an OSI environment using the OSAK API. Writing OSI applications requires the OSI Application Developer's Toolkit which is installed with the base system.

To run a Digital application using OSAK software, install the OSAK software wherever you run the application. For example, an application running on two DECnet-Plus systems uses the OSAK software on both systems, as shown in Figure 2-3.

Figure 2-3 OSAK Software on DECnet-Plus for OpenVMS Systems



LKG-6717-93R

2.5.1 OSAK API

The OSAK API provides routines you can use, linked with your own programs, to manage interactions between two open systems. The OSAK software is Digital's implementation of:

- The Session layer
- The Presentation layer
- The Association Control Service Element (ACSE) of the Application layer

The OSAK API is made up of three sets of routines:

- A set for making outbound requests and responses
- A set for receiving inbound indications and confirmations
- A set of housekeeping routines, which do not result in protocol exchanges

All calls to OSAK services are nonblocking because the OSAK API allows data to be passed to OSAK either in a single call or spread over `osak_select`, which you can use to block until a service request is complete.

You can check for completion of a nonblocking call by polling or, on an OpenVMS system, by asynchronous event notification. Asynchronous event notification is not available on an ULTRIX or Digital UNIX system.

The OSAK programming guides contain more information about blocking and nonblocking calls.

OSAK uses a parameter block interface. You can allocate memory for the parameter block and the data structures it contains either statically or dynamically. The parameter block contains all possible parameters for all possible services. OSAK uses only the relevant parameters in each service call.

You must always pass parameters between your application and OSAK in encoded form. On OpenVMS or Digital UNIX systems, you can use an ASN.1 (Abstract Syntax Notation One) compiler to help you do this.

DECnet-Plus for OpenVMS Components

2.5 OSAK

2.5.2 OSAK Software and Management Facilities

OSAK software consists of:

- An OSI upper layer protocol machine (see Section 2.5.2.1)
- Network control and management facilities (see Section 2.5.2.2)
- OSAK trace utility (see Section 2.5.2.3)

2.5.2.1 Protocol Machine

The protocol machine contains data structures and routines that implement the Session, Presentation, ACSE, and ROSE protocols, according to the standards listed in Table 2–2.

Table 2–2 OSAK Software: Summary of ISO Standards Implemented

Standard	Title
ISO 8327	<i>Information Processing Systems — Open Systems Interconnection — Basic Connection Oriented Session Protocol Specification</i>
ISO 8823	<i>Information Processing Systems — Open Systems Interconnection — Connection Oriented Presentation Protocol</i>
ISO 8650	<i>Information Processing Systems — Open Systems Interconnection — Protocol Specification for the Association Control Service Element</i>
ISO 9072–1	<i>Information Processing Systems — Text Communication — Remote Operations — Part 1</i>
NIST Version 3	<i>NIST Special Publication 500-177 — Stable Implementation Agreements for Open Systems Interconnection Protocols, Version 3, Edition, 1 December 1989</i>

2.5.2.2 Network Control and Management Facilities

The OSAK software is largely automatic, but you may want to manage the software for two purposes:

- Monitoring the OSI network
- Creating a passive address (see Section 2.5.3)

You can use Network Control Language (NCL) commands to manage the OSAK software.

Address management facilities for applications that run over OSI Applications Kernel software are provided through the OSAK module entity. The OSAK module entity is an immediate subordinate of the global entity `node`. Note that OSAK does not provide facilities for managing the applications themselves.

You can find details of NCL command syntax for the OSAK module entity and its subordinate entities in the *DECnet-Plus Network Control Language Reference* guide or in the NCL on-line help.

2.5.2.3 OSAK Trace Utility

The OSAK trace utility consists of two components: the trace emitter and the trace analyzer. This utility enables you to trace the activity of the protocol machine, and can help you track the source of any problems that occur when applications are running.

You can use the OSAK trace utility to trace protocol activity in the OSI upper layers:

- At the ACSE level, trace information includes the ACSE protocol control information (ACSE-PCI) and user data. User data is traced in either hexadecimal or generic ASN.1 form.
- At the Presentation layer, you can trace session service data units (SSDUs) containing user data and presentation protocol control information (PPCI).
- At the Session layer, you can trace transport service data units (TSDUs) passed between the OSAK software and the transport service, which contain session protocol control information (SPCI) and user data.

2.5.3 Active and Passive Addresses

An OSAK address can be either active or passive.

- **Active**

An active address is available continuously to receive incoming connections. As soon as the OSAK software receives an incoming connection, the software passes the contents of the call directly to the appropriate application.

- **Passive**

A passive address is registered in NCL in an osak application invocation subentity. As soon as the OSAK software receives an incoming connection, the software confirms the transport connection and creates a process to handle the call. See the *DECnet-Plus Network Control Language Reference* guide for details of setting up a passive address. Also see the *DECnet-Plus OSAK Programming* guide for details of the advantages and disadvantages of active and passive addresses.

2.6 FTAM

The File Transfer, Access, and Management (FTAM) software in DECnet-Plus for OpenVMS implements several standards that define the upper three OSI layers of the OSI Reference Model, including FTAM and ACSE components of the Application layer.

FTAM performs file operations and management between OSI-compliant systems. When implemented by different vendors, the FTAM standard enables the use of files on these vendors' systems. FTAM can transfer unstructured files with binary data and sequential text files with either a stream or variable record format.

To create and manage local files, FTAM uses the OpenVMS Record Management Services (**RMS**).

FTAM offers the following features:

- FTAM operates on files stored on both local FTAM systems (local files) and remote FTAM systems (remote files).
- Using the DCL interface, you can copy, append, delete, rename, and inspect the file attributes of files on OSI-compliant FTAM systems.
- Journaling supports FTAM's recovery and restart capability by maintaining a docket of recovery information. In the event of a recoverable error, FTAM tries to negotiate with the peer FTAM system for a recovery or restart.

DECnet-Plus for OpenVMS Components

2.6 FTAM

2.6.1 FTAM Gateway

The DAP–FTAM gateway allows a DECnet-Plus node to participate in an OSI network in a simplified way. You can think of this software as a server that receives Data Access Protocol (DAP) messages through DECnet and uses that information to establish and maintain a connection with a remote FTAM system.

The DAP–FTAM gateway simplifies communications for DECnet-Plus nodes because users can issue DCL commands to communicate with remote OSI systems through the gateway. For the DCL user, communication with a remote FTAM application is handled the same as any DECnet dialogue.

2.7 Virtual Terminal (VT)

The DECnet-Plus for OpenVMS Virtual Terminal (VT) software in DECnet-Plus for OpenVMS is an implementation of the ISO Virtual Terminal Protocol (VTP):

- ISO 9040 Virtual Terminal Basic Class Service
- ISO 9041-1 Virtual Terminal Basic Class Protocol — Part 1: Specification

VTP is an Applications layer protocol. This protocol allows remote logins and access to remote applications between DECnet-Plus systems and any remote systems, including multivendor systems, that also run an ISO-compliant VT implementation.

VT allows you to access remote OSI systems from a Digital terminal in the following ways:

- Entering a Digital Command Language (DCL) command at your terminal
- Using the Telnet/VT gateway
- Using the VT/Telnet gateway
- Using the LAT/VT gateway
- Using the VT/LAT gateway

For VT concepts information, see the *DECnet-Plus FTAM and Virtual Terminal Use and Management* guide.

2.7.1 VT Gateways

The VT gateways allow any DECserver terminal server, DECnet node with LATmaster, or TCP/IP Telnet node to participate in an OSI VT network. These gateways can be thought of as servers that receive messages of one protocol and use that information to establish and maintain a connection with a remote system using another protocol:

- Telnet/VT gateway — Telnet messages are received and translated into VTP messages.
- VT/Telnet gateway — VTP messages are received and translated into Telnet protocol messages.
- LAT/VT gateway — LAT messages are received and translated into VTP messages.
- VT/LAT gateway — VTP messages are received and translated into LAT protocol messages.

With the VT gateways, systems that do not have a VT implementation can access systems that do.

2.7.1.1 LAT/VT and VT/LAT Gateways

Without logging in to an account on the local system, you can use the LAT/VT gateway or the VT/LAT gateway to connect to a remote OSI system that has a Virtual Terminal responder installed. The only requirement for this function is that at least one LAT/VT gateway or VT/LAT gateway is enabled on your LAN on a system that runs Digital's local area transport (LAT) protocol. This is true for any terminal server that supports LAT, for example:

- Any computer terminal
- Any personal computer running MS-DOS, OS/2, or Macintosh
- Any OpenVMS or VMS V5.5 (or later) system
- Any VMS V5.4 (or later) system, running the optional LAT software

You can also use the appropriate command from a remote OSI VT system to access a system using the LAT protocol.

2.7.1.2 VT/Telnet and Telnet/VT Gateways

If you have a VT/Telnet gateway on your network, you can use the telnet command from an Internet system to access a remote OSI system that has a Virtual Terminal responder installed. If you have a Telnet/VT gateway, you can access the OSI system using the `set host/vtp gateway alias name` command line.

2.8 Management Tools and Utilities

Network management is provided with the Network Control Language (NCL). Network management implements the DECnet-Plus layered model, based on the Digital hierarchical structure called Enterprise Management Architecture (EMA).

The DECnet-Plus for OpenVMS network management software allows:

- System or network managers to control and monitor the operation of a network and provide information related to network traffic and performance.
- Network operating parameters to be configured.
- The manager to start up and shut down network components as needed once repaired.

In addition, network management software can provide information warning network managers of faulty or failing network components, both hardware and software.

2.8.1 Network Control Language (NCL)

NCL is the DECnet-Plus network management command-line interface. The structure and syntax of NCL reflects the DECnet-Plus internal structure of the network management components. NCL directives, or commands, let you manage DECnet-Plus entities by means of their unique network entity names.

NCL can also be used to test specific components of the network. NCL enables transmission and reception of test messages either between systems or through controller loopback arrangements. The messages can then be compared for possible errors. NCL aids in isolating network problems.

DECnet-Plus for OpenVMS Components

2.8 Management Tools and Utilities

2.8.1.1 Network Management Graphical User Interface (NET\$MGMT)

You can access NCL through either a command-line interface or a graphical user interface (GUI). The GUI allows you to view the status of network components and control those components from a Motif-based window interface. As such, it can be invoked using the same methods you use to invoke any other Motif application. You can run this application locally by issuing the following command:

```
$ run sys$system:net$mgmt
```

The application will check for and load the Helvetica 12-point 75-pitch font. In the unlikely event that this font is not present, the application will exit with an error message.

Once you have started NET\$MGMT, you can refer to the on-line help pull down menus for more information. Also, refer to the *DECnet-Plus for OpenVMS Network Management* guide.

2.8.2 Network Control Program (NCP)

DECnet-Plus for OpenVMS allows the use of NCP for the remote management of DECnet Phase IV nodes.

To aid the transition from DECnet Phase IV to DECnet-Plus for OpenVMS, the NCP emulator tool provides a necessary interface for the installation and use of layered products that issue DECnet Phase IV NCP commands.

The NCP emulator tool is not intended for management of DECnet-Plus for OpenVMS. It may be used to manage remote DECnet Phase IV nodes with the TELL and SET EXECUTOR NODE commands.

Because some NCP commands do not have equivalent NCL commands, Digital cannot guarantee that all layered products can be installed and used. For information on support for specific layered products, contact your local Digital office.

The installation procedure places documentation for the NCP emulator tool in the file SYS\$UPDATE:NCP_EMULATOR.TXT.

During installation, the DECnet Phase IV version of NCP is renamed SYS\$COMMON:[SYSEXE]NCP_PHASEIV.EXE and the NCP emulator tool is placed in the file SYS\$COMMON:[SYSEXE]NCP.EXE.

2.8.3 Common Trace Facility (CTF)

CTF software assists in network problem solving. CTF lets you collect and display information about specific protocol exchanges between systems. This information is often useful when you attempt to solve the following problems:

- Suspected configuration problems
- Failures while establishing or using network links
- Network overload
- Interoperability problems
- Problems with name and address resolution

CTF is not supported by all DECnet-Plus software products. Refer to your Software Product Description (SPD) for further information.

2.8.4 CMISE API

DECnet-Plus for OpenVMS supports an ISO CMISE application programming interface (API) conforming to the Service Definitions in ISO 9595. The API allows for development of applications that can communicate with other management applications conforming to ISO 9595 on remote nodes in the network.

2.8.5 Network Management Support Tools

The `decnet_register` tool assists with setting up Local namespaces, DECdns distributed namespaces, and managing the node names and addressing information they contain. The `decnet_migrate` tool helps you perform network management tasks and learn NCL.

These tools perform the following functions:

- The `decnet_register` tool manages node names in the namespaces used within your network.
 - Registers node names, node synonyms, and addresses in your namespaces.
 - Adds addresses to a node registration.
 - Removes addresses from a node registration.
 - Modifies a node registration's synonym or addresses.
 - Displays a node registration and verifies its internal consistency.
 - Exports node registration information from a namespace to a text file.
 - Imports node registration information from a text file into a namespace.
 - Updates a node's registered address information with information it obtains from the node itself.
 - Renames a registered node in a namespace.
 - Deregisters a node from a namespace.

The `decnet_register manage` command invokes the `decnet_register_decdns.com` command procedure (located in `sys$manager:`) to create and manage the required DECdns namespace directories and to set their protections.

- The `decnet_migrate` tool is useful for network management support:
 - Converts NCP commands to NCL commands.
 - Creates and edits NCL files using a Language-Sensitive Editor (LSE).
 - Gathers and reports detailed information about the network's current configuration.
 - Sets up interphase routing link-creation commands for use by Phase V routers.

2.8.6 DECnet-Plus Event Dispatcher (EVD)

The DECnet-Plus for OpenVMS Event Dispatcher (EVD) software reports significant events that occur during network operation. An **event** is an occurrence of a normal or abnormal condition that an entity detects. As directed by you, DECnet-Plus for OpenVMS maintains records of specific or general categories of events. These records can help you track the status of network components.

DECnet-Plus for OpenVMS Components

2.8 Management Tools and Utilities

Many events are informational. They record changes that you or the DECnet-Plus for OpenVMS software makes to components of the local system, or to those remote system entities that the local system's Event Dispatcher is tracking. Other events report potential or current problems in the physical parameters of the network.

You can set up event dispatching on a particular system, between two systems, or across multiple, distributed systems. The event-dispatching component can report events that occur on any system that runs DECnet-Plus software, and it is also capable of logging the events of remote Phase IV nodes via the DECnet-Plus Event Dispatcher relay.

2.8.7 CMIP Management Listener (CML)

The CMIP Management Listener (CML) is the DECnet-Plus management module that implements DNA Common Management Information Protocol (DNA CMIP). CMIP defines the exchange of network management information.

CML provides access to CMIP. NCL accesses management directives defined for DECnet-Plus entities using CMIP.

Part II

Conceptual Information

Part II consists of two conceptual chapters that describe fundamental DECnet-Plus networking and X.25 networking concepts. This section includes the following chapters:

- Chapter 3 — DECnet-Plus for OpenVMS Concepts
- Chapter 4 — X.25 Networking Concepts

DECnet-Plus for OpenVMS Concepts

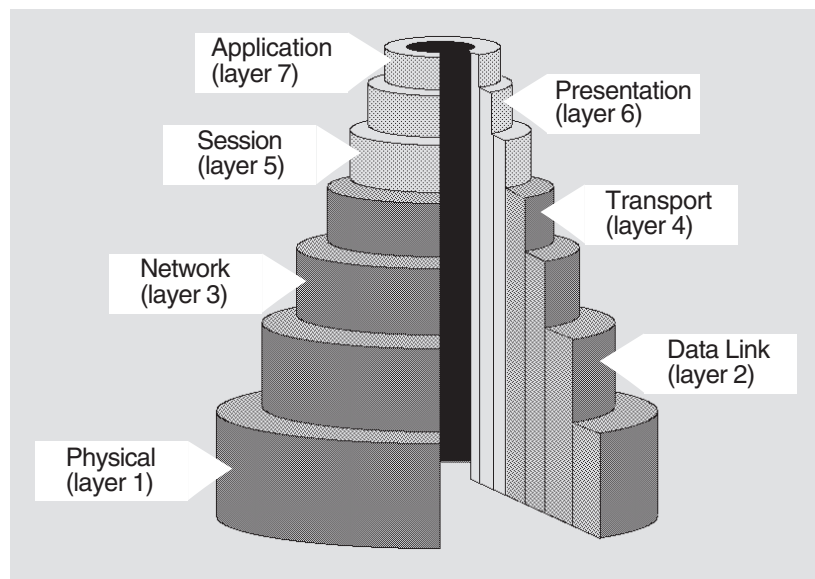
All Open Systems Interconnection (OSI) communications software implements global standards that are developed by the International Organization for Standardization (ISO). An OSI standard specifies a protocol or defines a service for one functional layer of the OSI Reference Model.

This chapter introduces OSI architecture, protocols, and standards, and how they apply to DECnet-Plus for OpenVMS.

3.1 Communications Architecture: The OSI Reference Model

The OSI Reference Model is a hierarchical architecture of seven layers for data exchange between systems with incompatible operating systems. The architecture provides standard protocols, services, and interfaces so systems that implement these standards can communicate. Figure 3–1 shows the OSI layers.

Figure 3–1 OSI Reference Model



LKG-6735-92R

DECnet-Plus for OpenVMS Concepts

3.1 Communications Architecture: The OSI Reference Model

Table 3–1 lists the functions of the OSI layers.

Table 3–1 Layers of the OSI Reference Model and Their Functions

Layer	Name	Responsibilities
Upper Layers		
7	Application	Provides for distributed processing and access; contains the application programs and supporting protocols (including File Transfer, Access, and Management (FTAM) and the Association Control Service Element (ACSE)) that use the lower layers; has no formal upper boundary, since it includes application programs and their individual user interfaces.
6	Presentation	Coordinates the conversion of data and data formats to meet the needs of the individual application processes.
5	Session	Organizes and structures the interactions between pairs of communicating application processes.
Lower Layers		
4	Transport	Provides reliable, transparent transfer of data between end systems, with error recovery and flow control.
3	Network	Permits communications between network entities in open systems on a subnetwork, intermediate systems, or both.
2	Data Link	Specifies the technique for moving data along network links between defined points on the network, and how to detect and correct errors in the Physical layer.
1	Physical	Connects systems to the physical communications media.

3.1.1 Comparison of DNA Phases

DNA Phase V integrates the lowest four architectural layers of OSI and DNA, while also offering the upper three layers of both DNA and OSI. Table 3–2 shows the names of the DECnet-Plus layers with the corresponding DNA Phase IV names.

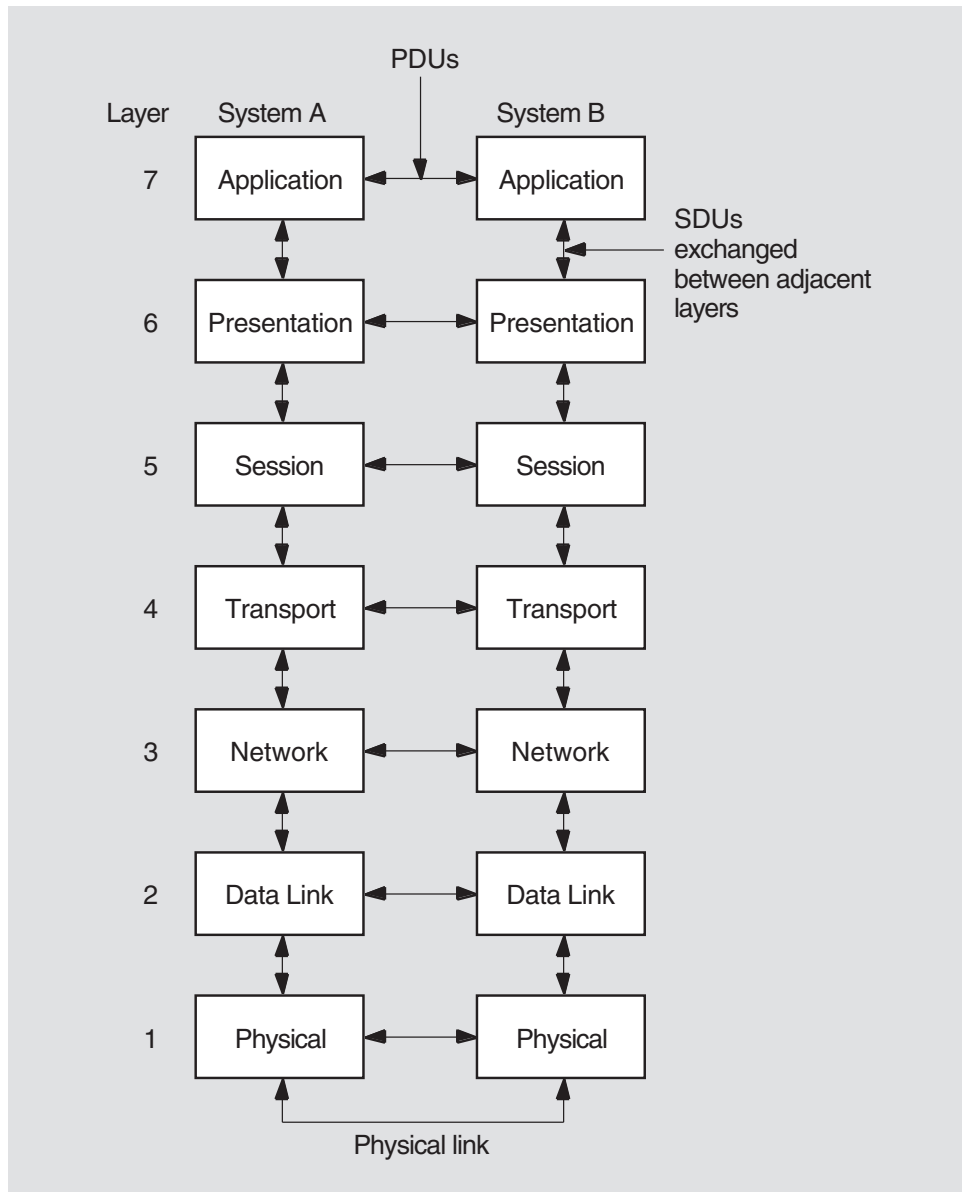
Table 3–2 Layer Names: DNA Phase IV and DNA Phase V

DNA Phase IV	DNA Phase V
DECnet and User Application layer	DECnet, OSI and User Application layer
Session Control layer	Presentation, Session layers
End Communications layer	Transport layer
Routing layer	Network layer
Data Link layer	Data Link layer
Physical Link layer	Physical layer

3.2 Data Flow

Except for the Application layer, each layer provides a service to the layer immediately above (its **user**). Every layer is supported by all the lower layers, and adds functional value to the service available from the layer immediately below. For example, the Transport layer adds in-sequence delivery of data and also retransmission of data lost by the lower layers. Figure 3–2 shows how data flows during communications.

Figure 3–2 OSI Reference Model: Data Flow



LKG-6496-92R

Entities within each layer on one system use the services provided by its next lower layer to communicate with other (peer) entities on a second system.

3.3 OSI Terminology

The **Reference Model** is a network design of layers that work together. Each **layer** is an independent and self-contained group of interconnected functions with its own purpose and protocols. Layers also provide services. Each layer uses the services provided by the layer beneath it.

For each layer, OSI has two types of international standard:

- **Protocol specification** — defines the protocol rules and formats.
- **Service definition** — describes the capabilities of the protocol or the service it provides.

3.3.1 Protocols

A **protocol** is a set of rules and procedures. Protocols govern the behavior of open systems at each layer. Some layers (for example, the Application layer) have several protocols.

A **protocol machine** is software that implements a particular protocol.

A **protocol data unit** (PDU) is information made up of protocol control information (PCI) from the current layer and, possibly, user data from the layer above. PDUs are exchanged between peer protocol machines using service primitives. A given service primitive can correspond to zero, one, or more PDUs. The integrity and identity of a PDU remains intact while it is being exchanged.

The underlying layer places each PDU received from the above layer into a **service data unit** (SDU). The SDU becomes user data in the PDU of its underlying layer. PDUs remain intact until they arrive at the protocol machine of the peer entity. The peer's protocol machine separates the PCI from user data and processes the PCI. Figure 3–3 shows the relationships among the PCI, user data, SDUs, and PDUs.

Figure 3-3 The Relationship of PDUs, User Data, and SDUs



LKG-4439-93R

3.3.2 Protocol Stacks (Towers)

Applications can communicate with each other only if they agree on the protocols they both will use and the operational parameters of these protocols. Also, exact address information is needed to indicate to each layer of protocol where to deliver data. This required information is encoded in a **protocol stack** or **protocol tower**, which is a data structure for encoding address and protocol information about a service user (application). The tower is a protocol sequence (an ordered list of protocol identifiers for each layer from the Network layer upward) with associated address and protocol-specific information.

3.3.3 Dialogues

OSI protocol machines ensure that open systems can establish message exchanges, called **dialogues**. Though many dialogues can occur simultaneously, each dialogue is independent of other dialogues. Protocol machines cooperate to conduct a dialogue through a predictable sequence of states. The portion of a dialogue conducted at the Application layer is called an **association**; the portions of dialogues conducted at other layers are called **connections**.

Associations or connections are established between service users by their peer service providers; for example, the Association Control Service Element (ACSE) establishes FTAM associations. Connections use the name of the service provider that establishes them; for example, a connection established by Presentation for ACSE is called a presentation connection.

DECnet-Plus for OpenVMS Concepts

3.3 OSI Terminology

3.3.4 Entities

Protocol machines operate within system-specific implementations, or processes. For every dialogue, one or more processes activate the protocol machines of each layer independently. An instance of such protocol-machine activity is called an entity. OSI **entities**, which are the manageable components that make up the network, relate to entities on other systems for example, a routing circuit. The relationship between processes and entities is implementation-specific.

Each entity communicates with equivalent entities on different systems, called **peer entities**.

3.3.5 Services

A **service** is an interface provided by a service element or a layer for accessing one or more functions. An **application service element** (ASE) is a component that provides an application-level function. FTAM and ACSE are examples of service elements.

The service element or layer that provides a service is called a **service provider**. An application program, service element, or layer that uses a service is called the **service user**.

3.3.5.1 Service Primitives

To request or receive indications of a service, peer entities exchange messages called **service primitives**. A service primitive carries parameter values for a specific service. There are two pairs of service primitives:

- Request and indication primitives — exist for all services
- Response and confirm primitives — exist for confirmed services

Each pair of service primitives generates only one unit of user data, a protocol data unit (PDU).

Service users and service providers originate primitives as follows:

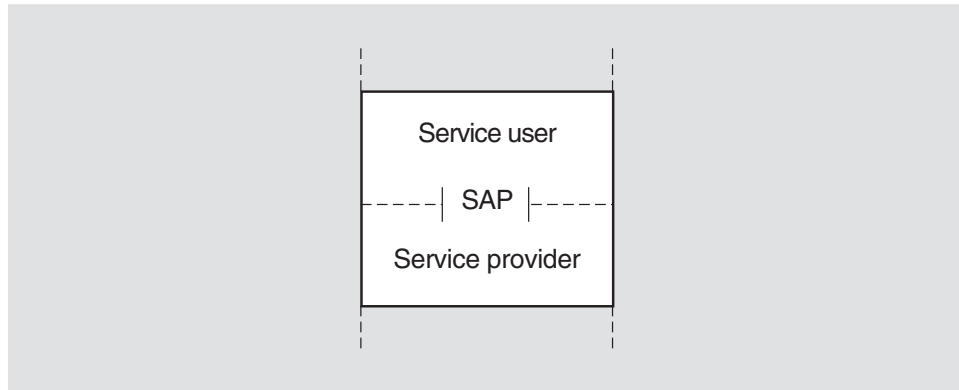
- Request and response primitives begin with service users, which use those primitives to request services and negotiate parameter values.
- Indication and confirm primitives begin with service providers, which use those primitives to pass on information that comes in for their users.

3.3.5.2 Service Access Points (SAPs)

A service provider delivers services to a service user at an interface called a **service access point** (SAP). A SAP is the point at which an entity provides a service to a user entity in the layer above.

Figure 3–4 shows how a service user, a service provider, and a SAP interact.

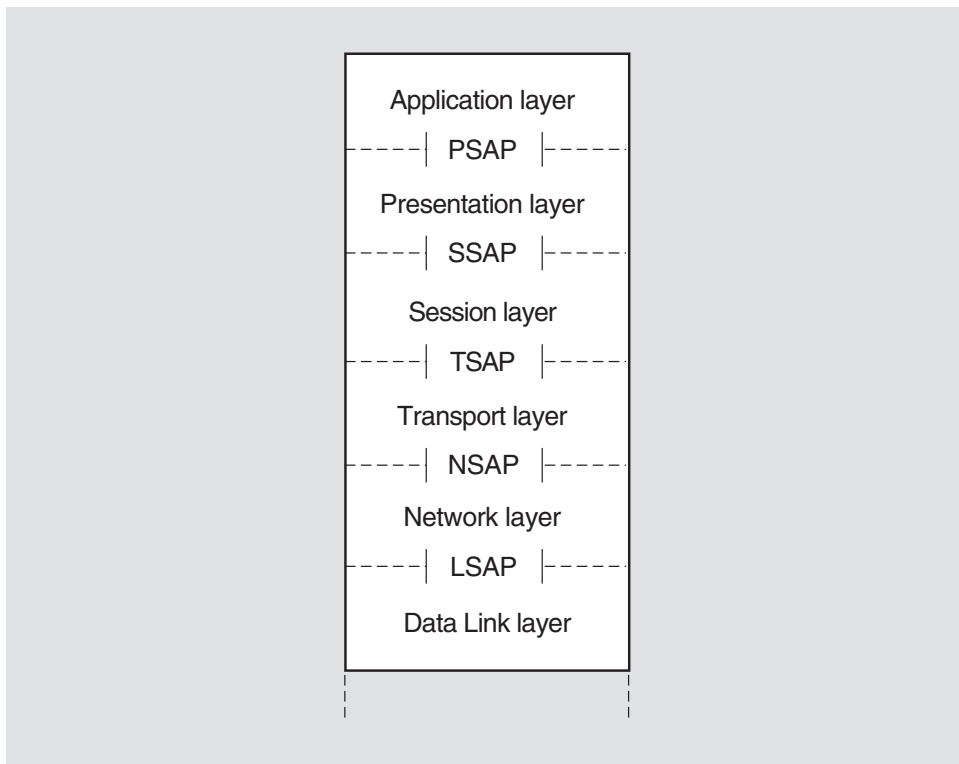
Figure 3-4 Service User, Service Provider, and SAP Interaction



LKG-4436-93R

Except for the Application and Physical layers, SAPs exist at the boundaries of all OSI layers. SAPs are named according to the layer providing the services. For example, transport services are provided at a Transport SAP (TSAP) at the top of the Transport layer and any SAP that the Presentation layer provides is called a Presentation SAP (PSAP). Figure 3-5 illustrates SAP nomenclature.

Figure 3-5 SAPs at Different OSI Layers



LKG-4437-93R

SAPs are logical constructs defined by an identifier called a **SAP selector**. To create a SAP, a system manager or application user defines a unique SAP selector for a specific layer. SAP selectors are the building blocks of addresses.

DECnet-Plus for OpenVMS Concepts

3.3 OSI Terminology

A given address takes the name of the uppermost SAP that contributes a selector to the address. For example, a presentation address accesses the Application layer.

A **presentation address** (p-address) contains one or more SAP selectors at the upper layers. The composition of a given address depends on the purpose of the address. For example, an address that allows the Transport layer to access the Application layer has SAPs at three layers: Transport, Session, and Presentation. A p-address is used for upper-layer addressing and has SAPs at four layers: Network, Transport, Session, and Presentation.

In addition, on a LAN, the Data Link layer uses the source service access point (SSAP) and the destination service access point (DSAP):

- SSAP — 1-byte field in an LLC frame that identifies the **link service access point** (LSAP) of the sending client protocol. (The LSAP is the ISO 8802-2 (LLC) protocol identifier that identifies a Network layer entity.)
- DSAP — 1-byte field in a LLC frame that identifies the LSAP of the receiving client protocol.

3.4 The Upper Three OSI Layers

Each of the upper three layers provides services for user and application processes.

The **Session layer** service is used by a pair of communicating application processes to structure and control their dialogue, conducted over the transport connection.

The **Presentation layer** works in conjunction with the Session layer to provide a specialized data transfer service to the Application layer. It coordinates any necessary conversion of data between a pair of communicating application processes.

Because the semantics and syntax used by one application are unlikely to be the same as those used by another on a different system, the Presentation layer transfers data in a system-independent way. The data must be in a form that both applications can recognize; for example, data with floating-point numbers can pass between two systems even if each system has a different internal representation of floating-point numbers.

The **Application layer** contains many different entities, including the parts of an application program that interact with the communications environment on a system. The Application layer has a mixture of standard protocols. Some directly support application processes, others support those that give direct support to the application processes, and some do both, such as the ACSE protocol.

3.4.1 Presentation and Session Entities

Presentation and Session entities provide the Presentation and Session connections. To form an association, an OSI application entity requires a corresponding Presentation entity and Session entity. For the FTAM software, for example, these Presentation and Session entities occur within the same process as the FTAM entity.

The combined effect of the Presentation and Session entities is to manage data transfer for Application entities. For OSI applications:

- Presentation handles syntax transformation and also establishes Presentation contexts and manages them, if necessary.
- Presentation and Session both perform connection management functions for their own connections.
- Session also transfers information and manages each dialogue between peer Application entities.
- Presentation supports this use of Session services by providing the Application layer with a set of services that correspond to the data-transfer and dialogue-management services of the Session layer.

3.5 DNA Session Control Layer

The Session Control layer provides the DNA user and application interfaces. Session Control performs address resolution and selection, connection control, and manages transport connections on behalf of applications. It also supports the applications environment, including \$IPC and \$QIO programming interfaces.

3.5.1 Objects and Applications

The terms "object" and "application" are used differently in DECnet-Plus than in Phase IV:

- A **Phase IV object** (also referred to as a **Phase IV application**) is a Digital or user-written component that is identified by a number and associated name (for example, object number 27 for mail). This information is stored in an object database at each Phase IV node.
- A DECnet-Plus **application** is equivalent to a Phase IV object. DECnet-Plus applications are identified by program name and can be specified in the application database at the system.
- A DECnet-Plus **object** or **object entry** is a resource name stored in a namespace. The resource can be a system, a service made available by an application, or some other resource.

3.5.2 Protocol Towers

The sequence of protocol identifiers in a protocol tower typically extends from the Network layer up to the DNA application. Session Control builds and maintains multiple towers for the local node object: one tower for each protocol and each address through which Session Control can access the object. A **protocol tower set** includes all the protocol towers that the system builds for the object.

When a DNA application on one node requests a connection to another node, Session Control requests that DECdns provides `DNA$Towers` attribute information for the remote node. Session Control compares the tower information with the tower information it maintains for the local node and selects a protocol tower common to both nodes. The connection is then made automatically without a user or application needing to know what lower-layer protocols the nodes are using.

DECnet-Plus for OpenVMS Concepts

3.5 DNA Session Control Layer

3.5.3 Address Resolution

The Session Control layer maps the global name of the node object with a set of protocols and addresses that can be used to communicate with that node. The Session Control layer performs the following functions for address resolution:

- Maintains, in the namespace, the following information in the `DNA$Towers` attribute for local objects:
 - Protocols supported on the local system
 - DECnet-Plus network address of the local system
- Obtains information about the remote object from the namespace to determine the sequences of protocols and address information required to communicate with the remote node or application.
- Selects all common towers and passes this information to the address selection function.

You can autoconfigure addresses for DECnet-Plus systems, which can change over time. Session Control creates and maintains the protocol and address information in the towers in the namespace for the node object entry.

Session Control supports a `RegisterObject` function to maintain the address attribute of an object entry if the underlying protocol tower address element of the node changes. Applications can request that Session Control perform this function for their objects.

3.5.4 Address Selection

The Session Control layer initiates a connection based on the address of a system and the Session Control application being connected to that system. It receives a list of common protocol towers with the address-resolution function and looks for a set of mutually supported protocols. It tries to make a connection using all towers in common; the connection fails if all towers are tried and none works. The selected protocols and address information are used for communication with the remote system.

3.5.5 Connection Control

The Session Control layer performs connection services for Session Control applications. It requests, maintains, and disconnects or aborts transport connections. Session Control can request a connection using the destination address. It can receive a connect request, validate the access control information supplied with the request, and identify the remote application making the request. The Session Control layer sends and receives data over transport connections.

The Session Control layer forms the interface between all DNA applications and the Transport layer. It allows use of the NSP Transport and OSI Transport protocols.

3.5.6 Digital-Supplied Session Control Layer Applications

Session Control applications in DECnet-Plus for OpenVMS provide general-purpose network services. A Session Control layer application is identified by application type, which is a discrete numeric identifier for either a user task or a DECnet-Plus program, such as file access listener (FAL). DECnet-Plus for OpenVMS uses application type numbers to enable logical link communications using the NSP Transport Service or the OSI Transport Service.

DECnet-Plus has two types of Session Control applications:

- Applications with a 0 application type
These applications are usually user-defined images for special-purpose applications. They are named when a user requests a connection.
- Nonzero applications
Nonzero applications are known applications that provide specific network services such as FAL. They can also be user-defined tasks; these applications should be for user-supplied known services. Application type numbers for all nonzero applications range from 1 to 255. The number serves as a standard addressing mechanism across a heterogeneous network.

3.6 Transport Layer

The **Transport layer** ensures the reliable transfer of data. Each Transport protocol provides for the establishment of a transport connection, which carries a stream of two-way communications traffic between two processes on the same or different systems. A transport connection is a temporary logical connection that normally exists until one of the processes terminates the connection.

The Transport layer offers an interface to the service user at the Session layer. DNA applications can use either the \$IPC or \$QIO programming interface to make a connection request through the proprietary Session Control layer to the Transport layer. OSI applications can make connection requests through ACSE, the Presentation layer, and the Session layer to the Transport layer.

3.6.1 Transport Layer Ports and Session Layer Ports

DECnet-Plus uses a mechanism called **logical link** for communication between processes. A Phase IV logical link in DECnet-Plus terms is either a **Transport layer port** or a **Session layer port**.

A logical link is a temporary connection either between processes on source and destination systems or between two processes on the same system. A logical link carries a stream of regular data and interrupt data of full-duplex traffic between two user-level processes. Each logical link is a temporary data path that exists until one of the two processes terminates the connection.

3.6.2 Supported Transport Protocols

The OSI Transport Protocol, the implementation of which provides the OSI Transport Service, bridges the gap between the service provided by the network and the service desired by the transport user.

The Transport layer supports both the Digital NSP and the OSI Transport Protocols. An OSI application makes a connection request to the Transport layer to access the OSI transport. The choice of protocol for a non-OSI connection is made during connection establishment: either the connect request specifies a Transport protocol, or the Session layer selects an appropriate Transport Protocol. Multiple transport connections, using any of the Transport Protocols, can be made simultaneously.

DECnet-Plus for OpenVMS Concepts

3.6 Transport Layer

3.6.2.1 OSI Transport Protocol

The OSI Transport Protocol permits communication between DECnet-Plus systems and other vendors' systems that also implement the OSI Transport Protocol.

The OSI Transport Protocol conforms to the ISO 8072 Service Definition and the ISO 8073 Protocol Standard. They define **OSI Transport Protocol classes** 0, 2 and 4 (TP 0, TP 2, and TP 4).

This protocol can use two types of ISO network service:

- Connection-Oriented Network Service (CONS)
- Connectionless-Mode Network Service (CLNS)

The OSI transport conforms to the RFC 1006 Standard and to the RFC 1006 extension Internet Draft. They define how to implement ISO 8073 Transport Class 0 on top of TCP (RFC 1006) and how to implement ISO 8073 Transport Class 2, non-use of Explicit Flow Control on top of TCP (RFC 1006 extension). The network service used is provided by TCP.

Table 3–3 describes these classes, their functions, and which network service can be used.

Table 3–3 Functions of the OSI Transport Protocol Classes

Protocol Class	Functions	Network Service
TP 0	Provides a basic transport service.	CONS and RFC 1006
TP 2	Provides all functions of TP 0 plus multiplexing of more than one transport connection over a network connection or TCP connection. Also provides flow control over CONS.	CONS and RFC 1006 extension
TP 4	Provides all functions of TP 2 plus error detection and recovery.	CONS and CLNS

Some other differences are that:

- TP 0 relies on the upper layers to do its error correction. This class is disconnected if the underlying Network layer is disconnected.
- TP 2 and 4 use disconnect requests.
- TP 4 reassigns the OSI transport connection to another Network layer connection if the existing one fails.

Digital recommends TP 4 for both DNA and OSI applications because it has the greatest set of capabilities suitable for all purposes, unless when using RFC 1006 or RFC 1006 extension.

When a transport user sets up a transport connection, a preferred protocol class for the connection is specified in the connection request. The responding transport user must either agree to this protocol class, or suggest an alternative protocol class that is acceptable to the initiating user. If no such agreement is possible, the transport connection cannot be set up.

3.6.2.2 Network Services Protocol (NSP)

The proprietary NSP provides a transport service that supports error detection and recovery and uses CLNS. NSP capabilities are similar to those of OSI Transport Protocol class 4 for CLNS connections.

NSP makes a transport connection between two DECnet-Plus systems or between a DECnet-Plus system and a Phase IV system. NSP is the only transport that can connect to Phase IV systems.

NSP establishes transport connections by exchanging control messages with a peer NSP module. The connection comprises two data subchannels, one for normal data exchange and one for other data (such as expedited data messages and flow control messages). NSP transfers data in segments, if necessary. The segments are then reassembled in correct sequence at the destination.

3.6.3 OSI Transport Service

The Transport layer provides the OSI transport service, which fulfills communications requirements for clients in the higher layers. The OSI transport service software implements the OSI Transport Protocol, allowing two OSI transport users to set up and use an OSI transport connection. A **transport user** uses the OSI transport service to set up a **transport connection** with another transport user.

The OSI transport service is a consistent interface to transport users, regardless of the type of network over which they make OSI transport connections.

DECnet-Plus for OpenVMS offers three classes of OSI transport service, each of which is appropriate to a particular network service because each provides the additional functions not provided by that service. TP 0, 2, and 4 identify the different classes of OSI transport service offered in DECnet-Plus for OpenVMS (see Table 3–3).

3.6.3.1 OSI Transport Service Functions

Transport users employ the OSI transport service to set up connections and exchange data. Depending on the type of network service used by an OSI transport connection, the OSI transport service offers all or only some of the following functions:

- Mapping OSI transport connections to transport users
When making an OSI transport connection, the transport user provides a unique reference for each of the two communicating transport users. This reference is called a **transport service access point (TSAP)**. Messages sent over an OSI transport connection include the TSAP of the transport user to which they are addressed. The OSI transport service uses this TSAP information to route messages to their proper destination.
- Multiplexing OSI transport connections
The OSI transport service uses a single network connection to send and receive message traffic for several OSI transport connections. This process is called **multiplexing**.
- Splitting and recombining data
The OSI transport service software cannot split an OSI transport connection among several network connections. However, the software does recombine incoming data from multiple Network connections for a particular transport connection.

DECnet-Plus for OpenVMS Concepts

3.6 Transport Layer

- Concatenating Transport layer protocol data units

Messages sent over an OSI transport connection are in the form of **Transport layer protocol data units** (TPDUs). To reduce the traffic between the Transport and Network layers, the OSI transport service software concatenates certain TPDUs to make one network service data unit (NSDU). The software does this with one AK (acknowledge) TPDUs and one data TPDUs from the same transport connection. The software never concatenates other types of TPDUs, nor TPDUs from different transport connections. The OSI transport service software processes incoming NSDUs if the combination of TPDUs is within the Transport Protocol standard.

- Controlling flow

The transport users at either end of an OSI transport connection might operate at different speeds. The OSI transport service software uses flow control to stem the output from a local user when the user at the other end cannot process the data at the required rate to keep up. Without flow control, one user could send messages over the connection faster than the other user could receive and process them, leading to the receiving user running out of buffers and discarding the messages.

The OSI transport service software in DECnet-Plus for OpenVMS always uses flow control with class 2 connections when operating over CONS.

- Providing error detection and recovery

In addition to setting up the OSI transport connection for data transfer, the OSI transport service ensures that the message exchange occurs without corruption or loss of data, and that the messages are delivered in sequence. The OSI transport service keeps a record of all the messages sent and received and their sequence, and, if they have not been received by the other end, retransmits them.

The software calculates the retransmission timers for each network. An X.25 network, for example, is very likely to need a longer retransmission timer than a LAN.

3.6.3.2 OSI Transport Templates

When a transport user makes a connection request, one of the parameters of this request is the name of a **transport template**, which is a predefined set of parameters for setting up the OSI transport connection. An OSI transport template specifies the type of network service to be used for the OSI transport connection, and the preferred protocol class.

You can use an OSI transport template to provide defaults for any parameters not included in the connection request.

3.6.3.3 OSI Transport Connections

An **OSI transport connection** is an end-to-end connection. It is a reliable two-way, data-transfer path between two OSI transport users. An OSI transport connection has three phases:

- Setting up the connection — an OSI transport user (the **initiating user**) on one end system (the **initiating host**) sends a connection request TPDUs to another OSI transport user (the **responding user**) on a second end system (the **responding host**). When a successful connection is made, data transfer can take place in either direction.

- Using the connection through which to transfer data — OSI transport connections support two kinds of data transfer:
 - **Normal data** transfer for usual message exchange or
 - **Expedited data** transfer which bypasses any blockage due to the flow control applied to normal data; only for sending small amounts of data; has its own type of TPDU and transmission rules.
- Releasing the connection — either transport user can release the OSI transport connection by sending a disconnect TPDU.

You can set up OSI transport connections:

- Between two systems on the same ISO 8802-3 LAN.
- Between two systems that are connected, either directly or via an X.25 connection.
- Between two systems that are connected directly by an X.25 point-to-point link.
- Between two systems on different subnetworks, where the linking subnetworks might mix technologies.
- Between two systems that are connected via TCP/IP.

3.7 Network Layer

The Network layer permits communications between network entities in open systems on a subnetwork, intermediate systems, or both. The Network layer provides conformance to ISO standards for packet formats and network addressing:

- ISO 8473 — Internetwork Protocol (Inactive Network Layer Protocol)
- ISO 8208 — X.25 Packet-Level Protocol (PLP)
- ISO 8878 — X.25 to provide CONS
- ISO 9542 — ES-IS Routing Exchange Protocol
- ISO 10589 — IS-IS Routing Protocol

3.7.1 Network Services

The Network layer is responsible for connecting subnetworks to form a network, and supports connections to the following:

- Broadcast subnetworks, such as ISO 8802-3 LANs
- Nonbroadcast subnetworks of permanent point-to-point links, such as leased or private links
- Nonbroadcast subnetworks
- Dynamically assigned X.25 data links

An OSI transport connection between two OSI transport users is supported by a network connection between the end systems on which the OSI transport users are running. This network connection is set up and maintained by the Network layers of the two end systems. The Network layer, therefore, provides a **network service** to the OSI transport service.

DECnet-Plus for OpenVMS Concepts

3.7 Network Layer

The Network layer offers two types of network service:

- **Connectionless-Mode Network Service (CLNS)**
- **Connection-Oriented Network Service (CONS)**

CLNS and CONS have both similarities and differences:

- Similarities: Peer entities communicate according to an agreed protocol through the services provided by the lower layer.
- Differences:
 - In connection-oriented transmission, resources have to be reserved to support data exchange on the connection that are not needed in connectionless transmission.
 - In connectionless transmission, each access to a data service needs additional addressing information. This overhead is avoided when the data exchange takes place using a connection-oriented service.
 - The connectionless service offers greater flexibility for implementing routing techniques.

Service-type conversion takes place only in the Transport and Network layers. This means, for example, that a presentation connection must be supported by a session connection, but that a change to using the connectionless network service can be done through the functions of the Transport layer.

The type of network service used depends on the topology of the network between the two end systems. The transport user selects the type of network service for a transport connection by supplying in the connection request a transport template that specifies the desired service.

3.7.1.1 Connectionless-Mode Network Service (CLNS)

Connectionless-Mode Network Service is provided to OSI transport and operates according to a datagram model. Each message is routed and delivered to its destination independently of others. For example, the DNA Network (Routing) layer provides this type of service. The Network layer allows a CLNS connection using an X.25 virtual circuit in one of two ways:

- X.25 static circuits
An X.25 virtual circuit is permanently assigned to the circuit. Network management enables the circuit and establishes and clears the calls. The connection is maintained until the circuit is turned off.
- X.25 dynamically assigned circuits
An X.25 virtual circuit is created only when data needs to be transferred. These circuits are similar to Phase IV data link mapping (DLM) circuits.

CLNS offers the following features:

- To exchange data, users do not first set up and subsequently release a connection. Data exchange is achieved with each access to a service.
- Each data transfer is entirely self-contained. All the information needed to deliver the data is given when the service is invoked.

With CLNS, the Network layer supports three types of connections:

- Broadcast circuits

Circuits on which multiple systems are connected. A message can be transmitted to multiple receivers and all systems are adjacent, for example, ISO 8802-3 (CSMA-CD) LANs.

- Point-to-point (nonbroadcast) circuits
Circuits that connect only two systems. A point-to-point configuration requires a separate physical connection between each pair of systems that communicate directly with each other. Examples are HDLC, DDCMP, and X.25 static circuits.
- X.25 dynamically assigned (DA) circuits
Circuits used to exchange Internet (ISO 8473) PDUs over a PSDN to another system, either DECnet-Plus or multivendor. The ES-IS Protocol (ISO 9542) does not run over an X.25 DA circuit.

There are two forms of the CLNS Network service:

- CLNS with Internet/ES-IS (end-system-to-intermediate-system routing)
Any transport connection can use CLNS with Internet/ES-IS. The communicating end systems can be on the same or different subnetworks. This network service is provided by the implementation of the ES-IS Internet routing protocols, which route packets from the end system to an intermediate system on the same subnetwork. The intermediate system ensures that packets reach their final destination. Two end systems that implement ES-IS on the same subnetwork can communicate without an intervening intermediate system.
- CLNS with the Inactive Network Layer Protocol
A transport connection can use CLNS with the Inactive Network Layer Protocol (ISO 8473) when the two end systems are on the same LAN. This network service is provided by the inactive subset of the Internet Protocol. No intermediate system is involved in the network connection.

Both forms of CLNS support only TP 4 connections.

3.7.1.2 Connection-Oriented Network Service (CONS)

Connection-Oriented Network Service is provided to OSI transport and operates according to a connection-oriented model. A connection is set up between two communicating end users, is used for data exchange, and is then broken by either end. SDUs sent over the connection do not have to contain a destination address. For example, X.25 provides this type of service.

CONS offers the following features:

- The users, supported by their service providers, set up a connection by negotiating the context for data transfer.
The connection has a definite lifetime because after it is set up and data exchange has taken place, it is released by the users (or the service providers).
- The connection exists specifically between two users, allowing them to make multiple data transfers without specifying the destination on each transfer.

A transport connection can use CONS when the underlying network connection is an X.25 connection. An X.25 connection can be:

- Between two systems attached to a PSDN, either directly or via an X.25 gateway.

DECnet-Plus for OpenVMS Concepts

3.7 Network Layer

- A point-to-point connection to a DTE using the Link Access Protocol Balanced (LAPB) as the data link protocol.
- A direct connection between two systems on the same IEEE 802.3 LAN, using the logical link control (LLC) type 2 (LLC2) protocol.

Note that in each of these cases, the two end systems are on the same subnetwork.

3.7.2 Broadcast Network Connections: The Routing Process

With CLNS, the Network layer receives user data from the Transport layer and determines the path along which the data travels to its destination. This decision process is the main function of **routing**. The Network layer provides end-to-end data transfer, or routing, as a service to clients in the Transport layer. This data transfer between the two communicating systems is called a **network connection**.

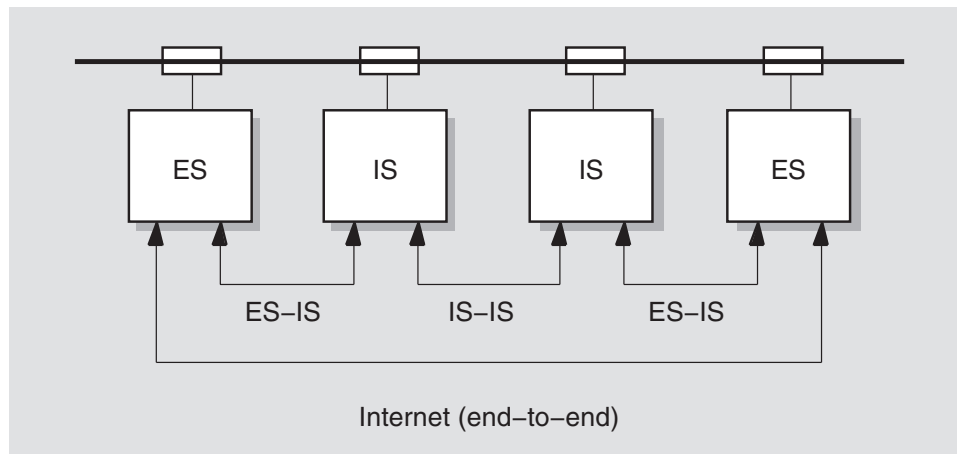
An **end system (ES)** is either the source or destination of data sent over a network connection. An end system receives data units, called packets, addressed to it and sends data units to other systems on the same subnetwork. (DECnet Phase IV calls this type of system an "end node" and the "subnetwork" an area.)

An **intermediate system (IS)** is a routing system that receives data packets from a source end system, or from the previous intermediate system on the route, and passes them on to the destination end system, or to the next intermediate system on the route. (DECnet Phase IV calls this type of system a "router.")

An intermediate system interconnects two subnetworks. A **subnetwork** is a network, within a group of interconnected networks, of OSI systems that all use a common addressing format. A subnetwork forms an autonomous whole, for example: ISO 8802-3 (CSMA-CD) LAN, HDLC data link, or X.25 connections.

In the Network layer, the **ES-IS** protocol (ISO 9542) provides the communication between an end system and the nearest intermediate system. ES-IS is a connectionless protocol. **IS-IS** protocol (ISO 10589) provides the communication between two intermediate systems. The Internet standard provides the end-to-end connectionless link between end systems. Figure 3-6 shows how ES-IS relates to other ISO protocols.

Figure 3-6 ES-IS and IS-IS Protocols



LKG-6499-92R

The Phase IV routing components "circuit" and "line" are used differently in DECnet-Plus. These DECnet-Plus "entities" are called, respectively, **routing circuits** and **routing ports**:

- Routing circuit — connection between the Network layer and the Data Link layer below. Routing circuits operate over physical lines. A routing circuit is a logical (virtual) link.
- Routing port — connection between the Network layer and the Transport layer above.

For additional information on routing, see the *DECnet-Plus Planning Guide*.

3.7.2.1 Compliance with OSI Packet Formats and Addressing

An OSI **administrative domain** consists of a combination of end systems, intermediate systems, and subnetworks. You can also subdivide the administrative domain into **routing domains**, in which all systems and subnetworks use the same routing protocols.

The Network layer complies with the ISO standards for network packet formats and addressing. Data messages are forwarded through the network in self-contained packets that include the source and destination addresses.

DECnet-Plus for OpenVMS systems support the packet format that is specified in ISO standard 8473, and can exchange data with other vendors' systems that also conform to the ISO packet format standard.

The Network layer accepts messages from the Transport layer and encloses them in packets called network protocol data units (NPDUs). The NPDU includes the Network layer header that contains the source and destination addresses for the data. The Network layer address for a system is called the network service access point (NSAP). The NSAP is located at the boundary between the Network layer and the Transport layer, where communication between the layers takes place. For complete information about NSAPs, see the *DECnet-Plus Planning Guide*.

The Network layer uses the destination NSAP address to forward the NPDUs to the destination system. The Network layer also provides compatibility with Phase IV.

DECnet-Plus for OpenVMS Concepts

3.8 Data Link Layer

3.8 Data Link Layer

The **Data Link layer** provides a communications path between directly connected systems in a network. It controls the movement of information between systems, including the transmission and receipt of data. In providing delivery of data to the adjacent node, the Data Link layer performs some or all of the following functions: establishment of the link (initializing and conditioning the line), error detection and recovery, data flow control, data framing control, and packet sequence control.

DECnet-Plus for OpenVMS uses synchronous, asynchronous (VAX only), Ethernet, and FDDI communications controllers to interface with other network nodes.

LAN connectivity is provided by the CSMA-CD and FDDI controllers and drivers supporting ISO 8802-2 logical link control (LLC) type 1 connectionless service and ISO 8802-3 LLC type 2. DECnet-Plus also supports Ethernet V2 packets on CSMA-CD devices.

Use of FDDI packets larger than 1500 bytes requires a Phase V router on the FDDI LAN. As with cluster alias support, the Phase V router may be configured to run the Phase IV distance vector routing protocol or the Phase V Link State Routing Protocol.

WAN connectivity is provided by WAN device drivers supporting host-based synchronous communications options for wide area networking.

All the data link devices support DDCMP, HDLC/LAPB and SDLC protocols. BISYNC and GENBYTE (VAX only) protocols are also supported on some options. WAN device drivers are required by X.25 to establish host-based wide area connections.

Synchronous controllers use DDCMP or HDLC, either when directly connected or when connected via modems, to provide full- or half-duplex communications over point-to-point lines. Synchronous DDCMP multipoint tributary connections are also supported. Asynchronous controllers (on VAX systems) use DDCMP, either when directly connected or when connected via modems, to provide only full-duplex communications over point-to-point lines. Error correcting and data suppression modems are not supported.

Asynchronous lines (on VAX systems) are supported only to other systems running DECnet-Plus for OpenVMS VAX, DECnet-VAX, DECnet-RSX, and DECnet-DOS.

DDCMP operation (on VAX systems) is not supported in cases where an asynchronous physical communications line is emulated by lower-level protocols or communications subsystems. Examples of this include X.29 virtual terminals, asynchronous connections as emulated by terminal servers, and connections via data switches.

DECnet-Plus for OpenVMS allows up to four circuits to be defined and operational on an end system. This capability allows a single end system to be connected to up to four separate LANs or WANs. Note that all circuits must be equal in capacity and connectivity.

3.8.1 Support for CSMA-CD Protocol on a LAN

The Data Link layer allows the transmission of data over a local area network cable by means of the CSMA-CD (Carrier Sense Multiple Access with Collision Detection) protocol. CSMA-CD ensures equal access to multiple systems connected to the LAN. DECnet-Plus for OpenVMS supports both the existing Ethernet protocol and the protocol that complies with ISO 8802 (also known as IEEE 802).

The Ethernet and ISO 8802 protocols are compatible, with only slight differences in packet format. A DECnet-Plus system transmits in both ISO and Ethernet formats, and listens for frames in ISO and Ethernet formats. If a Phase IV system is connected to the LAN, the DECnet-Plus system also transmits in Ethernet format.

CSMA-CD LANs are similar to LANs defined by ISO 8802-3. LANs are privately owned, reliable, high-speed networks that connect information-processing systems in a limited geographic area, such as an office, a building, or a complex of buildings. It is a best-effort delivery system.

CSMA-CD allows multiple stations to access the broadcast channel at will, avoids contention by means of carrier sense and deference, and resolves contention by means of collision detection and retransmission. Each station awaits an idle channel before transmitting and can detect overlapping transmissions by other stations.

CSMA-CD stations provide for multiaccess connection between a number of systems on the same broadcast circuit. This kind of routing circuit is a path to many systems. Each system on one CSMA-CD station is considered adjacent to every other system on that station and equally accessible.

The media is passive coaxial cable or shielded twisted-pair cable that uses Manchester-encoded, digital baseband signaling, with interconnections containing all active components so that no switching logic or central computer is needed to establish or control communications.

At the Data Link layer, network control for the LAN is multiaccess, fairly distributed to all systems. The frame length allocation is from 64 to 1518 bytes (including an 18-byte envelope).

A system on an ISO 8802-3 (CSMA-CD) LAN is connected to the CSMA-CD station by:

- A LAN communications controller
- A transceiver
- A transceiver cable

When manufactured, LAN controllers are given a 48-bit hardware address.

A particular ISO 8802-3 system is identified by the hardware address of its station (line); this hardware address is stored in read-only memory in the LAN controller. When DECnet-Plus starts a CSMA-CD station, it constructs a physical address for the system. When you shut off machine power or change the state of the CSMA-CD station to off, the LAN controller resets the physical address to the original hardware address.

DECnet-Plus for OpenVMS Concepts

3.8 Data Link Layer

DECnet-Plus has no restrictions on the number of end systems on a LAN. In addition, you do not need an intermediate system on a CSMA-CD LAN.

1. When no intermediate system is on the LAN:
 - An end system trying to communicate with another end system sends the data by using a specific multicast address.
 - Only a system whose link service access point (LSAP) matches the destination DSAP in the multicast packet will respond. The destination system responds with an end system hello, sent directly to the originating system, informing the source system of its address. (For definitions of LSAP and DSAP, see Section 3.3.5.2.)
 - The source system and the destination system speak directly because each knows the LAN address of the other.
2. When one or more intermediate systems is on the LAN:
 - The first message from the source is sent to a randomly selected intermediate system.
 - The intermediate system detects that both the source and destination systems are on the same LAN.
 - The intermediate system forwards that first message to the destination and also sends a redirect message to the source system.

In this way, end systems learn the LAN addresses of other end systems so they can speak directly.

ISO 8802-3 (CSMA-CD) LANs support a bus topology, a single communications medium to which all the systems are attached as equals. The single network cable replaces the numerous interconnecting cables usually required in traditional WANs. This type of network is also called a **broadcast LAN**. The maximum possible distance between systems on the LAN is 2.8 kilometers (1.74 miles).

3.8.1.1 Extended LANs

You can connect segments of coaxial cable to extend LANs beyond the 500-meter (1640 feet) limit of a single segment. Extended LANs create larger networks in terms of distance and also in terms of the number of connections that can be made. (A 500-meter segment supports 100 physical connections.) Repeaters and bridges join cable segments:

- A repeater simply extends the LAN by retransmitting all network traffic that originates on one segment onto the attached segment, enabling connected segments to function as one cable.
- A bridge, in addition to extending the LAN, filters network traffic between segments.

If data packets are addressed to a destination system on the attached segment, the bridge transmits the packet to the segment. If data packets are addressed to a local destination system, the packets are confined to the segment to which the source system is attached. The primary benefit of bridges is to reduce network traffic.

3.8.1.2 Intermediate System

If the end systems communicate only with each other, you do not need an intermediate system on a LAN, but can use host-based routing. Optionally, you can use one to limit multicast traffic. To route messages off the LAN over other routing circuits, you must configure an intermediate system such as DDCMP circuits.

If a LAN is operating with more than one area and with one or more level 1 intermediate systems, you need a level 2 intermediate system to transport messages between areas.

3.8.1.3 Multicircuit End Systems

DECnet-Plus for OpenVMS allows multiple circuits to be active and usable simultaneously on an end system. For example, you can connect an end system to two LAN cables. Both routing circuits are used and traffic is split between the circuits, but no routing occurs over these circuits. A DECnet-Plus for OpenVMS end system can support a maximum of three circuits. This feature provides for redundancy and increased data throughput without requiring an intermediate system.

3.8.1.4 Areas and Multihomed Systems

You can partition a large DECnet-Plus routing domain into subdomains called **areas**. For Phase IV, an area is a group of network nodes that can run independently, with all nodes in the group having the same area address. DECnet-Plus areas are similar to Phase IV areas except for the following new features:

- Multihomed systems, which have more than one area address
- Single-area LANs

A node can be in only one area in a network. DECnet-Plus systems, however, can have more than one area address. Such a system is a **multihomed** system. You can assign up to three area addresses to a DECnet-Plus system.

A DECnet-Plus area is a set of systems that all share the same area address (or addresses). An area (and the systems within the area) can have more than one area address. For example, if an area in a DECnet-Plus network is connected to an X.25 public network, a system in that area of the network would have two addresses: one for the DECnet-Plus network and one for the X.25 network. A system cannot have addresses on two networks that are not connected.

If you connect two areas to a DECnet-Plus LAN, the level 2 intermediate systems automatically combine themselves into one area with two area addresses. Phase IV LANs can be divided into several areas. Mixed Phase IV and DECnet-Plus LANs can also include several areas.

3.8.2 Support for the HDLC Protocol

High-level Data Link Control (HDLC) protocol data links are ISO, synchronous, point-to-point links that are basically the same in function as existing DDCMP synchronous links. However, HDLC is a bit-oriented protocol, whereas DDCMP is a byte-oriented protocol.

HDLC operates over synchronous, switched, or nonswitched communications links. HDLC supports a broad range of existing subsets, including the subset used in X.25 networks.

DECnet-Plus for OpenVMS Concepts

3.8 Data Link Layer

HDLC operates in either of two modes:

- Balanced mode — Operational mode used over full-duplex links
- Normal mode — Operational mode used over half-duplex links

HDLC links use UI frames and XID frames. A UI frame is an unnumbered, information frame that carries data not subject to flow control or error recovery. An XID frame exchanges operational parameters between participating stations.

3.8.2.1 LAPB Support

DECnet-Plus for OpenVMS also supports a modified form of HDLC called link access protocol balanced (LAPB). LAPB is the CCITT-approved link level protocol for X.25 connections. LAPB defines the procedure for link control in which the DTE/DCE interface is defined as operating in two-way asynchronous balanced mode (ABM). LAPB is for the reliable transfer of a packet from a host to an X.25 packet switch, which then forwards the packet on to its destination.

3.8.3 Support for the DDCMP Protocol

DDCMP is designed to provide an error-free communications path between adjacent systems. It operates over serial lines, delimits frames by a special character, and includes checksums at the link level.

DECnet-Plus for OpenVMS continues to support proprietary DDCMP data links, which include these types of connections:

- Synchronous point-to-point
- Synchronous tributary multipoint
- Asynchronous point-to-point
- Asynchronous static (permanent)
- Asynchronous dynamic (switched temporary)

DDCMP provides a low-level communications path between systems. The protocol detects any bit errors that are introduced by the communications channel and requests retransmission of the block. The DDCMP module provides for framing, link management, and message exchange (data transfer). Framing involves synchronization of bytes and messages.

DDCMP pipelining permits several packets to be sent before an acknowledgment is received. Piggybacking permits an acknowledgment to be transmitted on a data packet.

The DDCMP protocol moves information blocks over an unreliable communication channel and guarantees delivery of routing messages. Individual systems on DDCMP routing circuits are addressed directly because no multicast or broadcast addressing capability is available.

The Data Link layer supports point-to-point, DDCMP links, either synchronous or asynchronous. The two types of asynchronous links are **static** (permanent) and **dynamic** (switched temporary).

3.8.3.1 Synchronous DDCMP

Synchronous links provide the medium to high-speed point-to-point communication. The synchronous DDCMP Protocol can run in full- or half-duplex mode. This allows DDCMP the flexibility of being used for local synchronous communications or for remote synchronous communications over a telephone line using a modem.

DDCMP is implemented in the driver software (WANDD) for the synchronous communications port.

3.8.3.2 Asynchronous DDCMP

Asynchronous links provide a low-speed, low-cost media for point-to-point communication. Asynchronous DDCMP can run over any directly connected station that the DECnet-Plus for OpenVMS system supports. Asynchronous DDCMP provides for a full-duplex connection. You can use it for remote asynchronous communications over a telephone line using a modem. Asynchronous connections are not supported for maintenance operations or for controller loopback testing.

Asynchronous DDCMP does not need to be predefined for dynamic connections. It is established automatically when a dynamic asynchronous DDCMP link is made.

DDCMP asynchronous links can be static or dynamic.

- **Static connection** — The asynchronous station is permanently configured as a communications link.

A static asynchronous DDCMP connection is a permanent DECnet-Plus connection between two systems physically connected by stations. You convert them to static asynchronous DDCMP stations by issuing NCL commands to set the stations to support the DDCMP protocol. The user at each system then turns the appropriate routing circuits and stations on for DECnet-Plus use. After the communications link is established, it remains available until a user turns off the routing circuit and station and clears the entries from the appropriate NCL script file.

Static asynchronous DDCMP configurations require the asynchronous DDCMP driver to be connected. The asynchronous DDCMP protocol can run in full-duplex operation on local asynchronous communication links.

You can configure a dial-up line as either a static or dynamic asynchronous station, but the dynamic connection may be more secure and convenient to use.

- **Dynamic connection** — A station connected to a terminal port is switched to an asynchronous communications station for the duration of a call.

A dynamic asynchronous connection is a temporary connection between two systems, generally over a telephone line using modems. The stations at both ends of the connection can be switched to asynchronous DDCMP communications stations and then switched back to terminal lines.

You can use dynamic asynchronous connections to establish a DECnet-Plus link to another system for a limited time or to create links to different systems at different times.

DECnet-Plus for OpenVMS Concepts

3.8 Data Link Layer

3.8.3.3 Converting DDCMP Links to HDLC Links

DECnet-Plus supports HDLC as an alternative to DDCMP to:

- Align DECnet-Plus more closely with international standards
- Support new industry-standard communications controller chips with higher-level capability

You can convert existing DDCMP point-to-point synchronous lines to HDLC lines. However, DECnet-Plus also supports DDCMP for compatibility and to provide these capabilities not available with HDLC:

- Continuing support for existing controllers that cannot use HDLC
- Use over asynchronous lines

3.9 Physical Layer

The **Physical layer** is responsible for the transmission and receipt of data on the physical media that connects systems. It transparently moves data between the system and the communications path signaling equipment. The Physical layer can include part of the device driver for a communications device and for communications hardware: interface devices, modems, and communication lines.

3.9.1 CSMA-CD LAN Interface

The Physical layer supports the CSMA-CD interface, which complies with ISO standard 8802-3 and the IEEE 802.3 standard. The Physical layer also allows CSMA-CD LAN connections based on the DECnet Phase IV Ethernet interface.

3.9.2 Modem Connect Module

The DECnet-Plus Modem Connect module defines the operation of synchronous and asynchronous devices. It provides for network management of stations (physical lines) that conform to industry standards for modem connection. You can establish and monitor the following types of links:

- Synchronous and asynchronous connections
- Point-to-point lines
- Tributary multipoint lines
- Leased lines

The Modem Connect module supports several industry standards for physical interfaces:

- Electronic Industries Association (EIA) RS-232-C, RS-422, and RS-423
- CCITT V.24
- CCITT V.25 and V.25bis for auto call and auto answer
- CCITT V.35

The Modem Connect module does not contain driver-specific code. It contains all the common routines related to network management for all synchronous and asynchronous drivers. The functions controlled through the Modem Connect module include:

- Data transfer services
- Network management of the port and line entities

- Line profile identification which defines any nonstandard operation of the line

3.10 Network Management

DECnet-Plus for OpenVMS network management capabilities include:

- The director-entity model.
- A command line management interface, the Network Control Language (NCL), which replaces the Phase IV Network Control Program (NCP).
- Remote management of Phase IV nodes using the NCP Emulator.
- DNA Common Management Information Protocol (DNA CMIP).
- DECnet-Plus for OpenVMS initialization scripts support.
- Maintenance operations: downline load, upline dump, remote console connection, and loopback testing support.
- An enhanced event logging (EVD).
- Common Trace Facility (CTF) for troubleshooting.
- Network management graphical user interface (NET\$MGMT), an enhanced Motif-based windows interface.

3.10.1 Network Management Components

The structure of DECnet-Plus network management defines formal relationships between the management software at the various layers and the directors that communicate with the layers on behalf of network managers.

The two major components of network management are directors and entities. **Directors** are interfaces for managing the entities. They are typically used by the network manager and usually involve a command language. NCL management **entities**, which are the manageable components that make up the network, relate to other entities on the same system.

The entity model defines the structure of the entities that constitute a distributed system and the management functions they provide.

- **node entity** — Top-level entity in the entity hierarchy; identified by a globally unique **node name**, which represents an individual computer system.
- **Module** — The next level of entity on a node; a group of network functions that provide a service. Each module includes the entities that make up the module. An example of a module is a protocol, for example HDLC, and the entities used to manage that protocol. Each entity defines the management functions, characteristics, status, and attributes that are pertinent for its operation.

There is only one occurrence, or instance, of a module entity in a node. For this reason, modules can be uniquely identified within a node by their class name.

- **Entities** — The next level down in the hierarchy; defined with NCL commands to allow management of some part of a module's functions. The LAPB module, for example, maintains `lapb link` entities for each communications link over which the protocol operates; `lapb link` is the full class name of the entity. Each instance of the `lapb link` entity requires further identification to allow it to be distinguished from the others.

DECnet-Plus for OpenVMS Concepts

3.10 Network Management

For further information on NCL commands, refer to the *DECnet-Plus Network Control Language Reference* guide.

3.10.2 How the Director Works

The Network Control Language (NCL) is a command line interface to the director, which interprets the input NCL commands, sends these commands to a target entity for processing, and presents the results back to the network manager.

Directors use common mechanisms to communicate with the entities they manage. The same director can manage both local and remote entities. If the managed entity resides on the local system, the director uses the local interface to access it. If the managed entity resides on a remote system, the director converts the NCL message to CMIP and sends the converted message to the entity agent on the remote system.

The management information and operations that pass between directors and entities are:

- Directives — The management commands issued by a director to an entity. They allow a director to read and alter an entity's management attributes, or to request it to perform a specific action.
- Attributes — A piece of management information maintained by an entity. Each attribute has a name allowing it to be accessed by management. The four types of attributes are:
 - Identification: identifies an entity to network management.
 - Characteristics: allows control of the operating parameters of an entity. In general, characteristic attributes take default values when the entity is created, but you can change them with NCL commands.
 - Status: allows you to inspect the current state of an entity. Unlike characteristic attributes, status attributes can change without management intervention.
 - Counters: values that indicate the number of times an operation has been performed by an entity or a particular condition has been detected.
- Events — an occurrence of a specific normal or abnormal condition.

3.10.2.1 Management Access Relationship

The **management access relationship** between a parent entity and a child entity indicates that the parent is capable of passing directives to its child entities. A **parent entity** is an entity that has created another entity (the child entity); a **child entity** is a lower class of entity that receives directives forwarded from its parent entity.

The DECnet-Plus director does not directly access the agents of local entities. To access a local entity's management interface, the director accesses the appropriate global node entity that is the parent of the target local entity. That node entity forwards the directive down the entity hierarchy; the forwarding process continues from higher to lower entities until the correct target entity receives the directive. The **agent access point** is the place of connection between a director or a parent entity and an agent.

3.10.2.2 Management of Remote DECnet Phase IV Nodes

The DECnet-Plus network management protocol is based on DNA **Common Management Information Protocol** (DNA CMIP) draft standard for network management operations. CMIP is used for encoding network management operations that can be performed on an entity. CMIP permits the exchange of information between a director and an agent. CMIP supersedes the Phase IV Network Information and Control Exchange (NICE) protocol.

DECnet-Plus software supports remote management of Phase IV nodes by continuing to support Phase IV NCP and the Phase IV network management protocol (NICE).

However, Phase IV applications that have been written to create logical links to the Phase IV Network Management Listener (NML) and then parse the returned NICE protocol messages are not supported for managing DECnet-Plus for OpenVMS systems. To run on a network composed of DECnet-Plus systems, those applications must be rewritten to use DECnet-Plus network management software and protocols. For example, the application may be rewritten to use the DECnet-Plus CML callable interface into network management. Also, Phase IV applications that use NCP to configure the application need to be converted to use NCL.

3.10.2.3 Support for Phase IV NCP and the NICE Protocol

DECnet-Plus software supports remote management of DECnet Phase IV nodes by use of NCP.EXE. This utility supports a significant range of NCP commands. It is not designed as a replacement for NCL.

3.10.2.4 Maintenance Operation Protocol (MOP) Support

With the Maintenance Operation Protocol (MOP), you can communicate with systems that are not fully operational, for example DECnet-Plus intermediate systems. Low-level maintenance functions of MOP run directly on top of data links, such as ISO 8802-2. MOP functions are often placed in ROMs on link controllers. MOP operations include:

- Downline load and upline dump
- Connection to remote consoles
- Loopback testing

DECnet-Plus for OpenVMS provides enhanced support for concurrent downline loads.

3.10.3 Configuration Procedure for DECnet-Plus for OpenVMS

The `net$configure.com` configuration procedure offers you several options for configuring DECnet-Plus:

- The FAST configuration option allows you to configure DECnet-Plus quickly, with a minimal number of questions. This new option is especially useful when you want your DECnet-Plus (Phase V) system to operate in the same way as a DECnet Phase IV system. Use this option when you are not completely familiar with the DECnet-Plus product and are upgrading a DECnet Phase IV node to DECnet-Plus. With the fast configuration option, you can quickly and easily configure a DECnet-Plus system with full network connectivity. The DECnet-Plus system configures itself by determining the Phase IV and OpenVMS operating system parameters. A local database holds naming information. You can reconfigure the system at a later time to include additional functionality.

DECnet-Plus for OpenVMS Concepts

3.10 Network Management

Currently, the fast configuration option is not supported on a node that is running in an OpenVMS cluster or on a node that is a DECdns server.

You invoke the fast configuration feature by specifying the following command:

```
$ @sys$manager:net$configure
```

- The Basic configuration option (*DECnet-Plus for OpenVMS Installation and Basic Configuration*) allows you to configure your system for DECnet-Plus by answering a few questions and using the default answers on others. For example, use the Basic configuration option if you want to:
 - Configure one communications device, or multiple communications devices used for DECnet-Plus communications
 - Configure default names for all devices and routing circuits
 - Autoconfigure your network addresses You invoke the Basic configuration option by specifying the following command:

```
$ @sys$manager:net$configure basic
```

- The Advanced configuration option (*DECnet-Plus for OpenVMS Applications Installation and Advanced Configuration*) allows you to customize your system's network configuration, such as to:
 - Use DECnet over TCP/IP or OSI over TCP/IP
 - Support multiple communications devices with a mix of protocols
 - Add more flexibility in how you run X.25 services
 - Specify your own names for certain network components rather than accepting the defaults
 - Configure DECnet-Plus for OpenVMS Virtual Terminal, OSAK, and/or FTAM software
 - Configure your system as a DECdns server
 - Configure your own network addresses (NETs) rather than have them autoconfigured

You invoke the Advanced configuration option by specifying the following command:

```
$ @sys$manager:net$configure advanced
```

You can also use `net$configure.com` to reconfigure all or some of the DECnet-Plus for OpenVMS system. Rerunning the configuration procedure modifies or replaces relevant initialization script files but does not affect running systems. You then execute the modified initialization script files to effect these changes.

The initialization scripts create and enable all required entities. Each entity is initialized through execution of a separate NCL script file. Using NCL scripts to initialize DECnet-Plus for OpenVMS systems replaces the Phase IV requirement of establishing a DECnet permanent configuration database at each node. Remote node information resides in either a local or distributed DECdns namespace.

For further information, refer to the *DECnet-Plus Network Management* guide.

X.25 Networking Concepts

4.1 Introduction

This chapter introduces basic X.25 networking concepts and how they fit into the DECnet standards for system interconnection as illustrated in Figure 2–2. Refer to the *X.25 for OpenVMS Management Guide* for specific information on managing and monitoring an X.25 system as well as descriptions of specific parts of an X.25 system (call handling, templates, application filters, server and relay clients, and addressing).

X.25 is a recommendation made by the Comité Consultatif International Téléphonique et Télégraphique (CCITT). The CCITT is a United Nations committee that makes recommendations on data communications services. The CCITT has made several recommendations to ensure that different networks are compatible in their operation. Many of its recommendations have been implemented widely by major network providers.

The X.25 recommendation specifies the interface between data terminal equipment (DTE) and data circuit-terminating equipment (DCE) for equipment operating in the packet mode on public data networks.

4.1.1 Packet Switching Data Networks (PSDN)

Packet switching is a method of sending data across a network. In this method, data is divided into blocks, called **packets**, which are then sent across a network. Networks using this method are referred to as packet switching data networks (**PSDNs**) or X.25 networks.

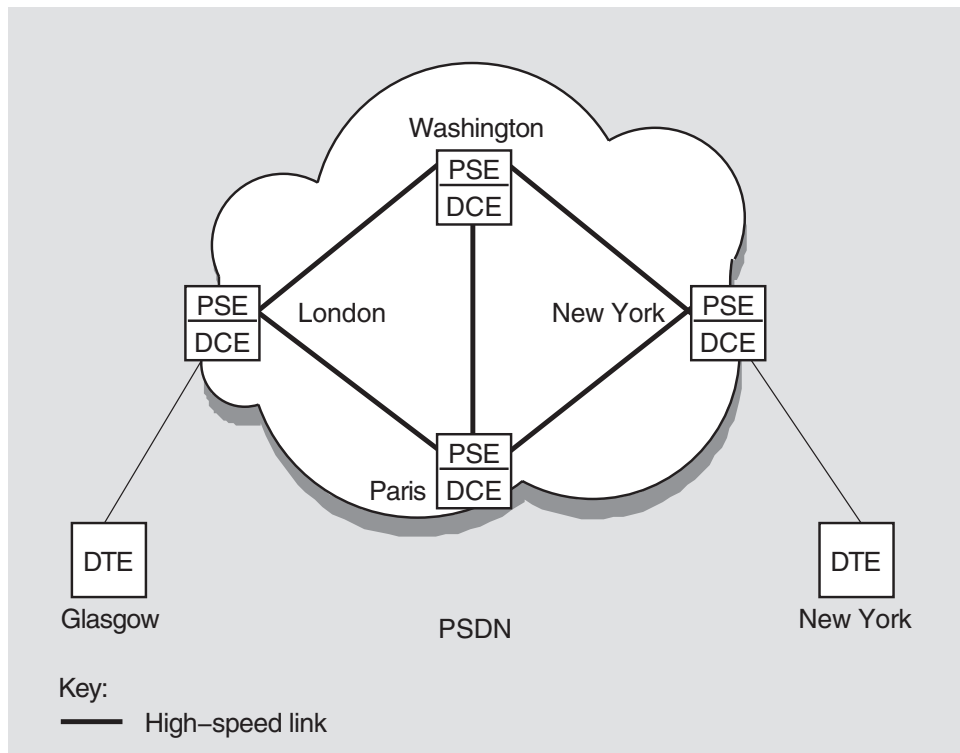
Figure 4–1 shows an example of the equipment involved in sending data across a PSDN. PSDNs are made up of packet switching exchanges (**PSEs**), and high-speed links that connect the PSEs. Each PSE contains data circuit-terminating equipment (**DCE**). The DCE is the point where all data enters and leaves a PSDN.

The user's computer that sends data to the DCE, and receives data from the DCE, is known as the data terminal equipment (**DTE**). Outside a PSDN, packets travel between the DTE and DCE. Inside a PSDN, packets travel between PSEs.

X.25 Networking Concepts

4.1 Introduction

Figure 4–1 Packet Switching Equipment



LKG-6508-92R

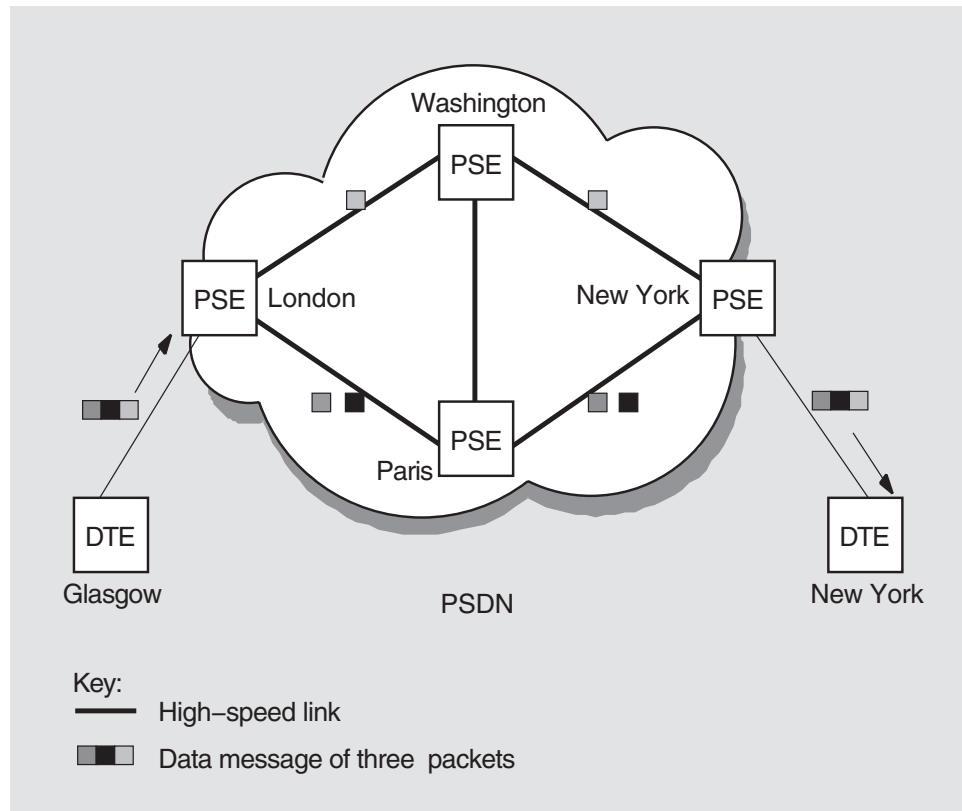
A PSDN is either owned by a country's Postal, Telegraph, and Telephone (PTT), Authority or it is privately run. Public PSDNs can be used by anyone, on payment of connection fees and regular bills. Private PSDNs are used to serve a single body, such as a large corporation.

Public and private PSDNs can be interconnected. It is, therefore, possible to pass data packets from one DTE to another no matter where in the world the DTEs are located.

The PSDN sends each packet to its destination by the best available route at the time. The packets that make up a data message may take different routes to reach the same destination; this will occur if certain routes are busy or if there is a line failure within the PSDN. Figure 4–2 shows an example of packets traveling by different routes across a PSDN.

When the packets reach their destination, they are reassembled so that the original order of the packets in the data message is maintained.

Figure 4-2 Packets Traveling Over a PSDN



LKG-6507-92R

There are two types of data terminal equipment:

- **Packet-mode DTEs**

These DTEs communicate directly with PSDNs using X.25 packets; for example, a computer system.

- **Character-mode DTEs**

These DTEs cannot assemble or disassemble packets and have no X.25 communication capability. An example of a character-mode DTE is an asynchronous terminal. Character-mode DTEs communicate with a PSDN through a device known as a packet assembler/disassembler (**PAD**). Character-mode DTEs are often referred to as X.29 terminals.

4.1.2 PADs and Character-Mode Terminals

PADs allow character-mode DTEs to access PSDNs. A PAD performs the following functions:

- Assembles a terminal's outgoing characters into packets for transmission across a PSDN, and disassembling incoming packets (that is, packets arriving from a PSDN) into individual characters.
- Controls the transmission and receipt of packets.
- Sets up virtual circuits.

X.25 Networking Concepts

4.1 Introduction

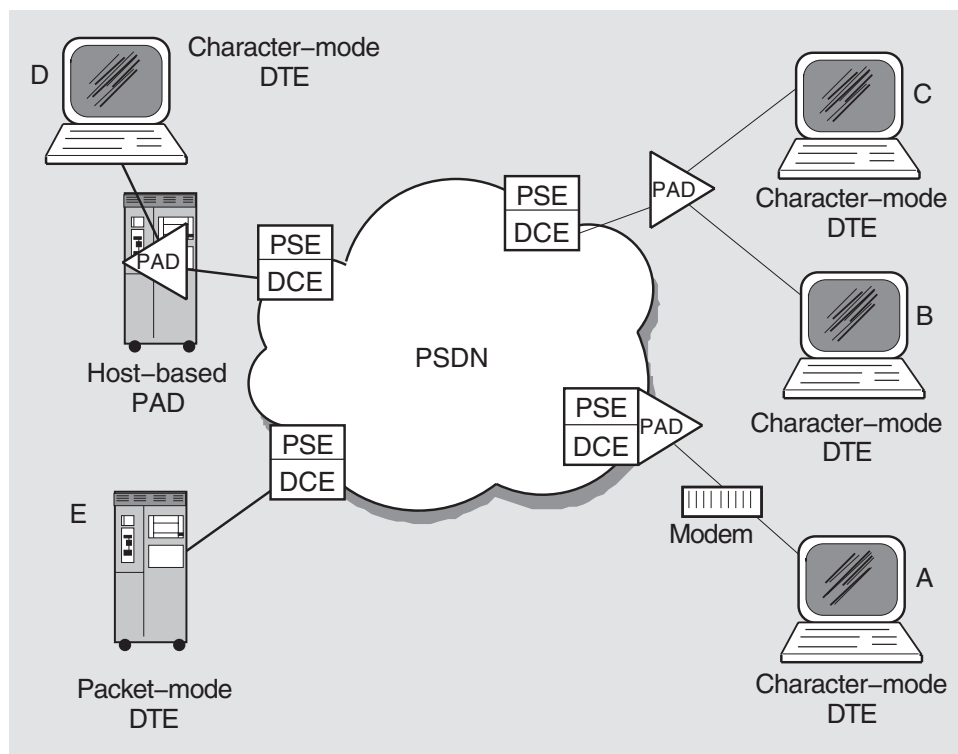
There are three types of PAD:

- Dial-in PADs provided by PSDNs for public use. This type of PAD allows users to dial in to a PSDN via a modem and from that PSDN access other DTEs. PAD functionality is provided by software in the PAD.
- PADs that allow the connection of character-mode terminals to a PSDN without the need to dial in via a modem. These PADs are typically one separate, dedicated piece of hardware to which both the character-mode terminals and the PSDN are connected. This type of PAD allows multiple character-mode terminals to be connected to the PSDN. PAD functionality is provided by software in the PAD.
- Host-based PADs, where the PAD functionality is implemented solely in the software installed on the host.

Figure 4-3 shows the following PADs:

- DTE A is connected to the PSDN via a modem.
- DTEs B and C are connected to the PSDN via a dedicated PAD.
- DTE D is connected to the PSDN through its own, host-based PAD.
- DTE E is a packet-mode DTE, which can connect directly to the PSDN without a PAD.

Figure 4-3 Connecting Character-Mode DTEs to a PSDN



LKG-6509-92R

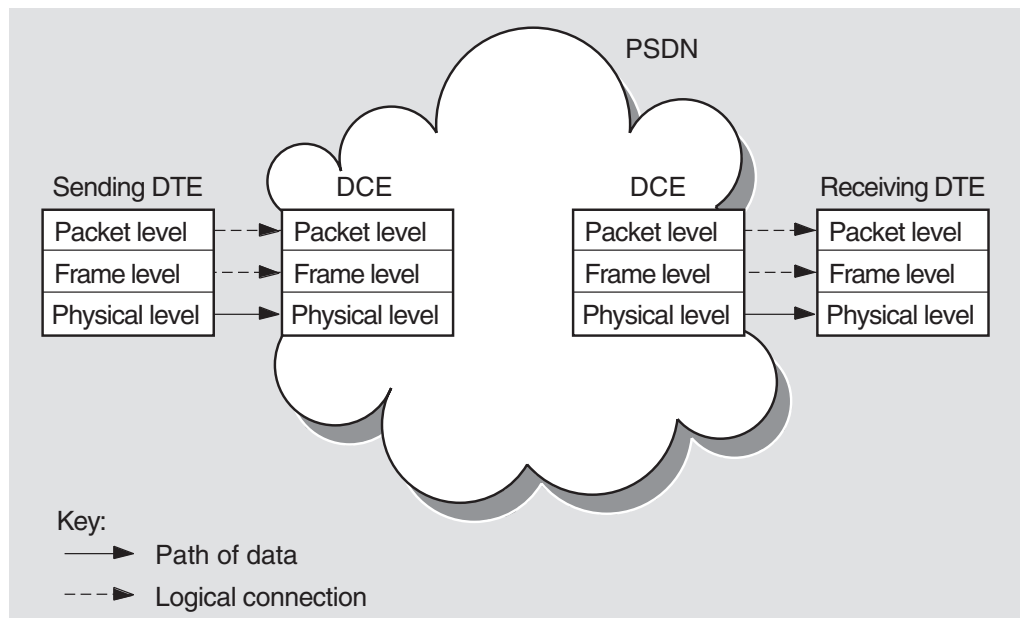
4.2 X.25 Protocols

A **protocol** is a set of rules for the formatting and sequencing of data. X.25 describes three protocol levels for the interface between a DTE and a DCE:

- Packet level
- Frame level
- Physical level

Figure 4–4 shows the three protocol levels.

Figure 4–4 X.25 Protocol Levels



LKG-6510-93R

At the sending DTE, data passes down through the levels and each level adds its own control information. When it reaches the physical level, the data is transmitted to the DCE.

At the DCE, the relevant control information is removed as the data passes up through the levels. When it reaches the packet level, data is in the form of X.25 packets, which can now be sent across a PSDN.

When the packets reach the remote DCE, they pass down through the levels and pick up control information. The DCE then sends them to the receiving DTE. At the receiving DTE, the data passes up through the levels.

Each level on the DTE communicates with the corresponding level on the DCE by using the same protocols; therefore, there are *logical connections* between corresponding levels. They are referred to as logical connections because data is not actually transmitted until it reaches the physical level.

Two DTEs implementing X.25 protocols can exchange data in packets, regardless of the DTE manufacturer. However, once the data is re-assembled in its original format, the receiving DTE may not be able to interpret the message. In this case, additional software may be required to make the data meaningful to the receiving DTE.

X.25 Networking Concepts

4.2 X.25 Protocols

4.2.1 Packet Level

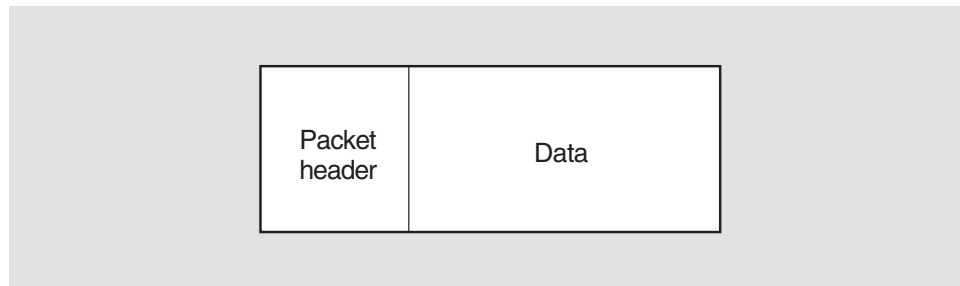
This level defines the procedures for formatting packets, sequencing packets, and establishing virtual circuits. Virtual circuits are the logical connections between two DTEs (see Section 4.4.2.1). Most packets contain user data, but some are for control purposes. Control packets are used for functions such as initiating and terminating virtual circuits between two DTEs and for interrupt messages.

The amount of user data allowed in a packet is known as the **packet size**. The maximum amount of user data allowed in a packet is set by the PSDN you use. A typical packet size is 128 bytes.

Each packet (refer to Figure 4–5) consists of a packet header followed by user data or control information.

- The packet header contains a logical channel number (**LCN**), and a packet type identifier. The LCN identifies the virtual circuit between the calling DTE and the called DTE (see Section 4.4.2). The packet type identifier labels the packet as containing either user data packet or control information.
- The remainder of each packet contains either user data or control information.

Figure 4–5 Structure of a Packet



LKG-8126-93R

4.2.2 Frame Level

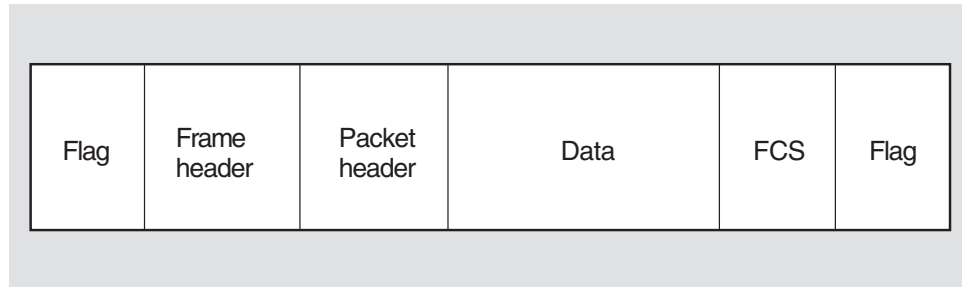
This level initiates communications between the DTE and the DCE and ensures that data is transferred between DTE and DCE without errors.

At this level each packet is enclosed within a frame. Each frame consists of:

- Flags that indicate the start and end of the frame.
- A frame header, which indicates whether the frame contains user data or control information.
- A packet from the packet level (provided that the frame does not contain control information).
- A 16-bit frame check sequence (FCS), which is used to determine whether the whole frame has been received without error after transmission.

Figure 4–6 shows the structure of a typical frame.

Figure 4–6 Structure of a Frame



LKG–6511–93R

4.2.3 Physical Level

This level defines the mechanical and electrical connections between a DTE and a DCE. It specifies electrical characteristics, data transmission speeds, and connector types.

4.3 Implemented Standards

X.25 is a CCITT recommendation that describes a standard for connecting to packet switching networks. The lower three layers of the OSI Reference Model correspond to the three levels in the Comité Consultatif International Téléphonique et Télégraphique (CCITT) X.25 recommendation.

The X.25 standard describes the protocols for the DTE-DCE interface.

4.3.1 Related CCITT Standards

The X.25 recommendation describes only those protocols involved in the DTE-DCE interface to packet switching networks. There are several other CCITT recommendations that are relevant to packet switching, including:

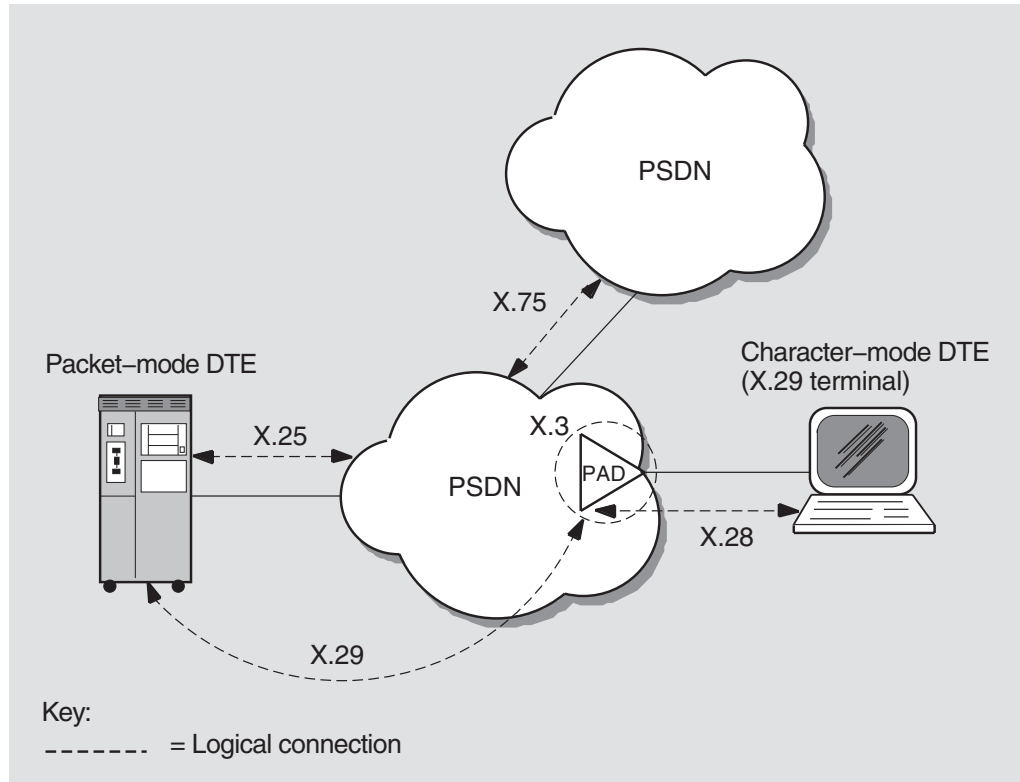
- X.3 — Defines the service provided to the terminal by the PAD. The service is described in terms of PAD parameters.
- X.28 — Defines the user interface to the PAD.
- X.29 — Defines the procedure for the exchange of control information and user data between a PAD and a remote packet-mode DTE using X.25.
- X.75 — Defines the protocols for communication between two PSDNs.
- X.121 — Defines the address encoding used in PSDNs.

X.25 Networking Concepts

4.3 Implemented Standards

Figure 4–7 illustrates the main packet switching recommendations.

Figure 4–7 CCITT Packet-Switching Recommendations



LKG-6512-92R

4.3.2 Related OSI Standards

DECnet-Plus for OpenVMS also implements the following PSDN-related standards from OSI:

- Connection-Oriented Network Service (CONS) — OSI Transport Classes 0, 2, and 4 over X.25 subnetworks
- Connectionless-mode Network Service (CLNS) — OSI Transport Class 4 over X.25 subnetworks
- Link Access Protocol Balanced (LAPB) to exchange frames between a DTE and a DCE
- LLC Type 2 (LLC2) for communications over a LAN

4.4 How Connections Are Made and Data Is Transferred

To establish a call and transfer data between DTEs via a PSDN, physical and logical connections must be made between the calling and called DTE.

4.4.1 Physical Connections

To transfer data, the DTE and DCE must be connected. This connection can take many forms but, in general, the connection is made using a physical line. Two types of line can be used to connect a DTE and a DCE:

- **Leased lines**

These are for the exclusive use of one DTE and are available only for communication between that DTE and the PSDN.

Leased lines are also known as dedicated or private lines.

Data transmission over leased lines is always synchronous and is usually full-duplex. Typical speeds of data transmission are in the range of 2400 to 64 000 bits/s. Packet-mode DTEs are usually connected to DCEs by leased lines.

- **Dial-up lines**

These are similar to public telephone lines and are available to more than one DTE.

Dial-up lines are also known as switched lines or telephone lines.

Data transmission over dial-up lines is usually asynchronous. Typical speeds of data transmission are in the range of 100 to 2400 bits/s. Character-mode DTEs are usually connected to DCEs by dial-up lines via a PAD.

4.4.2 Logical Channels

The physical line between a DTE and a DCE is divided into a number of **logical channels**. Logical channels allow many virtual circuits to exist on the physical line. Each logical channel is identified by a **logical channel number (LCN)**, contained in the packet header of every packet sent across a PSDN.

Before data can be sent across a PSDN, a virtual circuit must be established between a logical channel at the calling DTE and a logical channel at the called DTE. The circuits are referred to as "virtual" because a number of virtual circuits may use the same physical circuit to transmit packets from the calling DTE to the called DTE.

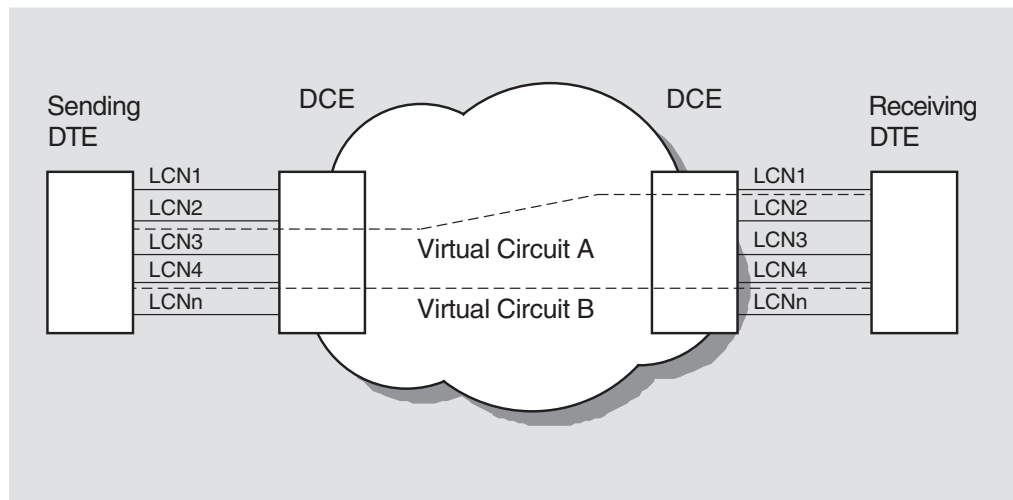
It is important to realize that the logical channel used between a DTE and DCE only has local significance. It is the responsibility of the PSDN to map the logical channels used by the calling and called DTEs into an end-to-end virtual circuit. This means that although a virtual circuit has end-to-end significance between the DTEs, different logical channels can be used by the virtual circuit on each side of the PSDN.

This concept is illustrated in Figure 4–8 in which two virtual circuits are depicted. Virtual Circuit A establishes an end-to-end connection using LCN 2 between the calling DTE and its associated DCE, and LCN 1 between the called DTE and its associated DCE. Similarly, Virtual Circuit B establishes an end-to-end connection using LCN 4 on both sides of the PSDN.

X.25 Networking Concepts

4.4 How Connections Are Made and Data Is Transferred

Figure 4–8 Virtual Circuits Versus Logical Channels



LKG–8135–93R

4.4.2.1 Virtual Circuits

There are two types of virtual circuits:

- Switched virtual circuits (SVCs)
SVCs are temporary circuits that require call setup and call clearing procedures. SVCs are established using call request packets that contain the DTE address of the remote DTE. Once established, the two DTEs can carry out two-way communications until the SVC is cleared. SVCs are cleared with clear request packets. Establishing an SVC is similar to making a telephone call.
- Permanent virtual circuits (PVCs)
PVCs are permanent circuits between two DTEs that need no call setup or call clearing procedures. Allocation of PVCs must be arranged with the supplier of the PSDN to be used. Once allocated, the use of a PVC can be requested using a management command. An allocated PVC is available for use until it is de-allocated.

A DTE may have many virtual circuits active at the same time over a single DTE-DCE interface. These circuits can be any combination of PVCs and SVCs.

4.4.2.2 Logical Channel Numbers

When you subscribe to a PSDN, you are allocated a range of LCNs for the different types of virtual circuits:

- Permanent virtual circuits (PVCs)
- Incoming (to the DTE) switched virtual circuits (ISVCs)
- Outgoing (from the DTE) switched virtual circuits (OSVCs)
- Incoming and outgoing (both ways) switched virtual circuits (SVCs)

An LCN is a 12-bit number. Some PSDNs use the first 4 bits as a **logical channel group number** (LCGN) to represent the use of the channel, and the remaining 8 bits to represent the LCN. Some PSDNs use the full 12 bits to represent the LCN.

X.25 Networking Concepts

4.4 How Connections Are Made and Data Is Transferred

4.4.3 DTE Addresses

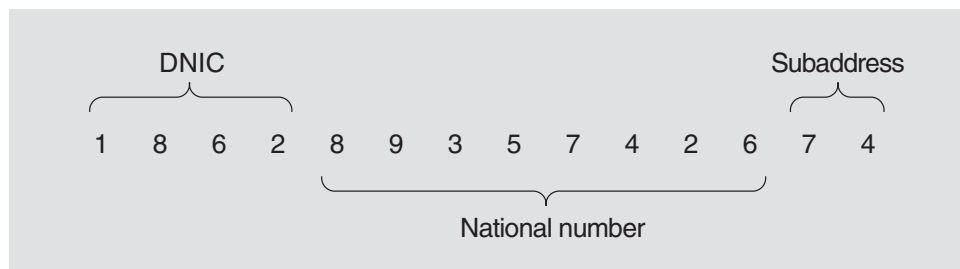
Each DTE has an address that the PSDN uses to route calls. DTE addresses are contained in the call request packets and are used for setting up SVCs. The address may be up to 15 digits long and consists of:

- A **Data Network Identification Code (DNIC)**, which identifies the country and distinguishes the PSDN from other PSDNs within the same country. This is usually the first four digits of the number.
- A national number, which identifies the DTE within the PSDN.
- An optional subaddress, which gives extra identification between the calling DTE and the called DTE. If included, this is usually the last two digits of the address.

DTE addresses may consist of up to 15 digits. However, for internetworking X.121 restricts international data numbers (that is, DTE addresses) to 14 digits.

An example of an X.121 format DTE address is shown in Figure 4–9.

Figure 4–9 DTE Address Example



LKG-6513-92R

Some networks use an abbreviated form of addressing to refer to DTEs on the same network. In such networks, an extra digit is often prefixed to the DTE address to indicate that the address of the called DTE is on a different network to the calling DTE.

4.4.4 Controlling the Flow of Packets

When a packet has been received by a called (remote) DTE, the PSDN sends a message to the calling (sending) DTE. This message is known as an **acknowledgment**. Because of the time it takes a packet to travel across a network, it is inefficient to wait for the acknowledgment of the packet before sending more. For this reason, PSDNs let you send a number of packets before you receive acknowledgment for the first one.

The maximum number of unacknowledged packets you are allowed to have on a virtual circuit is called the **window size**. Once the window size is reached, and acknowledgments for previously sent packets are received, more packets can be sent. Acknowledgments are often carried by incoming packets of data.

You can specify the window size you want to use, up to the maximum set by the PSDN. You may want to subscribe to larger window sizes if you have large amounts of data, or if there is a long delay between sending a packet and receiving the acknowledgment of that packet.

X.25 Networking Concepts

4.5 Optional Facilities of Public and Private PSDNs

4.5 Optional Facilities of Public and Private PSDNs

As well as switching packets, each PSDN offers optional facilities to its users. Some must be subscribed to, and others must be requested at the time the call is set up. A specific PSDN may support additional facilities that are not defined by the CCITT. A full list of such facilities can be obtained from the PSDN supplier.

According to CCITT X.25 (1984), these facilities fall into three categories:

- **General Optional User Facilities**

The optional user facilities that fall into this category allow tasks such as call redirection, barring of incoming calls, packet retransmission, and reverse charging to be performed. Refer to the *X.25 for OpenVMS Management Guide* for descriptions of the general optional user facilities supported.

- **Closed User Groups (CUGs)**

A CUG is a group of DTEs that are restricted to communicating with each other. The members of the CUG can prevent unwanted calls from non-CUG members (who are on the open network). Members of a CUG receive calls only from other members of that CUG.

Several optional facilities are available to CUGs. For example, within a CUG, one DTE can subscribe to facilities that allow it to make outgoing calls and receive incoming calls from the open network. This provides access to the open network for the specified DTE, but still prevents calls from the open network to other CUG members. Refer to the *X.25 for OpenVMS Management Guide* for descriptions of the optional facilities available to CUG members.

- **Bilateral Closed User Groups (BCUGs)**

A BCUG is a CUG consisting of only two DTEs. Access to BCUGs from the open network is more restricted than with normal CUGs because there is no optional facility allowing incoming access. Members of BCUGs can subscribe to a facility allowing outgoing access. Refer to the *X.25 for OpenVMS Management Guide* for descriptions of the optional facilities available to BCUG members.

4.5.1 DTE Parameters

DTE parameters govern the frame- and packet-level operation of DTEs. They specify properties such as window size and packet size. Various PSDNs support different parameters. For more information on the parameters supported, contact your PSDN supplier.

4.5.2 PAD Parameters

PSDNs offer optional PAD facilities that are represented by variables known as **PAD parameters**. You can set PAD parameters to indicate that you want to use a PAD facility and how you want to use it.

A set of PAD parameters is a **profile**; most networks define several profiles to suit different types of character-mode DTEs (X.29 terminals). When you connect to a PAD, you can select a profile to control the actions of the PAD. During a call, you can read and reset most of the PAD parameters.

Refer to the *X.25 for OpenVMS Management Guide* which describes the CCITT-defined PAD parameters, all supported by DECnet-Plus for OpenVMS.

4.6 Network Profiles

Digital supports the use of its X.25 products with a number of public and private PSDNs. A **network profile** contains the information necessary to support a particular PSDN including network parameters pertinent to a specific network. For example, it contains the default value and permissible range of values for the X.25 window size.

Details of the PSDNs supported by Digital's X.25 products can be obtained from your Digital representative. For details of the permissible network parameters for a specific network, refer to the documentation provided by your PSDN supplier.

4.7 PSDNs Supported by Digital's X.25 Products

Before a PSDN is supported by Digital, it has to go through Digital's own testing procedure. This is necessary to determine the network parameters to store in the associated network profile. The testing procedure produces a network profile for each PSDN. If you use the correct profile for the PSDN you are connecting to then your communications will be successful, and Digital will provide you with support in the event of any problems.

If you want to use a PSDN that is not currently supported by Digital, you can ask Digital to provide support for the network. Digital will then test the PSDN and produce a network profile for use with the PSDN. Your local Digital office can provide details on how this support service is provided.

Part III

Using DECnet-Plus Utilities

Part III provides information on using remote files with DECnet-Plus, using the DECnet-Plus login utility, and sending mail to DECnet-Plus nodes. This section includes the following chapters:

- Chapter 5 — DECnet Network Operations
- Chapter 6 — File Operations to and from Other DECnet and DECnet-Plus nodes

DECnet Network Operations

This chapter explains the functions you can perform in a DECnet network environment including remote login, file, and mail operations. It also explains how to develop command procedures and application programs that run over the network.

5.1 How You Can Use DECnet

You perform network operations as a natural extension of the input/output operations you perform on your own local system, because DECnet-Plus for OpenVMS networking capabilities are completely integrated into the operating system.

With appropriate access on a remote node, you can carry out general user operations over the network as easily as at the local node. Most DECnet-Plus for OpenVMS file-handling commands permit you to access files on remote systems in the same way as files on your own system as long as you have appropriate access.

5.1.1 Performing Operations using DCL Commands

You can use DCL commands to perform the following DECnet-Plus for OpenVMS operations over the network:

- Log in to another node on which you have an account.
- Access public directories or databases located on remote nodes.
- Display locally the contents of remote directories and files to which you have access.
- Copy files from node to node or append files on one node to a file on another node.
- Print files at the remote node where they reside, copy them to a remote printing device, or copy them to the local node for printing.
- Access and edit a file on a remote node using an OpenVMS editor.
- Create a new file in a remote directory.
- Delete or purge files from directories, search files, and compare the contents of files on different nodes.
- Perform sort and merge operations on remote files.
- Analyze the structure of files, convert their organization and format, and dump their contents in a specific format.
- Back up local files by using a remote save set on disk.
- Create, display, and delete logical names for nodes and devices that are to be included in remote file specifications.

DECnet Network Operations

5.1 How You Can Use DECnet

- Send electronic mail to, and receive mail from, a user on any other node in the DECnet network, by means of the Mail utility.
- Communicate interactively over the network by means of the Phone utility.
- Invoke help for DECnet-Plus by entering the following command:

```
$ help decnet_osi
```

5.2 Specifying Node Names

Node synonyms are required for layered products and utilities that do not yet support DECnet-Plus full names. Full name support is available for the SET HOST and Mail utilities, and for most DCL commands that use file specifications (such as COPY, DIRECTORY, etc.) and some utilities that use RMS.

5.2.1 Phase IV-Style Node Synonyms

Because a full name consists of the complete directory path from the root directory to the target object entry, directory, or soft link, full names can be long if a namespace has several levels of directories.

DECnet-Plus supports a 6-character Phase IV-style node name called the node synonym. A node synonym is a soft link that points to the full name of a node object entry. Node synonym soft links enable applications that do not support the length of full names to continue to use 6-character node names.

Node synonyms should not be viewed as a way for users to avoid typing the full name of a node. Logical names, and a feature called the local root, are faster and more convenient methods of shortening names. Both logical names and local roots are for use only on the system where they are defined; unlike node synonyms, they do not have global meaning throughout the namespace. Also, logical names and local roots can be used to name any resource (such as a disk or an application), not just node names.

5.2.2 Name Abbreviation Using Local Roots

The local root is a prefix that DECnet-Plus obtains automatically by stripping off the rightmost simple name of the local node's DECnet-Plus full name. For example, a node named IAF:.DIST.QA01 would have IAF:.DIST as its local root. Users will likely want to refer frequently to other names in the hierarchy in which their local node is named. The local root capability makes such name references more convenient by allowing users to omit the part of the full name that is also part of their node's full name.

DECnet-Plus Session Control creates the local root under the fixed name `dnsroot` in the `dns$system` logical name table.

5.2.3 Logging In to Other DECnet Nodes

To gain access to the network, log in to your account on the OpenVMS system. Before you perform an operation over the network, you can check the availability of the network by entering the following command:

```
$ show logical net$startup_status  
"net$startup_status" = "running-all" (lnm$system_table)
```

This command returns with a display indicating that all network software is loaded and running.

After you log in to a network node, you may be able to log in to other nodes on the same network. If you are authorized to access an account on another OpenVMS node or on a non-OpenVMS node that supports DECnet, you can log in to that node over the network. To log in to the other node, enter the DCL command SET HOST, specifying the remote node name, and follow the login procedure the remote node uses. Once you are logged in to the remote node, you can perform all general user operations on that system as though it were the local node:

To set host to another OpenVMS system, enter:

```
$ set host node-name
```

To set host to an Digital UNIX system with its synonym registered in DNS, enter:

```
$ set host node-name
```

To set host to another system using full name support, enter:

```
$ set host namespace:.abc.xyz
```

5.3 Accessing Remote Files

This section describes the format of the remote file specification and the access controls that affect access to remote files. This section also explains how to use logical names in specifying remote directories and files, and how to specify remote files located on OpenVMS clusters.

5.3.1 Specifying Files

You can access remote files by simply including in the file specification the name of the remote node on which the file is located. You can access files that are protected if the owner has granted you access, either by a proxy account or by providing you with the name and password of the account.

The full format for a remote file specification for DECnet-Plus for OpenVMS nodes is as follows:

```
node"username password"::device:[directory]filename.type;version
```

For example, to identify the file `example.lis` residing in the directory `information` on device `dba1`, which belongs to user `cmiller` whose password is `techwriter`, at node `boston`, you would use the following file specification:

```
boston"cmiller techwriter"::dba1:[information]example.lis
```

5.3.2 Specifying Files on a Non-OpenVMS System

If a file resides on a non-OpenVMS system, enclose the name of the file in quotation marks. For example, to access a file named `/usr/users/user/test` on an Digital UNIX node named `U32`, you would use the following format for the file specification:

```
U32"user password"::"/usr/users/user/test"
```

Note

DECnet-Plus for Digital UNIX file specification is case sensitive.

DECnet Network Operations

5.3 Accessing Remote Files

5.3.3 Protecting Files

You can use the SET PROTECTION command to protect a file or directory against unauthorized access. One way to access a protected remote file is to supply the user name and password of a user on the remote system who does have access to the file. If the user `walker`, who has the password `projmgr`, has access to the file, you could use the following file specification:

```
boston"walker projmgr"::DBA1:[INFORMATION]EXAMPLE.LIS
```

To avoid using a password in a file specification to be transmitted over the network, you may want to have a *proxy* account at the remote node that permits you to access specific directories and files as though you were a local user. If you have proxy access to a remote file, you can omit the access control information when specifying that file name, even if the file is protected against outside access. For example, use this file specification to access the file `tasks.lis` in the directory `project` on device `work` at remote node `austin`, at which you have a proxy account:

```
austin::work:[project]tasks.lis
```

If the remote node system manager has established a default DECnet-Plus for OpenVMS account for the remote node, you can specify a null access control string in the remote file specification to invoke the default DECnet-Plus for OpenVMS account. For example, the following file specification causes the file `trends.dat` at remote node `london` to be accessed using the default DECnet-Plus for OpenVMS account:

```
london" " : :dba0:[market]trends.dat
```

When you access a remote file, a process at the remote system actually performs the file access on your behalf. The remote process follows the rules normally used to access files on that system. The rights and privileges the remote process uses to access the file depend on the user name supplied. The user name can be one of the following (in order of precedence):

- A user name that you supply explicitly in an access control string included in the remote file specification. (If you specify a null access control string, the user name in the default DECnet-Plus for OpenVMS account, if one exists, is used.)
- The user name in your proxy account at the remote node, if one exists.
- The user name in the default access account, if one exists, for the file access listener (FAL) object at the remote node.
- The user name in the default DECnet-Plus for OpenVMS account, if one exists, at the remote node (by default, that user name is `decnet`).

5.4 File Operations

The following sections illustrate ways you can use DECnet-Plus for OpenVMS commands.

5.4.1 Displaying Remote Directories and Files

To list the files in a remote directory, use the `DIRECTORY` command. For example, the following command lists all files in the directory `[comments.public]` at remote node `concord`:

```
$ directory concord::disk1:[comments.public]
```

The `TYPE` command lets you display the contents of one or more remote files to which you have access. For example, to display the contents of the unprotected file `members.lis` in directory `[patriots]` at remote node `concord`, use the following command:

```
$ type concord::disk1:[patriots]members
```

For example, the following command displays the file `redcoats.lis` in directory `[british]` on the default device at node `lexington`:

```
$ type lexington::[british]redcoats
```

5.4.2 Public Directories

Information of interest to a number of users on the DECnet-Plus for OpenVMS network may be stored in central directories or databases accessible to everyone on the network. To make access to a public directory easier, define a logical name for the public directory. For example, you can use the logical name `public` to define the public directory `[comments.public]` at remote node `concord`:

```
$ define public concord::disk1:[comments.public]
```

Users logged in to any node on the network can then access the file `freedom.txt` located in the directory `[comments.public]`, for example:

```
$ directory public  
$ type public::freedom.txt)
```

They can also copy the file to the local node and then print it, as described in the next section.

5.4.3 Copying and Printing Remote Files

Use the `COPY` command to copy a file from one node to another. As part of the copy operation, you can create a new file from existing files.

5.4.3.1 The COPY Command

The following command copies the file `liberty.dat` on node `boston` to a new file of the same name on remote node `britain`:

```
$ copy boston::disk:liberty.dat;2  
_To: britain::dba0:[product]liberty.dat
```

The following command copies the remote file `liberty` from Digital UNIX node `boston` to a new file on local node `britain`:

```
$ copy boston::"/usr/users/user/test" liberty.dat
```

Both of the following commands copy the file `traitors.lis` from the local node to a file with the same name at remote node `concord`:

```
$ copy traitors.lis concord::disk1:[patriots]traitors.lis  
$ copy traitors.lis concord::disk1:[patriots]
```

DECnet Network Operations

5.4 File Operations

The following command is the wildcard form of the COPY command. The latest versions of all files in the user's default directory at the local node are copied to the remote node lexington.

```
$ copy *.* lexington::[british]*.*
```

The next command copies two local files, user1.dat and user2.dat, to a single new file, users.dat, at remote node lexington:

```
$ copy user1.dat,user2.dat lexington::[british]users.dat
```

To specify explicitly the access privileges of a particular account on a remote node when copying a file from that node, include the user name and password for that account in the file specification, as in the following example:

```
$ copy lexington"revere silver"::[statistics]summary.lis *
```

5.4.3.2 The APPEND Command

Use the APPEND command to add the contents of one or more input files to the end of an output file located at a different node. For example, to add the contents of the files data1.lis and data2.lis at remote node lexington to the end of the file results.lis at node britain, use the following command:

```
$ append lexington"revere silver"::[liberty]data1.lis,data2.lis  
_To: britain::dba0:[product]results.lis
```

5.4.3.3 The PRINT/REMOTE Command

To queue a file for printing at the remote node on which it exists, specify the remote file specification in the PRINT/REMOTE command. The PRINT/REMOTE command does not copy the file to the remote node; you must enter a separate COPY command if the file does not reside at the remote node on which it is to be printed. For example, the following commands cause the local file witches.lis to be copied to the default DECnet directory at remote node salem and queued for printing at node salem:

```
$ copy witches.lis salem::work:[project]  
$ print/remote salem::work:[project]report
```

Alternatively, you can copy a file to a printing device on the remote system. If the device is spooled (for example, a line printer), the file will be stored temporarily on disk and entered in the print queue for the device. For example, the following command performs the same operation as the two previous commands, causing the local file to be printed at the line printer at node salem under the default nonprivileged DECnet account:

```
$ copy report.lis salem::lpa0:
```

Note that the /REMOTE qualifier is required in the PRINT command whenever a remote file is specified, and that you cannot include any other qualifiers with this command. The PRINT/REMOTE command supplies a default file type of LIS.

5.4.4 Other Remote File Operations

In addition to displaying, copying, and printing remote files, you can also:

- Edit
- Delete
- Purge
- Search

- Compare
- Sort
- Merge
- Examine
- Back up

5.4.4.1 Edit

To invoke the EDT editor to edit the file `story.txt` in the directory `[manuscript]` on remote node `london`, enter:

```
$ edit/edt london::[manuscript]story.txt
```

5.4.4.2 Delete

To delete the file `details.lis;2` in the directory `information` at remote node `boston`, use this command:

```
$ delete boston"walker projmgr"::dba1:[information]details.lis;2
```

If you have a proxy account that allows you full access to the directory `suggestions` on remote node `austin`, you can delete all files in that directory, as follows:

```
$ delete austin::work:[suggestions]*.*;*
```

5.4.4.3 Purge

To purge all but the two highest-numbered versions of each file of the type LIS in the directory `proposals` at remote node `austin`, use this command:

```
$ purge austin::work:[proposals]*.lis/keep=2
```

5.4.4.4 Search

The following SEARCH command causes the files `members.lis` and `data.lis` at remote node `concord` to be searched for all occurrences of the character string `name`:

```
$ search concord::disk1:[club]members.lis,data.lis
_String(s):      name
```

5.4.4.5 Compare

This command compares the two highest-numbered versions of the file `test.dat` in the nonprivileged default DECnet account on remote node `boston`:

```
$ differences boston::test.dat
```

The following command compares two remote files and displays any differences found. The first file is `test.dat` at remote node `boston` and the second file is `test.dat` at remote node `london`.

```
$ differences boston::test.dat london::dba0:[product]test.dat
```

DECnet Network Operations

5.4 File Operations

5.4.4.6 Sort

The following command requests a default alphanumeric sort of the records in the file `random.fil` at remote node `boston`. The `SORT` program sorts the records on the basis of the contents of the first seven characters in each record and writes the sorted list into the output file `alphanm.srt` created in the default directory at the local node.

```
$ sort/key=(position:1,size:7) -
_ $ boston::dba1:[records]random.fil alphanm.srt
```

5.4.4.7 Merge

The example of a merge command shown below causes two identically sorted files, `file1.srt` and `file2.srt`, on the directory `project` at remote node `austin` to be merged into an output file. This output file, `mergefile.dat`, is created at the current default directory at the local node.

The input file qualifier `/CHECK_SEQUENCE` is specified to ensure that the input files are sorted in the correct order.

```
$ merge/key=(position:1,size:30) -
_ $ austin::work:[project]file1.srt,file2.srt/check_sequence -
_ $ mergefile.dat)
```

5.4.4.8 Examine

The following command analyzes the structure of the file `run.dat` at remote node `austin`:

```
$ analyze/rms_file austin::work:[production]run.dat
```

The following command causes records from the file `sales.tmp` at the local node to be added sequentially to the end of the output file `sales.cmd` at remote node `boston`. The file `sales.tmp` is sequential with variable-length record format, and the file `sales.cmd` is sequential with stream record format. When the `Convert` utility loads records from the input file to the output file, it changes the record format.

```
$ convert/append sales.tmp boston::dba1:[records]sales.cmd
```

Use the `DUMP/RECORDS` command to display the contents of remote files in ASCII, hexadecimal, decimal, or octal representation. The `DUMP` command qualifiers `/ALLOCATED` and `/BLOCKS` are not supported in the network context. The following command dumps the contents of the file `calc.dat`, which resides at remote node `boston`. The command formats the output both in octal words and in character strings, and queues the output to the system printer at the local node.

```
$ dump/records/octal/word
_ File(s): boston::dba1:[records]calc.dat/printer
```

5.4.4.9 Back Up

The following command saves the files in the directory `sched` on disk `DB1` at the local node in the `BACKUP` save set `sch.bck` at remote node `miami`. The `/SAVE_SET` qualifier is required to identify the output specifier as a save set on a `Files-11` medium.

```
$ backup
_ From: db1:[sched]*.*
_ To: miami::dba2:[save]sch.bck/save_set
```

5.5 Using the Mail and Phone Utilities

Any user can send electronic mail to, and receive mail from, any other user on the network. Mail can be exchanged between all DECnet and DECnet-Plus nodes.

5.5.1 The MAIL Command

To address the mail message to the intended recipient on the remote node, you normally use the format *nodename::username*. For example, to send mail to user longfellow on remote node salem, invoke mail and enter the following:

```
$ mail
MAIL> send
To: salem::longfellow
```

If the network connection to the remote node is not available, you receive the following message:

```
_SYSTEM-F-UNREACHABLE, remote node is not currently reachable
```

When someone on a remote node sends you mail, the sender is identified by node name as well as user name. For example, if user alcott at node concord sends you a mail message over the network, the sender is identified in the following way:

```
From: CONCORD::ALCOTT
```

You can receive notification of mail arriving from a remote node. The notice displayed on your screen is in the following form:

```
New mail on node 'local-nodename' from 'remote-nodename::username'
```

For example, if user alcott on node concord sends you mail on node literature, this notice is displayed:

```
New mail on node LITERATURE from CONCORD::ALCOTT
```

For example, user bronte is logged in to node literature in a cluster that uses the alias CLUST1. When she receives a reply to a message she sent user alcott at remote node concord, the message will indicate the cluster node name CLUST1 (rather than the node name literature), as in the following:

```
From: CONCORD::ALCOTT
To: CLUST1::BRONTE
```

Additionally, the mail notification that user BRONTE receives may indicate that the mail was received at a different node (for example, BOOKS) in the same cluster, as in the following:

```
New mail on node BOOKS from CONCORD::ALCOTT
```

5.5.1.1 Sending Files and Long Messages

You can send either messages or files over the network. If you are composing a long message for transmission to a remote node, you may prefer to use an editor to create the message file and then invoke MAIL to transmit the file. This method permits you to avoid the possibility of losing the network connection before you complete your message.

DECnet Network Operations

5.5 Using the Mail and Phone Utilities

5.5.2 The Phone Utility

To contact OpenVMS users over the network, you can also use the Phone utility, which allows you to have an “online” conversation with a user on another OpenVMS node that supports Phone.

To receive messages from the Phone utility, one of the following conditions must be met.

- Default access must be provided by a default access account for the Phone utility or by the default DECnet account.
- The sender must have a proxy account on your system or include, in the command, the name and password for an account on your system.

To address a user on a remote node, use the format *nodename::username*. Your outgoing connection identifies your local node. During your conversation, Phone creates a number of incoming links addressed to your node. You should not use a cluster alias node name with the Phone utility because links addressed to a cluster alias node name can be assigned to any node in the cluster.

File Operations to and from Other DECnet and DECnet-Plus Nodes

This chapter contains material to help you use DECnet-Plus for OpenVMS to initiate remote file operations in a heterogeneous network environment. This chapter discusses restrictions on using DCL commands and RMS service calls to access files on the following types of remote systems:

- OpenVMS to Digital UNIX or ULTRIX
- OpenVMS to MS-DOS
- OpenVMS to RSX using RMS-based FAL
- OpenVMS to OpenVMS

The chapter is organized by operating-system type: one section for each system with which your OpenVMS operating system running DECnet-Plus for OpenVMS can communicate. Each section describes differences in file system operation between the two systems and constraints on the use of OpenVMS file processing commands. The restrictions on the remote file operations you can perform from a DECnet-Plus for OpenVMS node to a particular node result from file system design differences and DECnet implementation restrictions between the systems.

The appropriate section for each remote system itemizes the OpenVMS Record Management Services (RMS) features that are supported between DECnet-Plus for OpenVMS systems, but are not supported when accessing files on a different system. The chapter also discusses limitations on the Digital Command Language (DCL) commands that you can use when communicating with the remote node. Throughout this chapter, comments are provided to help you handle the differences in file system design.

6.1 General DECnet Restrictions

This section is a brief summary of OpenVMS RMS features that are not supported by DECnet-Plus for OpenVMS for remote file access. The list is not complete; it is meant only to highlight the more important differences between local and remote file access capabilities.

- The following OpenVMS RMS service calls are not supported for network use:
 - \$ENTER
 - \$NXTVOL
 - \$REMOVE
- The Terminal XAB is not supported for network operations; it is ignored.
- Protection XAB fields that support access control lists are ignored for network operations.

File Operations to and from Other DECnet and DECnet-Plus Nodes

6.1 General DECnet Restrictions

- Only one data stream per open file is allowed. That is, the multistream (MSE) bit option of the file sharing (SHR) field of the FAB is not supported for network use.
- Access to files on magnetic tapes mounted on a remote OpenVMS operating system is not supported. You can, however, copy files from a local magnetic tape to disk on a remote node.
- When multiple Allocation XABs are linked to the FAB, they must be in ascending order by area number (AID field). Similarly, when multiple Key Definition XABs are used, they must be in ascending order by key of reference (REF field).
- File protection information may not be completely preserved if the two nodes do not fully support each other's protection attributes. An example of this incompatibility occurs between RSX-11M-PLUS and OpenVMS operating systems. Although both systems represent their protection masks as RWED, RSX-11M-PLUS interprets that as Read, Write, Extend, and Delete, while the OpenVMS operating system interprets RWED as Read, Write, Execute, and Delete. This results in the "E" protection field being unmappable between these two systems.
- The Journaling XABs are not supported.
- File monitoring is not supported.
- The user file open (FAB\$V_UFO) file processing option is not supported.

6.2 OpenVMS to Digital UNIX or ULTRIX Network Operations

This section pertains to an OpenVMS node communicating with a Digital UNIX or ULTRIX node running DECnet-Plus for Digital UNIX or ULTRIX. The discussion focuses on file operations initiated from the OpenVMS node, to access remote files by means of the FAL at the Digital UNIX or ULTRIX node.

6.2.1 File System Constraints

The file systems used by Digital UNIX or ULTRIX and OpenVMS are dissimilar in many respects. A fundamental difference between them involves the handling of file attribute information. When you create a file on an OpenVMS operating system, attribute information about the file is stored in a header block on disk for use when the file is subsequently opened.

The implication is that the structure of an established file cannot change. In contrast, Digital UNIX or ULTRIX does not save attribute information such as file format with a file; it expects you to provide this information when you open the file. File attribute information, however, is not an input to OpenVMS RMS when a file is opened.

To provide transparent access to files on a remote Digital UNIX or ULTRIX operating system, OpenVMS RMS restricts the types of files you can create and open on the remote node. When you access a Digital UNIX or ULTRIX file in record mode, OpenVMS RMS treats the file as having STREAM_LF (STMLF) format.

File Operations to and from Other DECnet and DECnet-Plus Nodes

6.2 OpenVMS to Digital UNIX or ULTRIX Network Operations

6.2.1.1 File Formats and Access Modes

Because of differences in file system design, the following types of file and access method are not supported by OpenVMS when communicating with a Digital UNIX or ULTRIX node:

- File organizations and record formats
 - Sequential
 - Fixed length (FIX) without implied carriage control
 - STREAM_CR (STMCR)
 - Stream (STM)
 - Variable length (VAR) without implied carriage control
 - Variable with fixed control (VFC)
 - Relative
 - All formats
 - Indexed
 - All formats
- Record attributes
 - FORTTRAN carriage control (FTN)
 - Print file carriage control (PRN)
 - None specified (embedded carriage control)
- Record access modes
 - Random access by relative record number
 - Random access by key value
 - Random access by record file address
 - Block I/O

For record mode access, the only file type in common between the two systems is a sequential file in STMLF (STREAM_LF) format. For convenience, however, when you are transferring a file to a Digital UNIX or ULTRIX node, OpenVMS RMS automatically converts an OpenVMS sequential file with fixed or variable format and implied carriage control to a sequential file with stream format and embedded carriage control. This automatic conversion is performed during a file create operation, and OpenVMS RMS returns an alternate success code (RMS\$_CVT_STM) to indicate that the file format has been modified.

Note also that when a STREAM-LF format file is retrieved from a Digital UNIX or ULTRIX node, OpenVMS RMS automatically changes the record attribute from embedded carriage control to implied carriage control.

To transfer files that cannot be copied directly, enter the following DCL command:

```
$ CONVERT/FDL=STMLF.FDL input-file output-file
```

The FDL control file STMLF.FDL contains the following information:

```
FILE
      ORGANIZATION      sequential
RECORD
      FORMAT             STREAM_LF
      CARRIAGE_CONTROL  none
```

The CONVERT command and associated FDL control file transform the input file to stream format with embedded carriage control and then copy it to the remote node according to the output file specification.

File Operations to and from Other DECnet and DECnet-Plus Nodes

6.2 OpenVMS to Digital UNIX or ULTRIX Network Operations

6.2.1.2 OpenVMS RMS Interface

The following OpenVMS RMS features, supported between two OpenVMS nodes, are not supported between an OpenVMS node and a Digital UNIX or ULTRIX node:

- OpenVMS RMS service calls
 - \$DELETE \$DISPLAY \$EXTEND \$FIND
 - \$FREE \$RELEASE \$RENAME \$REWIND
 - \$TRUNCATE \$UPDATE
- RMS extended attribute blocks
 - Allocation XAB
 - Key Definition XAB
 - Summary XAB
- Significant fields and bit options of the FAB
 - ALQ (allocation quantity) field
 - DEQ (default extend quantity) field
 - CBT (contiguous-best-try) bit of FOP field

6.2.1.3 File Specifications

The general format of a file specification for naming a file on a remote Digital UNIX or ULTRIX operating system is as follows:

```
node::name
```

The following are the major differences in syntax between file specifications on Digital UNIX or ULTRIX and on OpenVMS:

- No explicit device names are allowed. Instead, Digital UNIX or ULTRIX has a concept of special files.
- File names on Digital UNIX or ULTRIX are case sensitive (uppercase or lowercase).

Because of these differences, most accesses to a Digital UNIX or ULTRIX operating system require a foreign file specification. Without the foreign file specification syntax, the name is converted to uppercase by OpenVMS, and is then unlikely to match files on the Digital UNIX or ULTRIX operating system. The OpenVMS concepts of device and directory do not match the Digital UNIX or ULTRIX concept of path, nor does Digital UNIX or ULTRIX support separate file type or version fields. Therefore, OpenVMS-related name processing does not work with Digital UNIX or ULTRIX file names.

6.2.2 DCL Considerations

Of the OpenVMS DCL commands that you can use over the network, the following are not supported between OpenVMS and a Digital UNIX or ULTRIX node:

- ANALYZE/RMS_FILE
- BACKUP
- OPEN/WRITE
- RENAME

File Operations to and from Other DECnet and DECnet-Plus Nodes

6.2 OpenVMS to Digital UNIX or ULTRIX Network Operations

6.2.2.1 COPY

The /ALLOCATION and /EXTENSION qualifiers to the COPY command are not supported and are ignored if specified.

6.2.2.2 DIRECTORY

When you enter a DIRECTORY/FULL command to examine a Digital UNIX or ULTRIX file, the information displayed differs in the following respects from that displayed for an OpenVMS file:

- The file owner is displayed as [0,0] to indicate that this information is not available.
- The file REVISION number is not shown and file REVISION date and time information is not available from the Digital UNIX or ULTRIX operating system.

6.3 OpenVMS to MS-DOS Network Operations

This section pertains to an OpenVMS node communicating with a PC running the DECnet feature of PATHWORKS for DOS. The discussion focuses on file operations initiated from the OpenVMS node, to access remote files by means of the FAL at the MS-DOS node.

6.3.1 File System Constraints

The file systems used by MS-DOS and OpenVMS are dissimilar in many respects. A fundamental difference between them involves the handling of file attribute information. When you create a file on an OpenVMS operating system, attribute information about the file is stored in a header block on disk for use when the file is subsequently opened. The implication is that the structure of an established file cannot change. In contrast, MS-DOS does not save attribute information such as file format with a file; it expects you to provide this information when you open the file. File attribute information, however, is not an input to OpenVMS RMS when a file is opened.

To provide transparent access to files on a remote MS-DOS system, OpenVMS RMS restricts the types of file that you can create and open on the remote node. When you access an MS-DOS file in record mode, OpenVMS RMS treats the file as having stream format.

6.3.1.1 File Formats and Access Modes

Because of differences in file system design, the following types of file and access method are not supported by OpenVMS when communicating with an MS-DOS node:

- File organizations and record formats
 - Sequential
 - Fixed length (FIX) without implied carriage control
 - Stream_CR (STMCR)
 - Stream_LF (STMLF)
 - Variable length (VAR) without implied carriage control
 - Variable with fixed control (VFC)
 - Relative
 - All formats
 - Indexed
 - All formats

File Operations to and from Other DECnet and DECnet-Plus Nodes

6.3 OpenVMS to MS-DOS Network Operations

- Record attributes
 - FORTRAN carriage control (FTN)
 - Print file carriage control (PRN)
 - None specified (embedded carriage control)
- Record access modes
 - Random access by relative record number
 - Random access by key value
 - Random access by record file address

For record mode access, the only file type in common between the two systems is a sequential file in STM (stream) format. For convenience, however, when you are transferring a file to an MS-DOS node, OpenVMS RMS automatically converts an OpenVMS sequential file with fixed or variable format and implied carriage control to a sequential file with stream format and embedded carriage control. This automatic conversion is performed during a file create operation, and OpenVMS RMS returns an alternate success code (RMS\$_CVT_STM) to indicate that the file format has been modified.

Note also that when a stream format file is retrieved from an MS-DOS node, OpenVMS RMS automatically changes the record attribute from embedded carriage control to implied carriage control.

In general, you can copy text files created by the TPU or EDT editor to an MS-DOS operating system. OpenVMS batch log files, however, are stored in VFC format, and cannot be copied in that form to an MS-DOS operating system. To transfer this type of file, enter the following DCL command:

```
$ CONVERT/FDL=STM.FDL input-file output-file
```

The FDL control file STM.FDL contains the following information:

```
FILE
      ORGANIZATION      sequential
RECORD
      FORMAT             stream
      CARRIAGE_CONTROL  none
```

The CONVERT command and associated FDL control file transform the input file to stream format with embedded carriage control and then copy it to the remote node according to the output file specification.

6.3.1.2 OpenVMS RMS Interface

The following OpenVMS RMS features, supported between two OpenVMS nodes, are not supported between an OpenVMS node and an MS-DOS node:

- OpenVMS RMS service calls
 - \$DELETE \$DISPLAY \$EXTEND \$FIND
 - \$FREE \$RELEASE \$RENAME \$REWIND
 - \$TRUNCATE \$UPDATE \$WRITE
- RMS extended attribute blocks
 - Allocation XAB
 - Key Definition XAB
 - Summary XAB
- Significant fields and bit options of the FAB
 - ALQ (allocation quantity) field

File Operations to and from Other DECnet and DECnet-Plus Nodes

6.3 OpenVMS to MS-DOS Network Operations

DEQ (default extend quantity) field
CBT (contiguous-best-try) bit of FOP field

6.3.1.3 File Specifications

The general format of a file specification for naming a file on a remote MS-DOS operating system is as follows: `node::"device:\directory\name"`

The major difference in syntax between file specifications on MS-DOS and on OpenVMS is that the directory components of an MS-DOS file specification are in an incompatible format. For example: `\directory\`

As a result, you must use quoted strings when you access these MS-DOS files from an OpenVMS operating system.

On DOS-based systems, the FAL object accepts incoming requests using file specifications in OpenVMS syntax and maps those requests to file specifications for DOS. For example:

```
$ DIRECTORY PC::[REPORT]
```

This directory specification is mapped to the following directory specification:

```
$ DIRECTORY PC::\report\*.*
```

DOS file specifications are restricted to file names of eight characters, file extensions of three characters, and do not support version numbers.

6.3.2 DCL Considerations

Of the OpenVMS DCL commands that you can use over the network, the following are not supported between OpenVMS and an MS-DOS node:

- ANALYZE/RMS_FILE
- APPEND
- BACKUP
- OPEN/WRITE
- RENAME

6.3.2.1 COPY

The /ALLOCATION and /EXTENSION qualifiers to the COPY command are not supported and are ignored if specified.

6.3.2.2 DIRECTORY

When you enter a DIRECTORY/FULL command to examine an MS-DOS file, the information displayed differs in the following respects from that displayed for an OpenVMS file:

- The file owner identifier is displayed as [0,0] to indicate that this information is not available.
- The file ID identifier is displayed as NONE to indicate that this information is not available.
- The file attributes version limit identifier is displayed as 0 to indicate that this information is not available.
- The file REVISION number is not shown and file REVISION date and time information is not available from the MS-DOS operating system.

File Operations to and from Other DECnet and DECnet-Plus Nodes

6.4 OpenVMS to RSX Network Operation Using RMS-based FAL

6.4 OpenVMS to RSX Network Operation Using RMS-based FAL

This section pertains to an OpenVMS node communicating with an RSX node running either DECnet-11M or DECnet-11M-PLUS where the RSX file access listener (FAL) calls RMS-11 to perform local file operations. The discussion focuses on file operations initiated from the OpenVMS node, to access remote files by means of the FAL at the RSX node.

The following restrictions are related to incompatible features in file system design between the two systems.

6.4.1 File Formats and Access Modes

The following types of file and record attributes are not supported by OpenVMS when communicating with an RSX node running the RMS-based FAL:

- File organizations and record formats
 - Sequential Stream_CR (STMCR)
 Stream_LF (STMLF)
 - Indexed All prologue three formats
 - With 64-bit binary (BN8) key types
 - With 64-bit integer (IN8) key types
 - With collating (COL) key types
 - With descending key types (DSTG, DIN2, DBN2, DIN4, DBN4, DIN8, DBN8, DPAC, DCOL)
- Record attributes
 - Record attributes are compatible.
- File access modes
 - Modes are compatible.

6.4.2 OpenVMS RMS Interface

The following OpenVMS RMS features, supported between two OpenVMS nodes, are not supported between an OpenVMS node and an RMS-based RSX node:

- OpenVMS RMS service call:
 - \$RELEASE
- Significant fields and bit options of the FAB and CBT (contiguous-best-try) bit of FOP

6.4.3 File Specifications

The general format of a file specification for naming a file on a remote RSX-11M or RSX-11M-PLUS system is as follows:

```
node::device:[directory]name.type;version
```

The following are major differences in syntax between file specifications used on RSX and OpenVMS:

- RSX operating systems do not support dollar sign (\$), underscore (_) and hyphen (-) characters in file name components.
- The directory component in an RSX file specification cannot be a named directory list, such as [A.B.C]; it must be in UIC (user identification code) format, such as [15,1].

File Operations to and from Other DECnet and DECnet-Plus Nodes

6.4 OpenVMS to RSX Network Operation Using RMS-based FAL

- The file name component has a maximum length of nine characters and the file type cannot exceed three characters. RSX operating systems return an error if you specify a longer file name or file type.
- RSX operating systems use octal version numbers in file specifications whereas the OpenVMS operating system uses decimal version numbers.

6.4.4 DCL Considerations

Of the OpenVMS DCL commands that you can use over the network, the OPEN/WRITE command is not supported between OpenVMS and an RMS-based RSX node.

6.4.4.1 COPY

Because RSX-11M and RSX-11M-PLUS operating systems use octal version numbers in file specifications, any attempt to copy a file with a version number containing an 8 or 9 is rejected by the remote system. For example:

```
$ copy a.dat;9 rsx:.*.*
%COPY-E-OPENOUT, error opening rsx::a.dat;9 as output
-RMS-F-FNM, error in file name
```

There are two ways to circumvent this problem. You can specify an appropriate octal version number in the output file specification, or you can specify a null or zero version number in the output file specification to force highest version number processing on the remote node. This latter technique is particularly useful when several files are copied with one DCL command. For example:

```
$ copy a.dat;9 rsx::a.dat;11
$ copy b.dat;28 rsx:.*.*;
$ copy b.dat;28 rsx:.*.*;0
$ copy *.dat rsx:.*.*;0
```


Part IV

Glossary

DECnet-Plus Glossary

This DECnet-Plus Glossary covers terms that explain the features and operation of DECnet-Plus for OpenVMS covering these areas:

- OSI
- DECnet-Plus
- X.25
- WAN device drivers (WANDD)
- Ethernet and CSMA-CD LANs
- Digital Distributed Name Service (DECdns)
- Digital Distributed Time Service (DECdts)
- File Transfer, Access, and Management (FTAM)
- Virtual Terminal (VT)
- FTAM and VT gateways

Table 1 at the end of the glossary lists open-networking-related acronyms and their meanings. Table 2 lists the ISO and IEC acronyms in transition.

G.1 Definitions

A-ASSOCIATE request

Association Control Service Element (ACSE) service that initiates a communications channel for data transfer at the Application layer between two peer service users; used by applications such as FTAM and OSAK.

absolute time

Point on a time scale. For DECdts, absolute time references the Coordinated Universal Time (UTC) standard.

abstract syntax

Implementation-independent way of representing Application layer data or application or presentation protocol control information (PCI). Consists of a group of data types that describe a set of data structures that several applications can share.

abstract syntax notation

Notation in which an abstract syntax is described. The rules of abstract syntax notation are independent of the encoding techniques used to represent them.

Abstract Syntax Notation One

See ASN.1.

acceptor

Peer entity that responds to a service request.

access context

FTAM method of specifying the type of information for a file-access data unit (FADU) that can be accessed.

access control entry (ACE)

Defined as:

- DECdns — Entry in an access control set (ACS). Each entry includes the name of a **principal** (an individual or a group of individuals) and the access rights that the principal has to the associated name in the namespace.
- X.25 — Entry in an access control list (ACL) that grants or denies access to a particular system object.

access control information

Character string with login information that validates connect or login at a remote host. On an OpenVMS system, for example, user name and password are access control information.

access control list (ACL)

List of access control entries (ACEs) that grant or deny access to a particular system object.

access control set (ACS)

Attribute associated with DECdns clearinghouses, directories, object entries, and soft links that determines who has access to them, and what the access rights are; consists of one or more ACEs.

access plug-in

Plug-in responsible for the director's end of the Management Access Protocol.

access rights

Set of DECdns assignable rights that determines what users can do with clearinghouses, directories, and the names they contain. The access rights are read, write, delete, test, and control.

ACE

See **access control entry**.

acknowledgment

Message sent from a PSDN indicating that a packet has been received.

ACL

See **access control list**.

ACS

See **access control set**.

ACSE

See **Association Control Service Element**.

actively associated task

Task that asks OSI transport to associate it with a particular transport service access point (TSAP). When a connection request for that TSAP arrives, OSI transport directs the request to the task associated with that TSAP.

activity

Logical piece of work.

activity attribute

Any of a set of attributes that describe the current use of the FTAM file service; descriptive tool for describing specific actions that are local to a given FTAM regime and any nested regime.

adaptive routing

Feature of DECnet-Plus routing in which the intermediate system dynamically determines the most cost-effective path currently available, forwards messages through the network along this path, and automatically reroutes messages if a circuit becomes disabled. The path, or choice, depends on the current information stored in the routing database, which is updated regularly.

address prefix

Some leading portion of a network service access point (NSAP); can be of any length, from zero digits up to the full length of the NSAP; can be used in a reachable-address table to indicate that any packets with a destination NSAP beginning with a specified address prefix are to be routed through a specified circuit.

address resolution

Session Control function that maps from a DECnet-Plus full name to the identifiers of protocols and corresponding addresses that are mutually supported by the local system and the remote system on which the named object resides.

address selection

DNA Session Control function that provides transparent selection of protocols and addresses for transport connection establishment based upon the destination object name.

address tower

See tower.

addressing

Function that ensures that different communicating systems are identified correctly at all times. For example, in the Transport layer, provides a unique address to every transport service access point (TSAP). *See also DECnet-Plus addressing.*

addressing authority

Authority responsible for the unique assignment of Network layer addresses within an addressing domain. Example: the American National Standards Institute (ANSI).

addressing domain

Level in the hierarchy of Network layer addresses. Every NSAP address is part of an addressing domain administered directly by one addressing authority.

adjacency

Single connection to an adjacent node. Collection of state information representing a node in the local node's routing databases.

adjacency address

Address that identifies a local subnetwork access point and a subnetwork address of an adjacent system.

adjacent nodes

Nodes with direct lines between them; can communicate without an intermediate system. For example, all nodes on an Ethernet LAN are adjacent to each other.

adjustment

DECdts process of changing the system clock time by modifying the incremental value that is added to the clock's software register for a specified duration.

ADMD

See Administration Management Domain.

Administration Management Domain (ADMD)

X.400 Message Handling System public service carrier, for example, MCI mail and ATT mail in the U.S. and British Telecom Gold400mail in the U.K. The ADMDs in all countries worldwide together provide the X.400 backbone. *See also Private Management Domain.*

administrative domain

Collection of end systems, intermediate systems, and subnetworks operated by a single organization or administrative authority; can be subdivided into a number of routing domains.

advertisement

Information sent in the multicasts of DECDns servers to make their existence known; includes the server's address and attributes. This information is used by other servers and DECDns clerks in a local area network. The clerks cache the addresses.

AE qualifier (application-entity qualifier)

Identifier of a specific application entity within an application process.

AE title (application-entity title)

Unique name of an application entity. Consists of an application-process title and an application entity qualifier.

AFI

See **authority and format identifier**.

aged packet

Data packet that is discarded because it exceeded the maximum number of visits while being forwarded through the network.

agent

Defined as:

- Client/server model — Part of the system that performs information preparation and exchange on behalf of a client or server application.
- PSDN — Initiator of a call.
- Network management — Portion of an entity that is responsible for monitoring and controlling the entity's service element; accepts directives (management requests) from a director through the entity's management interface, and then performs the specified operations.

agent access point

Instance of a connection between a director or a parent entity and an agent.

aggregate throughput

See **throughput**.

algorithm

See **link state algorithm** and **routing vector algorithm**.

alias

FTAM and Virtual Terminal: Application name that corresponds to an address and, optionally, other information required to locate a remote system and its FTAM implementation.

alias node identifier

See **cluster alias**.

American National Standards Institute (ANSI)

U.S. standardization body that defines U.S. standards for the information processing industry; ANSI participates in defining network protocol standards.

AOW

See **Asia and Oceania Workshop**.

AP-title (application-process title)

Component of an application-entity title. Identifier of an application process.

API

See **application programming interface**.

application

Process that performs a specific network function, such as remote file operations and mail.

application context

Explicitly identified set of one or more application service elements (ASEs), for example, FTAM, related options, and any other necessary information or rules for an association.

application entity

Set of resources, for example, programs and process slots, that perform a communication function. Entity that invokes one or more protocol machines of the Application layer within an application process.

application-entity qualifier

See **AE-qualifier**.

application-entity title

See **AE-title**.

Application layer

Top layer (Layer 7) in the OSI Reference Model; provides for distributed processing and access; contains the applications and supporting protocols that use the lower layers. Has no formal upper boundary because it includes application programs, for example, electronic mail and file transfer, and their individual user interfaces. All OSI application programs are part of this layer.

application process

Defined as:

- Component carrying out a particular function on a computer system.
- Element within a real open system that processes the information for a particular application.
- Process within the Application layer that has evolved from a user's application program and uses any of the upper-layer services.

application-process title

See **AP-title**.

application programming interface (API)

Set of calling conventions defining how an application invokes a service through a software package.

application service element (ASE)

Application layer element that supplies a specific service to an application process.

architecture

Model that defines the relative positions of specific functions and the types of relationships that these functions can make. A layered architecture defines the grouping of functions within a layer and provides the basis for defining formal protocols. For example, the OSI Reference Model is a layered model, based on the concept of functional layers. *See also* **Digital Network Architecture** and **ISO Reference Model for Open Systems Interconnection**.

area

Group of systems, within a routing domain, that make up a single level 1 routing subdomain. These systems can run independently as a subnetwork.

area address

Address prefix, which consists of the initial domain part (IDP) through to and including the LOC-AREA field of an NSAP. A DECnet-Plus system can have more than one area address, but the systems making up an area must have at least one area address in common with each of their neighbors.

area router

See **level 2 router**, **DECnet-Plus level 2 router**, and **DECnet Phase IV level 2 router**.

area routing

Forwarding of data packets from one network area to another by the intermediate systems in the level 2 network.

ASE

See **application service element**.

ASN.1 (Abstract Syntax Notation One)

Widely-used abstract syntax notation that uses standard Backus-Naur Form notation to describe application syntaxes. Defined in ISO 8224. *See also* **basic encoding rules**.

associating with a TSAP

Assigning a *tsap-id* as a transport address for a given transport user, so it can receive a connection request.

association

Connection between two application entity invocations. Communications channel for data transfer that is established at the Application layer between two peer service users.

Association Control Service Element (ACSE)

Application service element (ASE) that provides association setup and release services. Element of the Application layer that manages associations for applications, such as FTAM, using a presentation connection. Negotiates and establishes all the services of the upper three layers.

asynchronous balance mode (ABM)

See **link access protocol balanced**.

asynchronous connection

Connection between two DECnet-Plus systems with a low-cost, low-speed asynchronous line; a full-duplex connection that can be used for remote asynchronous communications over a telephone line using a modem. There are two types of asynchronous DECnet-Plus connections—static asynchronous connections (See also **permanent virtual circuit**) and dynamic asynchronous connections (See also **switched virtual circuit**).

asynchronous response mode (ARM)

See **link access protocol**.

asynchronous transmission

Data transmission in which the time intervals between character transmissions differ. Each character is surrounded by a start bit and stop bits to allow the receiving device to recognize the beginning and end of each character (also called **start-stop transmission**). Most common over terminal lines and lines connecting other inexpensive devices, such as personal computers. *Contrast with synchronous transmission.*

attribute

Defined as:

- Network management — Piece of information, or characteristic, that describes part of an entity; also called **characteristic attribute**. Has a value that is generally contained within the entity, and can usually be examined, and possibly modified. Categories of attributes are: status, identifier, characteristic, and counter.
Controllable or observable part of an entity; variable that network managers and applications programmers can manipulate for optimal performance.
- FTAM — Information that states a property of something as one of a set of values, each of which has a defined meaning. See also **file attribute** and **activity attribute**.
- DECdns — Piece of information associated with a DECdns name or entity. DECdns creates some attributes for its own internal use. Other attributes are created and used by client applications interacting with DECdns. Some attributes consist of a set of values, such as the access control set (ACS), and others are single-valued. An object entry's attributes can describe its class, its network address, and its ACS, among other things. DECdns attributes can be one of four types: characteristics, counters, identifiers, or status attributes.

attribute group

Defined as:

- Network management — Architecturally defined, named collection of attributes for an entity class, grouped together, such as all information relating to errors. Identifiers of an entity that uniquely define that entity within its class. *Also called **entity identifier**.*
- DECdns — Particular category of DECdns attribute. DECdns attributes can be one of four groups: characteristics, counters, identifiers, or status attributes.

attribute set

Defined as:

- Network management — Value of an attribute that actually consists of a set of values.
- DECdns — Set of attributes maintained by a particular DECdns entity.

*Contrast with **single-valued attribute** and **set-valued attribute**.*

audit trail

Recording of all of the directives issued to an entity.

authentication

Security check to ensure that the requested action can be performed on a remote system by matching valid user name and password combinations; used by FTAM software and other applications.

authority and format identifier (AFI)

Part of the initial domain part (IDP) of an NSAP address that indicates the addressing authority responsible for the assignment of the IDP and its format, as well as the format (binary or decimal) of the domain-specific part (DSP).

availability

Proportion of time a specific piece of equipment, system, or network is usable, compared to the total time it is expected to be.

backbone

Name sometimes given to the level 2 routers that tie together all the areas of a network.

background skulk time

In DECdns, automatic timer that guarantees a maximum lapse of time between skulks of a directory, regardless of other factors such as DECdns namespace management activities and user-initiated skulks.

balanced mode

HDLC operational mode used over full-duplex links. *Contrast with **normal mode**.*

bandwidth

Range between the highest and lowest frequencies at which signals can pass through a receiver without the signal being distorted beyond recognition; reflects the medium's information-carrying capability. Usually expressed in terms of its signaling rate in Hertz (Hz) or bits per second.

baseband technology

Network technology such as Ethernet that uses one carrier frequency and requires all network stations to participate in every transmission. *See also* **broadband technology**.

basic encoding rules (BER)

Set of rules for encoding any language-specific data type (defined in ASN.1) into transfer syntax and for decoding from transfer syntax back into the language-specific data type; uses the style of encoding known as tag-length-value encoding. Sometimes incorrectly lumped under the term **ASN.1**, which properly refers only to the abstract syntax description language, not the encoding technique.

basic encoding rules of a single ASN.1 type

Transfer syntax that is obtained by applying the basic encoding rules of the ISO standard abstract syntax notation (ASN.1) defined by ISO 8825.

BER

See **basic encoding rules**.

bilateral closed user group (BCUG)

Closed user group consisting of two DTEs.

binary timestamp

Opaque, 128-bit (16-octet) binary sequence that represents a DECdts time value.

block

Contiguous unit of user information that is grouped together for transmission, such as the user data within a packet, excluding the protocol overhead.

bottleneck

Point in the network where traffic is delayed or blocked. Bottlenecks are the limiting factors in network performance.

bridge

Device that expands the extent of a LAN by connecting it to another LAN or physical link. Data Link layer relay for interconnecting LANs, used to increase the maximum number of stations, maximum distance, and total available bandwidth. Only data destined for different LANs passes through the bridge; thus a bridge improves performance by reducing traffic between LANs. *See also* **repeater** and **router**.

broadband technology

Network technology that multiplexes multiple, independent network carriers onto a single cable, accomplished with frequency-division multiplexing. Allows several networks to coexist on one cable; traffic from one network does not interfere with traffic from another because each uses a different frequency. *See also* **baseband technology**.

broadcast

Sending a single message that will be received by all nodes in a subnetwork for example, the Ethernet. *See also* **multicast**.

broadcast addressing

Type of multicast addressing in which all nodes receive the same message.

broadcast circuit

Circuit on which multiple nodes are connected. A message can be transmitted to multiple receivers, and all nodes are adjacent. Examples: Ethernet, ISO 8802-3, CSMA-CD.

broadcast end node adjacency (BEA)

End node connected to the same broadcast circuit as the local node. *See also* **adjacency**.

broadcast router adjacency (BRA)

Intermediate system (router) connected to the same broadcast circuit as the local node. *See also* **adjacency**.

broadcast subnetwork

CSMA-CD LAN functioning as a subnetwork. *See also* **subnetwork**.

buffer

Device or an area of memory used for temporary storage when transmitting data to compensate for a difference in rate of data flow or in time of occurrence of events between sender and receiver; used on intermediate systems to temporarily store data that is to be forwarded.

buffering level

Number of buffers provided at one time by the network software to hold data.

Single buffering tends to be less efficient than multibuffering but uses less memory on the local system. Multibuffering provides better performance if sufficient memory is available. With multibuffering, a network can send or process several buffers of data in quick succession.

bus

Defined as:

- LAN topology in which all nodes connect to a single transmission medium. All nodes are equal, and all nodes hear all transmissions on the medium. Bus topologies are reliable because failure of a node does not affect the ability of other nodes to transmit and receive.

- Flat, flexible cable consisting of many transmission lines or wires used to interconnect computer system components to provide communication paths for addresses, data, and control information.

cache

Process of temporarily storing new information so it will be quickly accessible for future use; used to minimize physical transfer of data between mass storage devices and memory; also is very fast memory used in combination with slower, large-capacity memories.

call

To make, or attempt to make, a connection with a remote DTE across a virtual circuit by sending call request packets.

call accept

Packet returned to the local DTE by a remote DTE that has received an incoming call packet and agrees to accept the requested virtual circuit connection.

call reference

Unique value used locally to identify a Modem Connect or X.21 call.

call request

Packet sent by a DTE to initiate setting up a switched virtual circuit with another DTE; results in the remote DCE sending the remote DTE an incoming call packet.

call sharing

Form of switched line sharing in which many clients have access to the same call on that line. *See also* **line sharing**.

carrier sense

Signal provided by the Physical layer to indicate that one or more nodes are transmitting on the Ethernet channel.

Carrier Sense, Multiple Access

See CSMA.

Carrier Sense, Multiple Access with Collision Detect (CSMA-CD) protocol

See CSMA-CD.

CCITT (International Telegraph and Telephone Consultative Committee)

Technical committee of the International Telecommunications Union (ITU) of the United Nations; also the international standards body for the international authorities that provide telecommunications, including the Post, Telephone and Telegraph Authorities (PTTs). Produces technical standards, known as Recommendations, for all internationally controlled aspects of analog and digital communications. *See also* **X recommendations**.

CCR (Commitment, Concurrency, and Recovery)

OSI application service element (ASE) used to create atomic operations across distributed systems; used primarily to implement two-phase commit for transactions and nonstop operations.

CEN/CENELEC

Union of the European Committee for Standardization (CEN) and the European Committee for Electrotechnical Standardization (CENELEC); makes up the Joint European Standards Institution; primary European standards body.

centralized management

Form of network management in which management is performed from a single point in the network.

channel

Defined as:

- Routing — Data path between two or more stations, including the communications control capability of the associated stations.
- X.25 — Logical path between a DTE and a DCE over which data is transmitted. Each channel is identified by a unique reference number, called a **logical channel number** (LCN).

characteristic

Attribute of a manageable entity that helps specify how it will function, for example, router type level 1 or level 2. As specified by the Enterprise Management Architecture (EMA), an entity has the following types of attributes: characteristics, status, and counters.

Characteristics are the tunable attributes of an entity. If modifiable, can be modified only through the direct result of a manager's directive.

character mode DTE

DTE that is unable to handle data in packet form; must interface through a packet assembler/disassembler (PAD) to connect to a PSDN; also known as a **Remote X.29 Terminal**.

checksum

Parameter that is carried in a block of data, and whose value can be used to determine whether the data was corrupted during transmission.

child directory

DECdns directory that has one or more levels of directories above it in a DECdns namespace. A directory is the child of a directory immediately above it. Any directory other than the root is a child of either the root or of some other directory in the namespace.

child entity

Lower class of entity that receives directives forwarded from its parent entity as defined by the management access relationship; is named by appending its subclass name and its entity identifier to the parent's entity name.

child pointer

Pointer that connects a DECdns directory to a directory immediately below it in a DECdns namespace; automatically created by DECdns when a manager creates a new directory. DECdns stores the child pointer in the directory that is the parent of the new directory.

circuit

Logical (virtual) link that provides a communications connection between adjacent nodes. Examples: point-to-point circuit, broadcast circuit, dynamically established link.

circuit switching

Technique that dynamically establishes a physical connection before information exchange and releases the connection following the exchange.

class name

Name of an entity class. For example, node is the global entity class.

class of protocol

Type of transport protocol. OSI Transport supports three transport protocol classes: class 0, class 2, and class 4. Each class defines a transport service with specific characteristics.

class-specific attribute

DECdns — Attribute that has meaning only to a particular class of DECdns object and to the application using that object class. A DECdns object's class is defined in the `DNS$Class` attribute.

clear

Stop a connection across a virtual circuit to a remote DTE by sending clear request packets.

clear confirmation

Packet sent under the following circumstances:

- To the local DCE by a DTE that has received a clear indication packet (a DTE clear confirmation packet).
- To a DTE that has requested clearing when a DCE receives a DTE clear confirmation packet through the network (a DCE clear confirmation packet).

clear indication

Packet sent by a DCE to a DTE when it is clearing a virtual circuit.

clearinghouse

Collection of directory replicas on one DECdns server. A clearinghouse takes the form of a database file. It can exist only on a DECdns server. Usually only one clearinghouse exists on a server, but there may be special cases when more than one exists.

clearinghouse object entry

Special class of object entry that describes a clearinghouse; pointer to the node address of a clearinghouse, which enables DECdns to find a clearinghouse and use and manage its contents; has the same name as the clearinghouse. A clearinghouse modifies and manages its own object entry when necessary; normally DECdns managers do not need to maintain it.

clear request

Packet sent by a DTE to initiate the clearing of a switched virtual circuit; can also be sent by a destination DTE that has received an Incoming Call packet and is unable to accept the requested virtual circuit connection.

clerk

Defined as:

- DECdns — Software that provides an interface between client applications and DECdns servers. The clerk receives a request from an application, sends the request to a DECdns server, and returns any resulting information to the application.
- DECdts — Software that synchronizes the clock for its client system by requesting time values from DECdts servers, computing a new time from the values, and supplying the computed time to client applications, such as the operating system.

client

Defined as:

- X.25 — Access system.
- Distributed processing — *See* **client/server model**.

client application

Defined as:

- DECdns — Any application that interacts with a DECdns server through the DECdns clerk.
- DECdts — Any application that interacts with a DECdts server through the DECdts clerk.

client entity

Entity that constitutes the client side of a client/server relationship.

client/server model

Model of interaction used in Digital's distributed processing products. A program at one system sends a request to a program at another system and awaits a response; the requesting system is the client; the system satisfying the request is the server.

CLNP

See **Connectionless Network Protocol**.

CLNS

See **Connectionless-Mode Network Service**.

CLNS with Internet/ES-IS

Connectionless network service used for transport connections in which the two end systems are on different subnetworks; provided by the implementation of the ES-IS and Internet routing protocols. The intervening subnetworks can be a mixture of technologies.

CLNS with null Internet

Connectionless network service used for transport connections in which both end systems are on the same 802.3 LAN. This network service is provided by the inactive subset of the Internet routing protocol.

clock

Combined hardware interrupt timer and software register that maintain system time. In many systems, the hardware timer sends interrupts to the operating system; at each interrupt, the operating system adds an increment to a software register which contains the time value.

closed user group (CUG)

Group of DTEs restricted to communicating with each other.

CLTP

See **Connectionless Transport Protocol**.

cluster alias

Optional node name and address used by some or all nodes in an OpenVMS cluster, allowing any of these nodes to be reachable on the network with the same address.

CMIP

See **Common Management Information Protocol**.

collision

Condition in which two packets are transmitted over a medium at the same time, making both unintelligible.

collision detect

Signal provided by the Physical layer to the Data Link layer to indicate that one or more nodes are contending with the local node's transmission. Stations detect the collision and retry the transmission.

Comite Consultatif Internationale Telegraphique et Telephonique (CCITT)

See **CCITT**.

Commitment, Concurrency, and Recovery

See **CCR**.

common carrier

Organization in the United States that offers standard and consistent communications services within a country. *See* **Postal, Telegraph and Telephone Authority**.

Common Management Information Protocol (CMIP)

ISO protocol that describes the exchange of network management information.

Common Trace Facility (CTF)

DECnet-Plus for OpenVMS problem-solving facility for looking at and analyzing data generated by events on a network.

communications buffers

Buffers allocated at each node for temporary storage of communications data.

communications link

Physical medium connecting two systems.

communications server

Special-purpose standalone system dedicated to managing communications activities for other computer systems.

computed time

Result of the synchronization process used to adjust the system clock time; the time interval that the DECdts clerk or DECdts server computes according to the intersections of the DECdts server time values it receives.

concatenation

Process of joining two or more items together, as when input files are appended to a new output file.

conceptual communications area

Data store used in virtual terminal associations; contains various data structures that model a terminal's behavior. The systems in a virtual terminal association maintain a separate copy of the conceptual communications area and share information about changes to these data structures.

confirm primitive

Primitive generated by a service provider to pass an incoming response to a service user.

confirmed event report

Report in which the event sink responds to a request after the event record has been stored in the event log.

confirmed service

Service whose requested parameter values are acknowledged and often negotiated by a two-way exchange of service primitives.

congestion

Condition in which a network or part of a network is overloaded and has insufficient communication resources for the volume of traffic.

congestion avoidance

DECnet-Plus routing mechanism that adjusts the load on the network to prevent congestion.

congestion loss

Condition in which data packets are lost when routing is unable to buffer them.

connection

Logical link between two open systems.

connection control

DNA session control function that controls system-dependent functions that create, maintain, and end transport connections.

Connectionless-Mode Network Service (CLNS)

Connectionless network service provided by OSI transport; operates according to a datagram model. Each message is routed and delivered to its destination independently of others. For example, the DNA Network (Routing) layer provides this type of service.

See also **CLNS with Internet/ES-IS** and **CLNS with null Internet**.

Connectionless Network Protocol (CLNP)

OSI protocol for providing CLNS (datagram service). A "gateway" protocol that creates an Internet header, allows dynamic routing, and is critical for X.25 communications.

OSI equivalent to the Internet Protocol (IP), sometimes called ISO IP.

connectionless service

Service in which no permanent connection is set up between the two communicating ends. Instead, each service data unit contains the addressing information that identifies its destination. Examples: LANs, Internet IP, OSI CLNP, and UDP.

Connectionless Transport Protocol (CLTP)

OSI protocol that describes end-to-end transport data addressing (via transport selector) and error control (via checksum), with no guarantee of delivery and no flow control.

Connection-Oriented Network Service (CONS)

Connection-orientated network service provided to OSI transport; operates according to a connection-oriented model. A connection is set up between two communicating end users, is used for data exchange, and is then broken by either end. Service data units sent over the connection do not have to contain a destination address. X.25, for example, provides this type of service.

connection-oriented service

Service in which communication proceeds through three well-defined phases: connection establishment, data transfer, connection release. A logical connection, such as virtual circuits, is established between end points. Examples: X.25 and OSI TP4.

connection request

Request for a connection to a peer entity or user.

connectivity

Degree to which network nodes are interconnected. Full connectivity means all nodes have links to every other node.

connector system

DECnet-Plus system that connects to PSDNs and also provides indirect connection to PSDNs for other systems connected to this system.

CONS

See **Connection-Oriented Network Service**.

console carrier

Maintenance Operations Protocol (MOP) function that provides access to the remote console subsystem of a network server on a LAN.

constraint set

Set of related statements limiting the range of structures allowed for a given type of file, and specifying how file-access actions can modify those structures without changing their essential nature.

contention control

Scheme of access control used by many networks. Control is distributed among the nodes of the network. Any node wanting to transmit can do so, accessing the network on a first-come, first-served basis. However, it is possible that two nodes are in contention, or start transmitting at the same time, in which case a collision occurs. Each node must then back off and retransmit after waiting a random period of time.

contents type

FTAM file attribute that defines file structure and contents, can be expressed as an abstract syntax/constraint set pair or as a document type. DECnet-Plus for OpenVMS and DECnet-Plus for Digital UNIX FTAM use only the document-type form of contents type.

control access

DECdns access right that permits users to change the access control on a name and do other powerful management tasks, such as replicate a directory or move a clearinghouse.

control object

Data structure in the conceptual communications area that models a terminal's control sequences, such as the control sequence that rings a terminal's bell.

control station

"Master node" at the end of a multipoint circuit. The control station controls the tributaries for that circuit; responsible for data link control.

controller

See **line controller**.

convergence

Degree to which DECdns attempts to keep all replicas of a directory consistent. *See also* **high convergence** and **low convergence**.

Cooperation for Open Systems Interconnection Networking in Europe (COSINE)

Program sponsored by the European Commission, aimed at using OSI to tie together European research networks.

Coordinated Universal Time (UTC)

International time standard maintained by the International Time Bureau.

Corporation for Open Systems (COS)

Corporation founded by U.S. computer vendors to provide a central clearinghouse for cross testing, conformance testing, certification, and promotion of their OSI products.

cost

Integer value assigned to a circuit between two adjacent nodes. The routing decision algorithm uses this value and selects paths with the lowest cost. Nodes on either end of a circuit can assign different costs to the same circuit.

cost path

Cost of each path between a source node and a destination node; sum of the costs assigned to the circuits that compose the path. DECnet-Plus routing forwards packets on the lowest-cost path even if that one does not have the fewest hops.

counters

Performance and error statistics kept for an entity by network management, such as lines and nodes.

courier

Local DECdts server that requests a time value from a randomly selected global server each time it synchronizes.

creation timestamp (CTS)

Attribute (named DNS\$CTS) of all clearinghouses, directories, soft links, child pointers, and object entries that contains a unique value reflecting the date and time that the name was created; actually consists of two parts: a time portion and a portion with the system identifier of the node on which the name was created. This guarantees uniqueness among timestamps generated on different nodes.

credit window

See **maximum window**.

CSMA (Carrier Sense, Multiple Access)

Medium-access control technique for multiple-access transmission media. A station that wants to transmit, first senses the medium and then transmits only if it is idle.

CSMA-CD (Carrier Sense, Multiple Access with Collision Detect Protocol)

Data link protocol used by Ethernet and ISO 8802-3 LANs; allows multiple stations to access the broadcast channel at will, avoids contention by means of carrier sense and deference, and resolves contention by means of collision detection and retransmission. Each station awaits an idle channel before transmitting and can detect overlapping transmissions by other stations.

CTF

See **Common Trace Facility**.

DA circuit

See **dynamically assigned circuit**.

DAP

See **Data Access Protocol**.

DAP-FTAM Gateway

DECnet-Plus for OpenVMS software that allows a DECnet-Plus for OpenVMS node to participate in an OSI network in a simple way; functions as a server that receives DAP messages through DECnet and uses that information to establish and maintain a connection with a remote FTAM system. The DECnet-Plus for OpenVMS user issues DECnet commands to perform remote file operations. *See also* **FTAM-FTP Gateway**.

Data Access Protocol (DAP)

Protocol for DECnet file transfer.

data chain buffer (DCB)

Descriptor that describes some or all of a protocol data unit (PDU). DCBs can be chained together to describe the entire PDU.

data circuit-terminating equipment (DCE)

Network switching equipment that provides the functions needed to establish, maintain, and terminate a connection and handle the signal conversion and coding between the DTEs that use either a physical circuit or a virtual circuit. The network switching exchange to which DTEs are connected. *See also* **data terminal equipment**.

data link

Logical connection between two systems on the same circuit on which data integrity is maintained.

Data Link layer

Layer 2 in the OSI Reference Model; specifies the technique for moving data along network links between defined points on the network and for detecting and correcting errors in the Physical layer.

data link mapping (DLM)

See **dynamically assigned circuit**.

Data Link Protocol

OSI rules and procedures that allow the data link service to be provided.

data link protocol data unit (DPDU)

Packet from a data link protocol.

data network identification code (DNIC)

In X.25, usually the first four digits of the DTE address that identifies the country, and distinguishes the PSDN from other PSDNs within the same country. Some networks require a leading zero to denote that a DNIC is being used.

data octet

Another term for **8 bits**.

data overrun

Data blocks received that arrived too quickly to be processed by the receiver and were, therefore, lost.

data packet

See **packet**.

data segment

For most PSDNs, 64 bytes.

data terminal equipment (DTE)

User's equipment (computer or terminal) connected to a modem (DCE) on a packet switching network; both sends and receives data. *See also* **data circuit-terminating equipment**.

data transfer facility

Facility for exchanging normal data over an OSI transport connection. The transfer occurs as a sequence of TPDU's from a sending TSAP to a receiving TSAP.

data-transfer regime

Regime that controls bulk data transfer; moves file data between FTAM peer entities.

datagram

Defined as:

- X.25 — Self-contained packet, independent of other packets, that does not require acknowledgment, and that carries information sufficient for routing from the originating DTE without relying on earlier exchanges between the DTEs and the network.
- OSI — Unit of data sent over the network that is handled independently of all other units of data. The unit of data (NSDU in ISO terminology) is passed between the Network layer and the Data Link layer. When a route header is added, the datagram becomes a packet (network PDU in ISO terminology). Datagrams may fail to be delivered with no notice sent to the sender.

DCE

See **data circuit-terminating equipment**.

DDCMP

Byte-oriented, Data Link layer, DNA protocol implemented in DECnet software; designed to provide an error-free communications path between adjacent systems. Operates over serial lines, delimits frames by a special character, and includes checksums at the link level.

DEC WANrouter system

DECnet-Plus intermediate system. Can run either the DECnet-Plus routing algorithm (link state) or the DECnet Phase IV routing algorithm (routing vector). Most DEC WANrouter products are multiprotocol routers. *See also* **DECnet-Plus level 1 router** and **DECnet-Plus level 2 router**.

DECdns (Digital Distributed Name Service)

DECnet-Plus software that provides network applications, especially DECnet-Plus, with a means of assigning unique names to network resources so a network application can find resources within the network.

DECdns clerk

Interface between client applications and DECdns servers; DECnet-Plus system running DECdns clerk software.

DECdns client

Application that interacts with a DECdns server through the DECdns clerk interface.

DECdns control program (DNSCP)

Command interface that DECdns managers use to control DECdns servers and clerks, and to manage the namespace and its contents.

DECdns entity

DECdns server, clearinghouse, or clerk software on a system.

DECdns namespace

See **namespace**.

DECdns server

Node running DECdns server software; handles name lookup requests and maintains the contents of the clearinghouse at its node.

DECdns-defined attribute

Standard attribute that DECdns associates with names in the namespace.

DECdts (Digital Distributed Time Service)

DECnet-Plus software that synchronizes the clocks in networked systems.

DECdts clerk

DECdts process that synchronizes the clock for its client system by requesting time values from servers, computing a new time from the values, and supplying the computed time to client applications, such as the operating system; DECnet-Plus system running DECdts clerk software.

DECdts entity

DECdts server or clerk software on a system.

DECdts server

Node running DECdts server software; DECdts entity that synchronizes with its peers and provides its clock value to clerks and their client applications.

decision

Routing process that determines the path, or route, along which a data packet travels to reach its destination; forwards packets on the lowest-cost path even if that one does not have the fewest hops. The path that the data takes through the network is transparent to users.

DECnet

Family of Digital proprietary hardware and software communications products that implement the Digital Network Architecture (DNA). These products allow all Digital systems to communicate with each other. *See also DECnet-Plus and DECnet Phase IV.*

DECnet/OSI for Digital UNIX

DECnet software implementation for Digital UNIX systems.

DECnet Phase IV

Family of Digital proprietary hardware and software communications products that implement the Digital Network Architecture (DNA) Phase IV.

DECnet Phase IV area

Area in which all level 1 routers are running the DECnet Phase IV routing vector protocol.

DECnet Phase IV end node

End node that receives and transmits data packets but does not route packets through; can receive and transmit only Phase IV-format-packets and can communicate with DECnet-Plus nodes only if their addresses are compatible with Phase IV.

DECnet Phase IV level 1 router

DECnet Phase IV router that receives and transmits data packets, forwards packets to other nodes within its same DECnet area, and forwards packets to the nearest level 2 router for out-of-area destinations. Can receive and transmit only Phase IV format packets and can communicate with DECnet-Plus nodes only if their addresses are compatible with Phase IV. Runs the DECnet Phase IV routing vector algorithm.

DECnet Phase IV level 2 router

DECnet Phase IV router that receives and transmits data packets, forwards packets to other nodes within its same DECnet area, and forwards packets for out-of-area destinations through other level 2 routers to the destination area. Can receive and transmit only Phase IV-format-packets and can communicate with DECnet-Plus nodes only if their addresses are compatible with Phase IV. Runs the DECnet Phase IV routing vector algorithm.

DECnet-Plus

Family of Digital hardware and software products that implements the Digital Network Architecture (DNA) Phase V, which integrates OSI and DNA protocols; compliant with OSI and compatible with DECnet Phase IV and TCP/IP.

DECnet-Plus addressing

DECnet-Plus addressing scheme that complies with the ISO 8348 Addendum 2 addressing standard; uses the concepts of global network addressing, addressing authorities, and addressing domains. ISO scheme designed to provide unique network addresses throughout the world.

DECnet-Plus area

Area in which all level 1 routers are running the DECnet-Plus link state routing protocol.

DECnet-Plus end system

DECnet-Plus system that receives and transmits data packets but cannot route packets through; can receive and transmit both Phase IV format and DECnet-Plus data packets; communicates with Phase IV nodes using Network Services Protocol (NSP) and with DECnet-Plus systems using either NSP or the OSI Transport service.

DECnet-Plus for OpenVMS

DECnet-Plus software implementation for OpenVMS systems.

DECnet-Plus level 1 router

DECnet-Plus multiprotocol router (intermediate system) that receives and transmits data packets, forwards packets to other nodes within its same area, and forwards packets to the nearest level 2 router for out-of-area destinations; can receive and transmit both Phase IV format and DECnet-Plus format data packets.

Can route OSI data and Internet Protocol (IP) packets over the same lines, conforming to the Integrated IS-IS Protocol specified in RFC 1195, "Use of OSI IS-IS for Routing in TCP/IP and Dual Environments."

When configured for multiprotocol support, can perform as an IP router in an IP network; transmits IP packets directly over the data link and does not encapsulate these packets within DECnet or OSI packets.

DECnet-Plus level 2 router

DECnet-Plus multiprotocol router (intermediate system) that receives and transmits data packets, forwards packets to other nodes within its same area, and forwards packets for out-of-area destinations through other level 2 routers to the destination area; can receive and transmit both Phase IV format and DECnet-Plus format data packets.

Can route OSI data and IP packets over the same lines, conforming to the Integrated IS-IS Protocol specified in RFC 1195, "Use of OSI IS-IS for Routing in TCP/IP and Dual Environments."

When configured for multiprotocol support, can perform as an IP router in an IP network; transmits IP packets directly over the data link and does not encapsulate these packets within DECnet or OSI packets.

decoding

Process by which the transfer syntax representation of a data value is transformed into the local representation of that value.

DED

See **dynamically established data link**.

dedicated line

Line for the exclusive use of a leasing customer without interchange switching arrangements. Same as **leased line** and **private line**.

delete access

Access right that grants users the ability to remove DECdns data from the namespace.

designated router

Router that solves the negative effects of excess routing information on a DECnet-Plus LAN configured with two or more routers. Creates a link state packet (LSP) that describes the entire LAN to the rest of the network and selects the router that will represent the pseudonode. (On an Ethernet or ISO 8802-3 LAN, the LAN broadcast link is considered to be a pseudonode with links to each attached system. Attached systems report their link only to this pseudonode. The designated router, on behalf of the pseudonode, constructs an LSP that lists the links to all the systems attached to the LAN at zero cost.)

destination address

Address that identifies a target system; can be an X.25, a physical, or an NSAP address.

destination service access point (DSAP)

One-byte field in a logical link control (LLC) frame on a LAN that identifies the link service access point of the receiving Data Link layer client protocol.

DEUNA

Network adapter for connecting UNIBUS-based hosts to a CSMA-CD local area network.

device driver

Software associated with each physical device in the system that serves as the interface between the operating system and the device controller.

device layer

Collection of director device plug-ins that handle the details of the I/O devices that interact with the manager.

device object

Data structure in the conceptual communications area that models characteristics of real devices and helps to map display objects.

dialogue

Sequence of message exchanges between open systems that represents a single association and the set of underlying connections.

dial-up line

Telephone communications circuit established by a switched circuit connection.

Digital Distributed Name Service (DECdns)

See DECdns.

Digital Distributed Time Service (DECdts)

See DECdts.

Digital Network Architecture (DNA)

Set of protocols governing the format, control, and sequencing of message exchange for DECnet and DECnet-Plus implementations; defines rules for all the layers of data exchange, from the lowest level of the physical medium of transmission to the top, user interface level; also defines standard network management and network generation procedures.

DNA Phase IV comprises Digital proprietary networking protocols. DNA Phase V integrates Digital proprietary networking protocols and OSI open standards and protocols. *See also* **architecture**.

directive

Management operation initiated by the director (the requester, which can be a human user) and sent to the entity (the responder) that performs the function.

director

Software management system that interacts with a user, initiates management operations on behalf of the user, coordinates management activities with entities, and provides high-level management applications.

directory

Logical unit for storing object entries under one name (the directory name) in a DECdns namespace; can also contain soft links and child pointers. Each physical instance of a directory is called a **replica**.

disconnect abort

Method by which nontransparent tasks can de-access a logical link by means of a disconnect abort operation without deassigning the channel. This form of disconnection indicates to the receiver that not all messages sent have necessarily been received.

display object

Data structure in the conceptual communications area that models a terminal's display and keyboard operations.

distance vector routing

See **routing vector algorithm**.

distributed management

Form of network management in which network managers and directors are dispersed across many systems.

distributed processing

Technology that enables the distribution throughout the network of computing power and storage facilities to user work areas, such as offices, laboratories, or desks on factory floors.

distributed system

Collection of computer systems, tied together by communications networks for the purpose of sharing resources; end users do not need to be aware of the physical location of the shared resources.

distributed system management model

Framework within which DECnet-Plus network management is designed.

DLM

See data link mapping.

dlogin

Feature of both Berkeley UNIX (called `rlogin`) and DECnet/OSI for Digital UNIX software that allows users of one system to log in to other systems. (Similar to Telnet.) Comparable to DECnet-Plus for OpenVMS software's `set host` and to the DECnet-Plus Virtual Terminal (VT) software.

DNA

See Digital Network Architecture.

DNIC

See data network identification code.

DPDU

See data link protocol data unit.

docket

Information associated with an OpenVMS file service regime that must be preserved to provide FTAM error recovery.

document type

As defined by the FTAM protocol, a set of related statements about a file. Some of these statements relate to the intended use of the file (for example, by specifying a constraint set and a file model); the others describe its structure, scope, abstract syntax, and transfer syntax.

domain

See routing domain.

domain-specific part (DSP)

Part of a network service access point (NSAP) address defined by the organization that implements the networking software.

downline loading

Transferring a copy of a system image from a load host node to a target node. Some systems, such as DEC WANrouter systems and DECserver terminal servers, automatically request a downline load of their image upon startup and reboot.

drift

Change in a clock's error rate over a specified time period.

DSP

See domain-specific part.

DTE

See data terminal equipment.

DTE address

Unique address given to a DTE.

DTE Class entity

Entity that defines a named class of DTEs; used to determine which DTE or X.25 Connector node to use when making incoming and outgoing calls.

DTSS

See **DECdts** and **DTSS module**.

DTSS module

Network management module for synchronizing and managing the system clocks in a distributed DECnet-Plus network; contains all DECdts management functions.

dynamic assignment (DA)

Use of a dynamically established data link by the Network layer in such a way that a connection is made only when data is required to be transferred, and the subnetwork address to which the connection is made is determined by the destination NSAP address.

dynamic connection management (DCM)

Use of a dynamically established data link by the Network layer in such a way that a connection is made only when data is required to be transferred.

dynamically assigned (DA) circuit

Routing circuit that DECnet-Plus uses to create a path between DECnet-Plus and X.25 implementations on the same node; the way through which a DECnet-Plus node running X.25 software communicates through a PSDN to another DECnet-Plus node.

dynamically established data link (DED)

Connection in which routing dynamically determines the address of the destination based on receive traffic. Examples: X.25, X.21.

encapsulation

Technique used by some layered protocols in which a layer adds header information to the protocol data unit (PDU) from the layer above. As an example, Digital's Internet Portal product encapsulates IP datagrams to produce DECnet packets that can be transmitted by Digital routers across a DECnet WAN.

encoding

Defined as:

- Process by which a sender applies the rules of a transfer syntax to transform local data into data that another FTAM system can receive and decode.
- Process by which the local representation of a data value is transformed into the transfer syntax representation of that value.
- Actual representation of a data value in a specific transfer syntax (equivalent to "encoded form").

encoding rules

Rules that guide the encoding of local data and decoding of received data.

end node

See **end system**.

end system

Nonrouting system; can receive data units addressed to it and send data units to other systems on the same subnetwork but cannot relay, route, or forward data packets to other systems.

End System to Intermediate System (ES-IS) Protocol

OSI protocol implemented at the Network layer of an end system (ES) that enables communication between end systems on different subnetworks. Packets are routed to an intermediate system (IS) on the same subnetwork as the source end system, and from there are routed to the destination end system, possibly via one or more subnetworks. Also called a routing exchange protocol.

Enterprise Management Architecture (EMA)

Digital model for the way in which components in a distributed system are managed.

entity

Defined as:

- DECnet-Plus— Active element in a layer. Entities in a layer communicate with other entities in the same layer, but in different systems.
- Network management — Individual, manageable piece of a network; has attributes that describe it, a name that identifies it, and an interface that supports management operations.
- FTAM — Instance of one or more protocol machines of a given layer operating within a process to support a specific dialogue.

entity class

Collection of entities that share the same properties and that have the same parent entity; each member of the class has a unique identifier within the class. Entity classes have class names.

entity group

Architecturally defined collection of entities. The entities in the group must have a common top entity and must all be of the same class.

entity hierarchy

Logical hierarchical tree structures of manageable entities in which child entities are below their parent entities. Children can be accessed only through their parents' agent.

entity identifier

See **attribute group**.

entity instance block (EIB)

Logical representation of an instance of an entity to network management software; contains all values for attributes of that instance.

entity management agent access (EMAA)

Internal utility of DECnet-Plus network management.

entity name

Label associated with some entities used to identify or locate them for management purposes.

entity type

Subgrouping of an entity that determines its relationship to other DECdts components: clerk or server.

epoch

Timestamp that identifies directory replicas as being part of the same set; used by DECdns when it skulks a directory, finding all replicas of the directory that are in the same epoch, and makes their contents consistent.

epoch number

Identifier that a DECdts server appends to the time values it sends to other servers. DECdts servers use time values only from other servers with whom they share epoch numbers.

equal cost path splitting

Routing process in which packets are forwarded over multiple equal cost paths to a destination node.

error

Difference between a system's clock value and the computed time.

error recovery

Procedures for recovering from the effects of detected OSI transport errors. Protocols may provide specific facilities for doing this.

error tolerance

Amount of system clock error to which the DECdts entity responds by abruptly setting the system clock to the computed time, rather than gradually adjusting the clock.

ES-IS

See **end system to intermediate system protocol**.

Ethernet

CSMA-CD LAN, similar to the one defined by ISO 8802-3, originally defined jointly by Digital, Intel, and Xerox. Composed of passive coaxial cable or shielded twisted-pair cable with interconnections containing all active components so that no switching logic or central computer is needed to establish or control communications. A best-effort delivery system. Supports the Ethernet protocol and the IEEE 802.2 and 802.3 protocols.

Ethernet protocol

Data Link layer protocol used by Ethernet LANs.

event

Occurrence of either a normal or abnormal condition that an entity detects; network- or system-specific occurrence that can be logged. Many events are informational, for example, the arrival of a PDU. Other events report potential or current problems.

event dispatcher (EVD)

DECnet-Plus network management software that reports events that occur during network operation on a particular system, between two systems, or across distributed systems.

event filtering

Process of deciding whether to pass an event to its destination, discard it, or subject it to further filtering. Event filtering can be applied to an event both at the event source where it is generated and to the event sink to which it is sent. User-specified.

event logging

See **event dispatcher**.

event sink

Event dispatching component to which an event is delivered.

event source

Event dispatching component where an event is generated.

event stream

Connection between an event source and an event sink, over which events are delivered.

expedited data

Service that transfers small amounts of data between peer entities without the need for normal token controls.

Small amounts of data that are not subject to the flow control applied to normal data. *Contrast with* **normal data**.

extended LAN

Multiple LANs connected with data link relays or bridges.

facility

Service or mode of operation that a PSDN is able to provide for a user upon subscription and/or request, for example, fast select or reverse charging.

FADU

See **file-access data unit**.

FAL

See **file access listener**.

FDDI

See **Fiber Distributed Data Interface**.

Federal Networking Council (FNC)

Body responsible for coordinating networking needs among U.S. Federal agencies.

FERPM

See **file error recovery protocol machine**.

Fiber Distributed Data Interface (FDDI)

ANSI standard for a high-speed, general-purpose network using an optical fiber medium for interconnection of computers and peripheral equipment; specifies a 100 mb/s data rate using 1300-nanometer light wavelength and limits networks to approximately 200 km in length, with repeaters every 2 km or less. The access control mechanism uses Token Ring technology and the topology is a dual-attached, counter-rotating Token Ring.

file-access data unit (FADU)

Data structure that FTAM file-access services use when accessing file data; either a whole file or a specific subtree comprising one or more nodes and associated file-contents data units such as records.

file access listener (FAL)

Image that provides authorized access to the file system of a DECnet-Plus node on behalf of processes executing on any node in the network. FAL communicates with the initiating node by means of the Data Access Protocol (DAP).

file-access structure

Structure that links together a file's file-access data units (FADUs), allowing for their identification, description, and manipulation.

file attribute

Permanent characteristic of a file, such as its file name. The values of some file attributes may change during the lifetime of a file. However, different FTAM users simultaneously accessing a file see the same values for file attributes.

file-contents data unit

Sequence of zero or more file-contents data elements. Each file-contents data element specifies a data type in terms of a specific abstract syntax.

file data

Nonstructural information that is stored within a file and comprises its contents (as opposed to its attributes).

file designation

System-specific information that identifies a file on its storage system.

file error recovery protocol machine (FERPM)

FTAM software that implements the FTAM error recovery protocol.

file model

Model of a file-access structure. *See also* **hierarchical file model**.

file-open regime

Regime that controls access to the contents of a currently selected file.

file protocol

Communications between FTAM initiators and responders and the operations of file services during an association.

file-selection regime

Regime that associates the initiator with a file and its attributes.

file service

Set of services provided by an FTAM entity for transferring, accessing, and managing remote files.

file-service model

Description of the services, regimes, and activity attributes that support filestore actions.

file specification

System-specific information that identifies a file on its storage system.

filestore

Organized collection of files that includes their attributes, such as file names. *See also* **virtual filestore** and **real filestore**.

filestore actions

Actions performed on files within the virtual filestore, including actions on complete files and their attributes (whole-file actions) and actions for accessing file contents (file-access actions).

filestore password

FTAM parameter whose purpose is to convey a password for authenticating the initiator to the responder. The FTAM responder equates the filestore password to an ULTRIX login password (on a DECnet-Plus for Digital UNIX node) or to a OpenVMS login password (on a DECnet-Plus for OpenVMS node).

File Transfer, Access, and Management

See **FTAM protocol**.

flag sequence

Series of ones and zeros that indicate the start and end of a frame.

flat file view

Access context for a file that recognizes two levels of nodes: a root node and, optionally, first-level nodes. The root node lacks an associated data unit, but first-level nodes can have an associated file-contents data unit.

flooding

DECnet-Plus routing process that propagates Link State Packets (LSPs) as a result of network changes, for example, node failure. Each new LSP is transmitted on all the circuits of an intermediate system, except the circuit on which it was received.

flow control

Function performed by a receiving entity to limit the amount or rate of data that is sent by a transmitting entity; allows communicating layers to match their data transfer and receive rates in order to ensure that one end of the connection does not send data faster than the other end can process it; keeps traffic within limits acceptable by the end-receiver or any intermediate receiver. At the terminal level, the flow control mechanism must guarantee that the flow of characters will stop if the buffer fills up.

FNC

See **Federal Networking Council**.

forward

Routing process that transmits, or forwards, data to the destination node or to another routing node. If a packet is addressed to the local node, routing on that node delivers it to the DECnet-Plus Transport layer. If a packet is addressed to a remote node, routing forwards it to the next adjacent node on the best available path.

FPM

See **FTAM protocol machine**.

fragmentation

Routing process of breaking a large datagram into multiple smaller datagrams for transmission. The reverse process is **reassembly**. *See also* **maximum transmission unit**.

frame

Unit delimited by flags that includes a header; used by the Data Link level to exchange packets and control-and-error information between a DTE and a DCE.

frame checking sequence (FCS)

16-bit, error-check polynomial that checks that the bit content of a frame is the same before and after transmission.

frame level

X.25 protocol level that defines the procedure for transferring packets without errors between a DTE and DCE; also known as **level 2**.

frame reject frame

See **FRMR**.

FRMR (frame reject frame)

Frame used by the DTE or DCE to report an error that cannot be recovered by the retransmission of the identical frame. Used only by LAPB/LAPBE.

FTAM (File Transfer, Access, and Management)

See **FTAM protocol**.

FTAM-1

Document type for unstructured text files.

FTAM-2

Document type for sequential text files.

FTAM-3

Document type for unstructured binary files.

FTAM application name

Name that corresponds to the FTAM responder on an FTAM system. *See also alias.*

FTAM element

FTAM-specific service element of the Application layer.

FTAM entity

Entity at the Application layer (application entity) that occurs within an application process and that involves the FTAM, ACSE, and presentation protocol machines, with the rules and procedures of the FTAM protocol machine controlling the operation of the ACSE protocol machine.

FTAM file specification

Unique string of characters that a DECnet-Plus FTAM system uses to select or create a file on another FTAM system in the same network.

FTAM-FTP Gateway

DECnet/OSI for Digital UNIX software that allows DECnet/OSI for Digital UNIX users to perform file operations between OSI and Internet systems; translates both ways between FTAM Protocol and File Transfer Protocol (FTP). The DECnet/OSI for Digital UNIX user issues either FTAM or FTP commands to perform remote file operations.

FTAM protocol

File Transfer, Access, and Management protocol (ISO 8571); ISO protocol at the Application layer that defines the rules for transfer, access, and management of files between multivendor systems on the same OSI network.

FTAM protocol machine (FPM)

Software that implements the FTAM file protocol.

FTAM regime

Regime that controls the binding of two FTAM entities to an association.

FTAM standard

FTAM International Standard (ISO 8571), which defines OSI file transfer, access, and management.

FTAM system

Open system with an FTAM implementation that conforms to the OSI FTAM standard.

full-duplex circuit

Circuit designed for transmission in both directions at the same time. *Contrast with half-duplex circuit.*

full-duplex transmission

Data transmission in both directions at the same time. *Contrast with **half-duplex transmission**.*

full name

Complete specification of a name in the namespace, including all parent directories in the path from the root directory to the object, directory, or soft link being named; can also include a namespace name, but not necessary when only one namespace exists in a network.

In a full name, the reserved namespace nicknames LOCAL: and DOMAIN: indicate to DECnet-Plus that the information for the node is contained in the Local namespace (for LOCAL:.) and in DNS/BIND (for DOMAIN:.)

full support

Level of support in which a value assigned by FTAM software to an FTAM file attribute corresponds to the value of a file attribute in the real filestore on DECnet/OSI for Digital UNIX nodes (their UFS filestore) and DECnet-Plus for OpenVMS nodes (their RMS filestore).

function code

On a DECnet-Plus for OpenVMS system, parameter to a \$QIO system service call that defines the specific function to be performed by that \$QIO.

function dispatch table (FDT)

Software that identifies the operations supported by an entity and specifies the routines to be called to process them.

functional unit

Logical grouping of services.

Set of low-level communications functions, each of which involves a minimal interaction between a user and a provider of a communications service; when combined in an established order, provide a specific high-level function such as controlling regimes or transferring data. The grouping of specific functions into functional units helps an initiator negotiate the functions required during an association.

gateway

Device or software that enables multivendor communication by converting the functions of one vendor's network into functions recognizable by the other; functions as a language translator, enabling "speakers" of different languages to communicate.

Gateway Access Protocol (GAP)

Protocol used between a host DECnet-Plus system and a DECnet-Plus system that is a DTE on a PSDN to provide the X.25 gateway access facility to a user on the host.

gateway client

Access system.

gateway system

Another term for **connector system**.

GeneralString

String composed of a definable character set that includes format effectors; by default, a GeneralString is an IA5String. DECnet-Plus FTAM software uses a GeneralString as its default to convey strings of characters from the ISO 8859 character set (DEC multinational character set). The ISO 8859 character set is an 8-bit character set containing both graphic characters and control characters.

general topology subnetwork

Nonbroadcast subnetwork.

global entity

Entity that has no parent, for example, the node entity. Entity whose agent can directly receive management requests from a director. All nonglobal entities are child entities of a global entity, and receive management requests through their parent global entity. A global entity class name must be unique among the set of all global entity class names.

global name

Portion of a DECdns entity name that is registered in a DECdns clearinghouse and addresses a parent agent of the target entity.

global network address

See **network service access point**.

global network addressing

See **DECnet-Plus addressing**.

global network addressing domain

Addressing domain consisting of all NSAP addresses in the OSI environment.

global server

DECdts server that frequently provides its clock value to courier servers on other LANs, or infrequently provides its clock value to systems that have failed to obtain the specified number of servers locally.

global server directory

DECdns directory where DECdts servers are stored.

global set

All of the global DECdts servers in a network.

GOSIP

See **Government OSI Profile**.

GOSIP specification

Government OSI procurement document that specifies to computer vendors what requirements their OSI products must meet to qualify for purchase by governmental organizations. Some countries, such as the United States and the United Kingdom, have their own separate GOSIP specifications.

Government OSI Profile (GOSIP)

Government procurement specification for OSI protocols.

GraphicString

String of printable characters, that is, without format effectors, taken from the GeneralString character set.

group

Class of object that lets you associate, or manage as a group, a set of names that have something in common; maps a specific name (the group name) into a set of names, denoting the group members. DECdns uses groups internally for access checking. DECdns managers can create groups that assign several users a single set of access rights to DECdns names. Applications that use DECdns can create groups for purposes other than access control.

half-duplex circuit

Circuit designed for transmission in either direction, but only one direction at one time. *Contrast with* **full-duplex circuit**.

half-duplex transmission

Data transmission in either direction, but only one direction at a time. *Contrast with* **full-duplex transmission**.

handshaking sequence

Exchange of transport connection information between two tasks; takes place to enable the successful completion of a transport connection.

hardware address

For an Ethernet device, the unique Ethernet physical address associated with a particular Ethernet communications controller (usually in read-only memory) by the manufacturer.

HDLC

See **High-level Data Link Control Protocol**.

header

Control information placed before the data in a frame or a packet, such as source or destination specification, priority, and packet or frame identification.

hierarchical file model

Model of an internal file structure comprising an ordered tree of file-access data units (FADUs) that defines the basic file-access structure used by FTAM software.

hierarchical namespace

DECdns namespace with one or more levels of directories beneath the root directory.

high convergence

Directory convergence setting for which DECdns attempts to propagate an update to all replicas. If the attempt fails (for example, if one of the replicas is unavailable), a skulk is scheduled for within 1 hour. Background skulks occur at least once every 12 hours.

High-level Data Link Control (HDLC) Protocol

ISO, bit-oriented, Data Link layer (OSI layer 2) protocol that operates over synchronous, switched, or nonswitched communications links; supports a broad range of existing subsets, including the subset used in X.25 networks. Similar to LAPB.

hop

Logical distance between two nodes. One hop is the distance from one node to an adjacent node. *See also* **path length**.

host

Defined as:

- DECnet-Plus node that provides services for another node. For example, a load host supplies image files for a downline load.
- System running any implementation of the OSI transport protocol.

host-based PAD

Packet assembler/disassembler (PAD) situated at the X.25 host node and not within the PSDN.

host-echo mode

Mode in which the PAD can operate at which time characters typed at the terminal are sent across the PSDN, and echoed across the PSDN.

IA5String

String of characters from the IA5 character set. This 7-bit character set contains both graphic characters (printable) and control characters (nonprintable format effectors).

IDI

See **initial domain identifier**.

IDP

See **initial domain part**.

IEEE (Institute of Electrical and Electronic Engineers)

Standards body that sponsored, among other things, the 802 series of physical communications protocols, including the 802.3 protocol.

IEEE 802.2

IEEE protocol that defines frame formats for 802.3 LANs.

IEEE 802.3 local area network

LAN using the IEEE 802.3 specification; a CSMA-CD network.

IEEE Project 802

IEEE project set up to develop standards for LANs.

implementation profile

Standard subset of functions and features from the full set of OSI standards. Profiles are designed to specify certain user functions such as simple file transfer.

inaccuracy

Bounded uncertainty of a clock value as compared to a standard reference.

inbound connection

Logical link connection requests that a task receives.

incoming call

Packet sent to a DTE by its DCE when a call request packet has been received through the network from another DTE.

indication primitive

Primitive generated by a service provider to pass an incoming request on to a service user.

initial domain identifier (IDI)

Component of the IDP of an NSAP address; value allocated under the addressing authority identified by the authority and format identifier (AFI).

For example, the IDI value is based on CCITT recommendation E.163 and is created from a public switched telephone network (PSTN) number that has been assigned to a company or organization. The parts of this IDI are:

- World zone number
- Country or geographic area number
- Local number

initial domain part (IDP)

Part of an NSAP address; identifies the authority that allocated the IDP; consists of an AFI and an initial domain identifier (IDI).

initialization failure

Failure to start a circuit or establish a connection with an adjacency.

initiating host

Host on which an initiating OSI application executes.

initiating task

OSI task that requests a transport connection with another OSI application.

initiator

Defined as:

- OSI — Peer entity that requests an association.
Application process that receives requests from system users for application functions and that activates an application entity to start OSI communications.
- Virtual Terminal — Component that starts the interaction with a remote virtual terminal responder.

initiator identity

FTAM parameter whose usual purpose is to identify the calling user. The FTAM software responder equates the initiator identity (initiator ID) to either a local Digital UNIX or a local OpenVMS user name, depending on the local DECnet-Plus system.

Institute of Electrical and Electronic Engineers

See **IEEE**.

INTAP (Interoperability Technology Association for Information Processing, Japan)

Technical organization with the official charter to develop Japanese OSI profiles and conformance tests.

integrated IS-IS

See **integrated routing**.

integrated routing

DECnet-Plus routing feature that allows a DECnet-Plus intermediate system to forward DECnet-Plus packets and Internet Protocol (IP) packets.

Allows for DECnet-Plus networks with IP routers, OSI routers, and multiprotocol routers. Each network area can be OSI-only (contain a mix of OSI-only and multiprotocol routers, but forward only OSI packets), IP-only, or multiprotocol.

Integrated Services Digital Network (ISDN)

Technology offered by the telephone carriers of the world that combines voice and digital network services in a single medium, making it possible to offer to customers digital data services through a single "wire." The standards that define ISDN are specified by CCITT.

interface

Boundary between two parts of a system across which communication is possible; may be defined through hardware or software. Example: the network management command interface NCL.

interface data unit

Unit of information transferred across the layer n service access point between a layer $n+1$ entity and a layer n entity in a single interaction.

intermediate system

Routing system that receives data packets from a system on one subnetwork and passes them on to a system on another subnetwork; receives data packets from a source end system, or from the previous intermediate system on the route, and passes them on to the destination end system, or to the next intermediate system on the route.

Intermediate System to Intermediate System Protocol (IS-IS)

OSI protocol (ISO 10589) that describes the exchange of routing information between intermediate systems.

Internal Organization of the Network Layer (IONL)

OSI standard for the detailed architecture of the Network layer. This architecture partitions the Network layer into subnetworks interconnected by convergence protocols (equivalent to internetworking protocols).

International Consultative Committee for Telegraphy and Telephony

See CCITT.

International Organization for Standardization (ISO)

Nonprofit, international organization based in Geneva, Switzerland, whose members develop standards for international, multivendor, open networking. Coordinates national standards bodies, such as the British Standards Institution (BSI) and the American National Standards Institute (ANSI). Along with CCITT, is developing OSI. *See also* **Open Systems Interconnection**.

International Standard

Standard approved by ISO. Each standard has a reference number, for example:

- End System to Intermediate System routing (ES-IS): ISO 9542
- Connection-Oriented Network Service (CONS): ISO 8348
- Connectionless-mode Network Service (CLNS): ISO 8348/AD1
- X.25 Packet-Level Protocol (PLP): ISO 8208
- X.25 to provide CONS: ISO 8878

International Telegraph and Telephone Consultative Committee

See CCITT.

Internet

Collection of packet switching networks, interconnected by gateways along with protocols, that allow them to function as a single, large network; refers specifically to the Defense Advanced Projects Research Agency (DARPA) Internet and the TCP/IP protocols it uses.

Internet service

Network layer OpenVMS service that allows communications to span subnetworks that have distinct addressing rules and communication protocols; allows a source system to route messages across subnetworks using one or more intermediate systems to reach a target system.

Internet subaddress

Subaddress that identifies the OpenVMS VAX Internet service provider of an X.25 subnetwork; forms the last two digits of an adjacent system's X.25 subnetwork address.

Interoperability Technology Association for Information Processing, Japan

See INTAP.

interrupt

Defined as:

- DECnet-Plus — Event (other than an exception or branch, jump, case, or call instruction) that changes the normal flow of instruction execution; generally external to the process executing when the interrupt occurs.
- X.25 — To send data to a remote DTE by using flow control for control packets rather than for data packets. Interrupt packets are paired; an interrupt packet can be sent only if a previous interrupt has been acknowledged.

Interrupt Confirmation

Packet sent to:

- The local DCE by a DTE that has received an interrupt packet (DTE interrupt confirmation).
- An interrupting DTE by a DCE that has received a DTE interrupt confirmation packet through the network (DCE interrupt confirmation packet).

interrupt message

During nontransparent task-to-task communication, a user-generated message sent outside the normal exchange of data messages. (This usage of the term *interrupt* is contrary to the normal usage, which means to designate a software or hardware interrupt mechanism.)

interval

Combination of a clock's value and the inaccuracy associated with it; range of values in the clock value minus its inaccuracy, and the clock value plus its inaccuracy used by DECdts.

I/O status block (IOSB)

On a DECnet-Plus for OpenVMS system, data structure associated with the \$QIO system service; holds information about how the I/O request completes.

IONL

See **Internal Organization of the Network Layer.**

ISDN

See **Integrated Services Digital Network.**

IS-IS

See **Intermediate System to Intermediate System Protocol.**

ISO

See **International Organization for Standardization.**

ISO Development Environment

See **ISODE.**

ISO Reference Model for Open Systems Interconnection

ISO's international standard for Open Systems Interconnection. Communications model that defines a hierarchical architecture of seven layers designed to make possible free interconnection between multivendor systems.

More briefly called **OSI Reference Model** and **OSI Model**.

ISODE

ISO Development Environment; public domain implementation of the upper layers of OSI.

kernel group attributes

Subset of file attributes and activity attributes that must always be defined. FTAM implementations must support kernel group attributes.

known subclass entity group

All of the entities within a particular subclass that have the same parent entity.

LAN

See **local area network**.

LAP

See **Link Access Protocol**.

LAPB

See **Link Access Protocol Balanced**.

LAPB module

Module that defines the X.25 level 2 protocol used to exchange frames between a DTE and a DCE.

layer

Independent, self-contained set of interconnected functions with its own characteristic purpose and protocols within the OSI Reference Model. Each layer performs functions, as specified by the architecture, for the layers above it.

Layered Environment Services (LES)

DECnet-Plus software that provides a general-purpose environment for communications software; offers efficient scheduling, preserves the modularity of communications architectures, and permits the implementation of protocols in a high-level language. LES services also provide: support for layered protocols, memory management, timers, and protocol tracing.

LCN

See **logical channel number**.

leap seconds

Infrequent adjustment to coordinated universal time (UTC) to account for the irregularity of the earth's rotation.

leased line

See **dedicated line**.

LES

See **Layered Environment Services**.

level 1 router

Intermediate system that sends and receives data packets and routes them from one node to another within a single area. Routes messages for other areas to the nearest level 2 router.

level 2 router (area router)

Intermediate system that sends and receives data packets and routes them from one node to another within its own area and also between areas and between networks.

LFDP

See **long format data packet**.

line

Distinct, physical path from system to system that provides direct communication.

line controller

Hardware device at each node that manages communications over each line; handles the Physical and Data Link layers. Use of these devices can significantly shorten the time required by CPUs for communications.

line sharing

Form of X.21 switched line sharing in which many clients have access to a line, but only one client has access to a single call. *See also* **call sharing**.

line speed

Maximum rate at which data can be transmitted reliably over a line; varies with the capability of the modem or hardware device that performs the transmitting.

link

Defined as:

- Communications path between two nodes.
- DECdns — *See* **soft link**.

Link Access Protocol (LAP)

X.25-defined procedure for link control in which the DTE/DCE interface is defined as operating in two-way simultaneous asynchronous response mode (ARM) with the DTE and DCE containing a primary and secondary function. *See also* **Link Access Protocol Balanced**.

Link Access Protocol Balanced (LAPB)

X.25-defined procedure for link control in which the DTE/DCE interface is defined as operating in two-way asynchronous balanced mode (ABM). Provides for the reliable transfer of a packet from a host to an X.25 packet switch, which then forwards the packet on to its destination.

Modified form of HDLC that CCITT approved as the link level protocol for X.25 networks. *See also* **Link Access Protocol**.

link service access point (LSAP)

ISO 8802-2 (LLC) protocol identifier that identifies a Network layer entity. For example, hexadecimal FE identifies the LLC service access point used by CLNP.

link state algorithm

DECnet-Plus routing algorithm; calculates routes based on the shared knowledge of all the end systems and intermediate systems within a DECnet-Plus routing domain. Level 1 routers share information about all nodes in their area; level 2 routers share information about all areas reachable in the domain. Intermediate systems flood adjacency information to the other intermediate systems in the network. *Contrast with* **routing vector algorithm**.

link state packet (LSP)

Packet with DECnet-Plus routing (also called link state routing) control information. Intermediate systems exchange information about what adjacent nodes exist on each circuit and the cost associated with the circuit. This information is stored in LSPs and is sent to all other intermediate systems in the area. Each intermediate system collects the LSPs from all other intermediate systems in the area. From this information, each intermediate system constructs the least-cost paths through the network.

link state PDU (LSP)

See **link state packet**.

LLC

See **logical link control**.

LLC2

X.25 module that defines the data link protocol used on LANs that conform to the OSI LLC type 2 standard; allows systems on a LAN to communicate with remote nodes over an X.25 network.

LNO

See **Local Naming Option**.

load

See **downline loading**.

load assist agent

Image that provides additional data required to perform a downline load to a node in an OpenVMS cluster.

load host

Node from which a system image is downline loaded to a target node. *See also* **downline loading**.

load tool

Utility used to create, modify, and administer the Local Naming Option of DECdns.

local application address

Address for incoming FTAM communications that consists only of a p-address and which accesses the local responder.

local area field (LOC-AREA)

Two-octet field within the domain-specific part (DSP) that identifies a particular routing subdomain within the entire DECnet-Plus routing domain. Combines with the initial domain part (IDP) field and the preDSP field to form the **area address**.

local area network (LAN)

High-speed, privately-owned data communications network, for example, Ethernet, that covers a limited geographical area, such as a group of buildings, a single building, or a section of a building. *Contrast with* **wide area network**.

local data

Any data stored locally by a system. Example: file data or the system information that maps to ISO service parameters.

local DTE

DTE at which the user is located.

local echo mode

Packet assembler/disassembler (PAD) mode during which characters typed at the terminal are echoed by the PAD.

local management access interface

Interface between the agent and the agent access module.

local name file

Text file containing a list of node names, synonyms, and network addresses. This information constitutes the local namespace for a DECdns clerk cache that is using the Local Naming Option.

Local namespace

A discrete, nondistributed namespace that stores name and address information locally in database files. The Local namespace is independent of DECdns and replaces functionality previously provided by the DECdns Local Naming Option. Depending on the number of address towers stored, the Local namespace is designed to scale up to at least 100,000 nodes.

Local Naming Option (LNO)

DECdns option that provides a per-node, nondistributed, namespace for small-scale DECnet-Plus networks that optionally do not make use of DECdns servers.

local node

Node at which the user is located.

local server

DECdts server that synchronizes with its peers and provides its clock value to other servers and clerks on the same LAN.

local set

All of the DECdts servers in a particular LAN.

logging

Network management facility that collects network events at a logging sink, such as a file or console. *See also* **event dispatcher**.

logical channel

Logical link between a DTE and its DCE. The physical communications line between a DTE and DCE is divided into a set of logical channels.

logical channel number (LCN)

Unique reference number that identifies a logical channel. A DTE recognizes a virtual circuit by its associated LCN.

logical connectivity

Ability of nodes to communicate.

logical link

Temporary connection between processes on source and destination nodes (or between two processes on the same node).

Logical Link Control (LLC)

OSI protocol used on LANs to provide the Data Link layer service.

long format data packet (LFDP)

Long format protocol header used for data packets on Ethernet subnetworks when communicating with DECnet Phase IV nodes. *Contrast with* **short format data packet**.

lookup

Process during which a DECdns clerk receives a request from a client application, sends the request to a DECdns server, and returns the information to the client.

loop node

Local node that is associated with a particular line and is treated as if it were a remote node. All traffic to the loop node is sent over the associated line; used for loopback testing.

low convergence

Setting that controls the degree to which DECdns attempts to keep all replicas of a directory consistent; indicates that DECdns does not immediately propagate an update, but waits for the next skulk to distribute all updates that occurred since the last one.

LSAP

See **link service access point**.

LSP

See **link state packet**.

mail

See **Message Handling System** and **Session Control application database**.

mail exploder

Part of an electronic mail delivery system that allows a message to be delivered to a list of addressees; used to implement mailing lists.

mail gateway

System that connects two or more electronic mail systems (especially dissimilar mail systems on two different networks), and transfers messages between them.

Maintenance Operations Module (MOM)

Maintenance Operations Protocol that defines downline loading and upline dumping.

Maintenance Operations Protocol (MOP)

DNA network management protocol used to perform functions such as downline loading, upline dumping, and circuit testing.

management access name

Another term for an entity name; label associated with an entity in order to identify or locate it for network management.

Management Access Protocol

Application layer protocol between the director and the entity.

management access relationship

Relationship between a parent entity and a child entity that indicates that the parent is capable of passing directives to its child entities; used to find entities for the purposes of managing them, except for the special cases of creating and recovering them. The management access relationships between entities are reflected in the management names of entities.

Management Event Notification (MEN) Protocol

DECnet-Plus network management protocol used for exchanging event information.

management hierarchy

Order of manageable DECnet-Plus entities defined strictly for the purpose of naming entities; does not imply or define how entities interact.

Management Information Control and Exchange (MICE) Protocol

DECnet-Plus network management protocol used to exchange management requests and data between nodes.

management model

See **distributed system management model**.

management namespace

Set of all entity names registered in the DECdns namespace.

master replica

First instance of a specific directory in the DECdns namespace. DECdns can create, update, and delete object entries and soft links in a master replica. The master replica is the only replica where DECdns can create child pointers and the only replica from which certain skulk operations involving new directories, deleted directories, and soft links can be performed.

maximum window

NSP and OSI transport entity characteristic for control of the number of data segments (PDUs) allowed to be transmitted over a particular transport connection before at least one acknowledgment must be returned from the destination system. If the number of PDUs already transmitted equals the maximum window and no corresponding acknowledgments have been received, transport stops sending PDUs over the transport connection and waits for an acknowledgment message.

maximum transmission unit (MTU)

Largest possible unit of data that can be sent on a given physical medium. Example: The MTU of Ethernet is 1500 bytes. *See also fragmentation.*

medium convergence

Directory convergence setting for which DECdns attempts to propagate an update to all replicas. If the attempt fails, the next scheduled skulk makes the replicas consistent. Skulks occur at least once every 12 hours.

MEN Protocol

See Management Event Notification Protocol.

message

Message block, or a series of message blocks, that constitute a logical grouping of information; each is delimited by communications control characters.

message handling system (MHS)

System of message user agents, message transfer agents, message stores, and access units which together provide OSI electronic mail; specified in the CCITT X.400 series of recommendations.

message transfer agent (MTA)

OSI application process used to store and forward messages in the X.400 message handling system. Equivalent to Internet mail agent.

MHS

See message handling system.

MICE Protocol

See Management Information Control and Exchange Protocol.

modem (modulator/demodulator)

Device that translates digital signals (electrical impulses) generated by a computer into analog signals (tones) that can be transmitted over telephone lines, and vice versa.

modem connect

Defined as:

- DNA — Class of communication links governed by industry standards for modem connection.
- X.25 — Module that defines the physical lines connecting a system to an X.25 network.

modulator/demodulator

See **modem**.

MOP

See **Maintenance Operations Protocol**.

MTA

See **message transfer agent**.

MTU

See **maximum transmission unit**.

multiaccess channel

Special type of broadcast circuit, for example, Ethernet, on which many transmitters contend for access.

multiarea network

See **multiple area network**.

multicast

Type of broadcast transmission in which a copy of a packet is delivered to a subset of all nodes in a subnetwork that are listening on the same address. *See also* **broadcast**.

multicast address

Address that designates a subset of nodes that are all listening for packets destined to this address.

multicast addressing

Addressing mode in which a data packet is targeted to a group of nodes that are of the same type, for example, all level 1 routers or all level 2 routers.

multicircuit end system

End system with two or more connections to the network, for example, two separate LANs; provides increased performance and reliability because traffic is split among the circuits, and if one circuit fails, connectivity is maintained.

Although traffic can be sent and received over any of the links, traffic is not forwarded from one link to another; does not perform the functions of a router.

multidrop

See **multipoint circuit**.

multihomed system

End system or intermediate system with more than one assigned address.

multiple area network

WAN divided into areas, with each area being a group of nodes. Nodes are grouped into areas for hierarchical routing purposes.

multiplexing

Using a single connection to carry several data streams and the mechanism for assigning these streams to that connection. For example, both NSP and the OSI transport service can multiplex several transport connections onto a single network connection.

multipoint circuit

Circuit that connects multiple systems; one is the **control station** and the others are **tributaries**. (DECnet-Plus does not support multipoint connections.) Also called **multipoint link** and **multidrop**. *Contrast with **point-to-point circuit**.* *See also **control station** and **polling**.*

multiprotocol routing

*See **integrated routing**.*

name resolution

DECdns process of mapping a name into the corresponding address.

name server

*See **DECdns server**.*

name service

Term for the software that manages the node name and addressing information for DECnet.

namespace

The set of names accessible to a name service. DECdns stores names in directory replicas in clearinghouses at each DECdns server. The Local namespace stores names in local database file.

namespace creation timestamp (NSCTS)

Unique timestamp (with attribute name of DNS\$NSCTS) automatically assigned to a namespace when installing and configuring DECdns on the first node in a new namespace.

namespace hierarchy

Logical hierarchy, or tree, of directories that exist beneath the root directory of a DECdns namespace.

namespace name

*See **namespace nickname**.*

namespace nickname

Name that a network manager or administrator assigns to a namespace when installing and configuring DECnet-Plus on the first node in a new namespace.

The following reserved namespace nicknames `local:` and `domain:` on a node full name indicate to DECnet-Plus that the information for the node is contained in the Local namespace (for `local:`) and in DNS/BIND (for `domain:`).

National Bureau of Standards

See **NIST**.

National Institute of Standards and Technology

See **NIST**.

navigation window

Navigation aid in the DECdns Browser.

NBS

National Bureau of Standards. See **NIST**.

NCL

See **Network Control Language**.

NCL script

File of NCL commands.

NCP

See **Network Control Program**.

NET

See **network entity title**.

network

Consists of two or more computer systems linked by communications hardware and software; an *open network* is a network of open systems; an *open system* is a computer system with communications software that implements formal, international open networking standards, for example, the OSI standards or RFC-compliant TCP/IP.

network address

Address that identifies a specific system on a network; can be an X.25, a hardware, or an NSAP address. Made up of a transport template name for the local DECnet-Plus system and a Network layer address of a target system. See also **network service access point**.

network addressing

See **DECnet-Plus addressing**.

network architecture

Specification of a network's functions and its parts, together with the ways in which the network is organized; specifies the layers of different functions in the network, ranging from data transmission at the lowest levels to user applications at the highest levels.

network connect block (NCB)

Data structure with information needed to set up a transport connection, or to accept or reject a request to set up a transport connection.

network connection

Association, established by the Network layer, for the transfer of data between two or more entities in the Transport layer.

Network Control Language (NCL)

Command-line interface to DECnet-Plus network management directors; used to manage DECnet-Plus nodes and their network components.

Network Control Program (NCP)

Command-line interface for managing DECnet Phase IV nodes and their network components.

network delay

Time it takes to get a unit of data from the source of a transmission to the destination; usually refers to delay from the network and not by system-dependent application processing delays at source and destination nodes.

network diameter

Distance (number of hops) between the two nodes in the network with the greatest reachability distance. The reachability distance is the path with fewest number of hops between two nodes.

network entity title (NET)

Address by which the Network layer is identified; identifies a particular system in a particular network. Is the same structure as an NSAP, but with a 0 selector field (SEL). Has two primary fields:

- Initial domain part (IDP), which consists of two subfields:
 - Authority and format identifier (AFI)
 - Initial domain identifier (IDI)
- Domain-specific part (DSP), which consists of four subfields:
 - preDSP
 - Local area (LOC-AREA)
 - Node ID
 - A 0 transport selector

When combined with a real transport selector, is transformed into an **NSAP**. *See also* **network service access point** and **transport selector**.

Network Information and Control Exchange (NICE) protocol

Protocol used to exchange DECnet Phase IV network management information.

Network layer

Layer 3 in the OSI Reference Model; permits communications between network entities in open systems on a subnetwork, intermediate systems, or both. Layer for all routing functions. Provides two types of services: CONS and CLNS. Also includes DECnet Phase IV support. Is functionally divided into two sublayers: the subnetwork independent sublayer and the subnetwork dependent sublayer. *See also* **CONS** and **CLNS**.

network layer protocol data unit (NPDU)

Combination of a service data unit (SDU), which is the basic unit of user data that the Network layer receives from the Transport layer above, plus a protocol control information (PCI) header subsequently added by the Network layer.

network management

Services for managing a DECnet-Plus network, such as configuring and tuning the network software, monitoring network performance, maintaining network operation, and diagnosing and troubleshooting network problems. User-performed with NCL. *See also* **Network Control Language**.

network performance

How a network performs, as measured against the expectations or requirements of users, customers, designers, or implementors, or as claimed by sales and marketing personnel. The criteria for network performance include parameters such as throughput, response time, and resource utilization.

Network service

OSI service provided to entities in the Transport layer at the upper boundary of the Network layer, as defined in International Standard ISO 8348.

network service access point (NSAP)

Global network address of a DECnet-Plus system; addressable point at which a network entity provides the network service to a network user; complete address that identifies both the particular network system and the transport module on that system that is to receive the data.

A DECnet-Plus address (NET) that contains a nonzero selector field (SEL). Has two primary fields:

- Initial domain part (IDP), which consists of two subfields:
 - Authority and format identifier (AFI)
 - Initial domain identifier (IDI)
- Domain-specific part (DSP), which consists of four subfields:
 - preDSP
 - Local area (LOC-AREA)
 - Node ID
 - Transport selector (SEL)

See also **network entity title** and **transport selector**.

network service data unit (NSDU)

Block of data transferred between the Transport layer and the Network layer.

Network service primitive

Basic operation carried out by the network service at the request of a network user.

Network service provider (Network service, NS Provider)

Software that provides a network service.

Network Services Protocol (NSP)

DNA protocol that operates in the DNA Transport layer. DECnet Phase IV nodes use NSP. NSP and OSI transport can reside simultaneously on a DECnet-Plus node.

Network service user (network user)

Software running above the Network layer on a given host and using the network service to communicate with some other software, possibly on another host.

network status notification

Notification with information about the state of both logical and physical links over which two tasks communicate. A nontransparent task can use this information to take appropriate action under conditions such as third-party disconnections and a partner's exiting before I/O completion.

network task

Nontransparent task that can process multiple inbound connection requests; that is, it has a declared network name or object number.

NICE protocol

See **Network Information and Control Exchange protocol**.

nickname

See **namespace nickname**.

NIST (National Institute of Standards and Technology)

Standards organization of the United States government. (Formerly called the National Bureau of Standards (NBS).) *See also* **OIW**.

node

Defined as:

- DECnet-Plus — *See* **system**.
- FTAM — In the hierarchical file model, the structural element of a file with which a data unit is associated and whose sequential relationship to other nodes, if any, determines the file's structure.

node address

Required, unique numeric identification of each node in a network; provides addressing information for other nodes to access it. These addresses must reflect the logical network configuration (where nodes fit into the network), and may need to change as that configuration changes.

In DECnet-Plus, the OSI NSAP standard defines the syntax of a network address. *See also* **network service access point**.

node name

Alphanumeric identification associated with a node address of a system in a one-to-one mapping.

node name synonym

See **node synonym**.

node synonym

DECnet Phase IV-compatible node name (six characters or less) that maps to a DECnet-Plus-style name for the same node. For example, XYZ_CORP:.sales.west_coast.ElanaCole might map to ElanaC.

node synonym directory

Required DECdns directory called .DNA_NodeSynonym that stores node synonyms. Created by the decnet_register by default during DECnet-Plus configuration.

nonadjacent nodes

Nodes without direct lines between them; can communicate only if intermediate systems forward the data along the path between the source and the destination.

nonbroadcast circuit

Circuit other than a broadcast circuit. For example, a multipoint DDCMP circuit is not a broadcast circuit because a packet cannot be received by more than one node.

nonbroadcast subnetwork

Subnetwork, or area, made up of one of these kinds of connections: multipoint links, permanent point-to-point links, or dynamically established data links.

nonrouting node

End system.

nontransparent task

Form of device-dependent I/O that uses system services to perform network-specific functions; can initiate and complete a logical link connection, exchange messages between two tasks, and terminate the communication process. Application that has direct access to network-specific information and operations, such as optional user data on connects and disconnects and interrupt messages, to monitor the communications process; can receive and process multiple inbound connection requests.

Example: a nontransparent task on a DECnet-Plus for OpenVMS system can create and use an OpenVMS mailbox to receive information that is not available to a transparent task with transparent communication.

normal data

User information sent under normal circumstances over the transport normal data service.

Data that is normally exchanged during communication, as distinct from expedited data sent over a connection. *Contrast with* **expedited data**.

normal mode

HDLC operational mode used over half-duplex links. *Contrast with* **balanced mode**.

normalization

Estimation of the change in a counter value over a specified time period.

NSAP

See **network service access point**.

NSAP address

Address of the network service access point. See also **network service access point**.

NSAP selector

See **transport selector**.

NSCTS

See **namespace creation timestamp**.

NSP

See **Network Services Protocol**.

null modem

Simple form of modem connection where only the data interchange circuits, and not the modem control circuits, are used.

Null Protocol

Protocol used in the Network layer of a LAN in place of the Internet Protocol (IP); used when the routing functions of IP are not required, for example, when the communicating hosts are on the same LAN.

null qualifier

Qualifier that a VAX FTAM facility accepts for the sake of compatibility but ignores when executing a command involving any remote files.

object class

Attribute of an object that reflects the purpose of that object within an application. The object class can be specific to one application or shared among a group of applications. An application interprets and uses an object's class-specific attributes based on the object's class.

object entry

Entry defining a physical resource (such as a node, disk, or application) stored in DECdns.

object identifier

Standard, unique name for an informational object such as a specific piece of information, a definition, or a specification. See also **ASN.1**.

OIW (OSI Implementors Workshop)

North American regional forum at which OSI implementation agreements are decided; equivalent to EWOS in Europe and AOW in the Pacific. (Also called **NIST OIW** and the **NIST Workshop**.)

opaque full name

Internal representation of a DECdns full name, beginning with the namespace creation timestamp (NSCTS); stored in binary format.

opaque simple name

Internal representation of a DECDns simple name; stored in binary format.

open network

Network made up of open systems. *See also* **Open Systems Interconnection**.

open system

System with communications software that implements the OSI standards for open networking. *See also* **Open Systems Interconnection**.

Open Systems Interconnection (OSI)

Set of international networking standards developed by the International Organization for Standardization (ISO). OSI standards are based on ISO's seven-layer model for communications and an associated set of communications protocols and services.

ISO is also developing a suite of protocols like the TCP/IP suite named TP4/IP. *See also* **International Organization for Standardization**.

operational mode

In High-level Data Link Control (HDLC), the particular operational state or protocol being used.

originating packet

Packet from the local node's Transport layer.

origination

Beginning point of communications on a circuit.

OSI

See **Open Systems Interconnection**.

OSI Applications Kernel Software (OSAK software)

Open System Application Kernel (OSAK) is Digital's implementation of the OSI upper layers. It provides OSI session, presentation and application services. These services are used by OSI applications such as FTAM, VT, X.400, and X.500. In addition, by using the OSAK programming interfaces, which provide access to OSI session, presentation and application layers, users can develop applications which layer on Digital's implementation of the OSI stack.

OSI Reference Model

See **ISO Reference Model for Open Systems Interconnection**.

OSI network address

See **network service access point**.

OSI presentation address

Address used to locate an OSI application entity; consists of an OSI network address and up to three selectors, one each for use by the transport, session, and presentation entities. *See also* **p-address**.

OSI Reference Model

See **ISO Reference Model for Open Systems Interconnection**.

OSI Transport address

See **transport service access point**.

OSI Transport filter

Filter used by the Transport layer for incoming calls using CONS over the X.25 network.

OSI transport template

Template defined in the X25 Access module; used by the Transport layer for outgoing calls using CONS over the X.25 network.

outbound connection

Task's transport connection request for a logical link connection to another node.

out-of-order packet caching

Taking packets of a user data message that have been received out of order (as a result of the routing process called **path splitting**) and reassembling them into their proper order; must be supported by destination nodes that receive data packets that have been forwarded over different paths. Also called **out-of-order packet reassembly**. See also **path splitting**.

packet

Defined as:

- Routing — Unit of data routed from a source node to a destination node. A message may be sent as several packets.
- X.25 — Unit of data switched through a PSDN; normally a user data field has a header carrying destination and other information.

packet assembler/disassembler (PAD)

Device at a PSDN node that allows access from an asynchronous terminal, such as a Digital VT series terminal. The terminal connects to the PAD, which formats the data sent from the terminal into packets (is assembled). Data sent to the terminal in packet format is disassembled by the PAD before transmission to the terminal.

packet fragmentation

See **fragmentation**.

packet level

X.25 protocol level; defines the procedure for the formatting of packets and for packet exchange. Also known as **Level 3**.

packet looping

Condition in which a packet revisits a node. See also **aged packet**.

packet mode DTE

DTE that can handle data in packet form and can assemble/disassemble packets, for example, a computer.

packet size

Amount of data in a packet.

packet switching

Method of transmitting messages, using data packets that occupy a channel only for the duration of transmission of the packet. Each packet sent can take different routes to its destination and packets from different users can be interleaved across the network connection during transmission. Long messages are subdivided into short packets (blocks of fixed length), which are then routed to their destination using addressing information carried by the packets.

packet switching data network (PSDN)

Refers to both public and private packet switching networks. For public networks, also means the set of equipment and interconnecting links that provide a public packet switching service to subscribers within a particular country.

packet switching exchange (PSE)

Equipment in a PSDN that is responsible for accepting and routing packets and ensuring their correct arrival.

packet switching network

Computer network that uses packet switching.

PAD parameter profile

X.25 — List of PAD parameter settings. *See also* **packet assembler/disassembler**.

p-address (presentation address)

Specifies service access points (SAPs) for the service providers of all the upper layers to be accessed. For the FTAM software, a p-address always contains presentation, session, and transport selectors. It also must have an NSAP. Within a p-address, each selector is terminated on its right by a delimiter (.). The sequence of selectors and delimiters for an FTAM p-address is:

psap. ssap. tsap.nsap.

See also **OSI presentation address**.

parent directory

Directory that has one or more levels of directories beneath it in a DECdns namespace. A directory is the parent of any directory immediately beneath it in the hierarchy.

parent entity

Entity that has created another entity (the child entity). Higher-level entity that forwards directives to its child entities as defined by the management access relationship.

partial support

Level of support in which FTAM software supports a constant value, such as No value available, in the virtual filestore, because the attribute lacks direct mapping to either a UFS attribute on a DECnet/OSI for Digital UNIX system or an RMS attribute on a DECnet-Plus for OpenVMS system.

path

Physical lines between source nodes and destination nodes; can comprise a sequence of connected nodes. The path that the data takes through the network is transparent to users.

path cost

The sum of the circuit costs along a path between two nodes.

path length

Total distance (the number of circuits) between a source node and a destination node, measured in hops. Each line between systems, including routing nodes and end nodes, equals one hop. *See also* **network diameter**.

path splitting

Routing process that splits the transmission load destined for a node over several paths of equal path cost. Paths are less likely to become congested, resulting in improved use of network resources. A destination node receiving data that has been split over several paths must support out-of-order packet caching. *See also* **out-of-order packet caching**.

PCI

See **protocol control information**.

PDU

See **protocol data unit**.

peer entities

Corresponding entities at the same layer on different open systems.

permanent virtual circuit (PVC)

Permanent, logical association between two DTEs that is analogous to a leased line. Transmission of packets on a PVC needs no call setup or call clearing by the DTE; packets are routed directly by the network from one DTE to the other. Supported in DECnet-Plus for DECnet Phase IV compatibility.

phase

Discrete period of protocol exchanges having a specific purpose that involves a single FTAM function. Phases occur sequentially and cannot overlap. Often two phases within one regime are paired, so that after the initial phase occurs, the other must eventually follow. Examples: the selection and deselection phases.

Phase IV area

See **DECnet Phase IV area**.

Phase IV end node

See **DECnet Phase IV end node**.

Phase IV level 1 router

See **DECnet Phase IV level 1 router**.

Phase IV level 2 router

See **DECnet Phase IV level 2 router**.

physical address

Unique address of each physical connection of a node to the physical media.

physical connection

Physical layer communications path between two systems.

physical connectivity

Physical layer connectivity that is a result of nodes being attached to each other via active lines and nodes.

Physical layer

Bottom layer (Layer 1) in the OSI Reference Model; connects systems to the physical communications media.

Physical level

X.25 protocol level that defines the characteristics of the physical link between the user's equipment and the PSDN equipment. Also known as **level 1**.

physical media

Any means in the physical world for transferring signals between OSI systems; considered to be outside the OSI Reference Model and, therefore, sometimes referred to as "Layer 0." The physical connector to the media can be considered as defining the bottom interface of the Physical layer, that is, the bottom of the OSI Reference Model.

plug-in

User-configurable portion of the director.

point-to-point circuit

Circuit that connects only two nodes. A point-to-point configuration requires a separate physical connection between each pair of nodes. Point-to-point systems communicate directly with other systems. (DECnet-Plus supports point-to-point and broadcast connections.) *Contrast with* **multipoint circuit**.

point-to-point line

Line that connects two systems by using a single circuit.

polling

Scheme for how the tributaries on a multipoint circuit gain access to the communications path.

Tributaries cannot access the path until polled by the control station, usually on a sequential basis. A tributary waits its turn to access the communications path. Tributaries always send data to the control station, which forwards the data to other tributaries, if necessary.

portal mode

Mode of the DEC X25gateway system that allows systems running X.25 software, both Digital systems and multivendor systems, to communicate over PSDNs and over DECnet-Plus networks.

POSI (Promoting Conference for OSI)

Consortium of executives from the six major Japanese computer manufacturers and Nippon Telephone and Telegraph; sets policies and commits resources to promote OSI.

positional file transfer

NIST implementation profile that requires an FTAM implementation to be able to transfer files with an unstructured or sequential flat constraint set, that is, files with an FTAM-1, FTAM-2, or FTAM-3 document type. Also known as **NBS profile T2** and **SPAG profile A112**. *See also simple file transfer.*

Postal, Telegraph and Telephone Authority (PTT)

Country's national communications provider, generally European.

presentation address

See p-address.

presentation connection

Link established by the OSI Presentation layer between two users of the presentation service.

presentation context

Association of an abstract syntax with a transfer syntax.

Presentation layer

Layer 6 in the OSI Reference Model; coordinates the conversion of data and data formats to meet the needs of the individual application processes; provides the proper means of transferring information while preserving its meaning.

Presentation service

OSI service provided by the Presentation layer, as defined in International Standard ISO 8822.

primitive

See service primitive.

primitive name

Name that denotes a single, unique object.

principal

Individual user, or group of users, for which access rights to DECdns names can be assigned and checked. A principal name is part of an access control entry (ACE). *See also access control entry.*

private line

See dedicated line.

Private Management Domain (PRMD)

X.400 message handling system private mail system. Example: NASAmail. *See also Administration Management Domain.*

profile

Defined as:

- Virtual Terminal — Specific set of predefined values that both systems in the Virtual Terminal association use to translate information by means of local mapping. Each profile has a unique name, which a Virtual Terminal initiator supplies to a Virtual Terminal responder when negotiating an association.
- X.25 — File that defines the default values for the call parameters of outgoing calls.

Promoting Conference for OSI

See **POSI**.

protocol

Defined as:

- General data processing — Set of rules and procedures.
- Data communications — Set of rules and procedures that defines message exchange between communicating processes; defines the format and contents of messages.

protocol control information (PCI)

Encoded specifications about services and their parameters; added by an OSI entity to the service data unit (SDU) passed down from the layer above, all together forming a PDU.

protocol data unit (PDU)

Data units (messages or blocks of data) passed between peer entities on different open systems. The information is made up of protocol control information (PCI) from the current layer and, possibly, user data from the layer above. OSI terminology for **packet**.

protocol ID

Five-byte field in the header of a Subnetwork Access Protocol (SNAP) frame on a LAN, used to identify the Data Link layer client at the destination system receiving this frame.

protocol machine

Set of data structures and routines that implements a specific protocol and controls the progress of a communication between peer entities.

protocol overhead

Part of communications data or processing not directly consumed by the users but necessary to successfully bring about the transfer of user information. Also called **control data**.

protocol sequence

Ordered list of protocol identifiers.

protocol transparency

Degree to which users of underlying protocols are aware of the specifics of those protocols.

protocol type

Two-byte field in the header of an Ethernet frame, used to identify the Data Link layer client at the destination system receiving this frame.

proxy login

Procedure that permits a remote user to access a specific account at the local node, without supplying the user name and password.

PSDN

See **packet switching data network**.

PSTN

See **public switched telephone network**.

PTT

See **Postal, Telegraph and Telephone Authority**.

public data network

PSDN administered by a public service provider.

public switched telephone network (PSTN)

Public packet switching data network.

PVC

See **permanent virtual circuit**.

RARE (Reseaux Associes pour la Recherche Europeenne)

European association of research networks.

RD

See **redirect packet**.

reachable address

Address prefix in a reachable-address table, for example, an address prefix that maps a destination NSAP address to a destination DTE address/DTE class pair for an outgoing call into an X.25 subnetwork.

reachable-address table

Table on a DECnet-Plus intermediate system that lists address prefixes that are reachable over a circuit; manually-entered routing information that allows packets to be forwarded to subnetworks that do not participate in DECnet-Plus routing, such as DECnet Phase IV subnetworks, X.25 subnetworks, and multivendors' systems.

reachable node

Node to which the local node has a usable communications path.

read access

Access right that grants the ability to view DECdns data.

read-only replica

Copy of a DECdns directory in which applications cannot make changes. Although applications can look up information (read) from it, they cannot create, modify, or delete entries in a read-only replica. Read-only replicas become consistent with other, modifiable replicas of the same directory during skulks and routine propagation of updates.

real file

FTAM-named collection of user data and its attributes that resides in a real filestore.

real filestore

Organized collection of files on a real FTAM system that includes their attributes, such as file names. Each virtual file reference maps to a real file in the real filestore.

reassembly

Routing process of reconstructing a complete data message from received fragments. *See also fragmentation.*

receive

Routing process that handles incoming messages that have been sent to the router.

receiver

FTAM entity that reads, or receives all or part of a file's contents, during the file-data transfer regime. During an association, an FTAM entity can alternate between sending and receiving data in any order.

Record Management Services (RMS)

On DECnet-Plus for OpenVMS systems, set of file-management procedures called by programs to process files and records within files. Used by FTAM software.

redirect NPDU

Network PDU issued by an intermediate system to a source node when it forwards a data network PDU onto the same circuit from which it was received; includes the Data Link layer address to which the network PDU was forwarded. This indicates to the sender of the original data network PDU that it should send subsequent data network PDUs destined for the same NSAP address directly to that address, as the destination can be reached either directly or by a better route.

redirect packet (RD)

Redirect packet as defined by ISO 9542.

Reference Model

See ISO Reference Model for Open Systems Interconnection.

regime

Period during which communicating FTAM entities share a standard state that permits a distinct set of FTAM services, such as those associated with data transfer, that are constrained by the current values of activity attributes.

relative time

Discrete time interval that is usually added to or subtracted from an absolute time; used by DECdts software.

Reliable Transfer Service Element (RTSE)

Lightweight OSI application service used above X.25 networks to handshake application PDUs across the session service and TP0. (Not needed with TP4.)

remote application address

Address for outgoing FTAM communications that contains a p-address and that accesses a remote responder. On DECnet/OSI for Digital UNIX systems, also contains an application-entity (AE) title. On DECnet-Plus for OpenVMS systems, optionally contains other information, for example, an AE-title.

remote DTE

DTE in a network other than the one at which the user is located.

remote node

Node in the network other than the local node.

Remote Operations Service Element (ROSE) protocol

Lightweight remote procedure call (RPC) protocol, used in these OSI application protocols: message handling, directory, and network management. Transaction-based protocol (as opposed to standard file transfer protocols). *See also* **remote procedure call**.

remote procedure call (RPC)

Easy and popular paradigm for implementing the client/server model of distributed computing. A request is sent to a remote system to execute a designated procedure, using specified arguments, and the result is returned to the caller. The OSI RPC protocol is Remote Operations Service Element (ROSE).

remote task

Task either executing at a remote host or originating there.

repeater

Bidirectional device that amplifies or resynchronizes signals into standard voltages, currents, and timing; propagates electrical signals from one Ethernet to another without making routing decisions or providing packet filtering; Physical layer intermediate system. *See also* **bridge** and **router**.

repertoire

Virtual Terminal — Set of representations of symbols, such as numeric, graphic and control characters. Each profile has a specific repertoire associated with it.

replica

Copy of a DECdns directory. The first instance of a directory in the namespace is the master replica. When DECdns managers make copies of the master replica to store in other clearinghouses, all of the copies, including the master replica, become part of the directory's replica set. Replicas other than the master are read-only replicas.

replica set

Set of all copies of a directory. Information about a directory's replica set is contained in an attribute of directories and child pointers called DECdns\$Replicas. The attribute contains the type of each replica master, secondary, read-only and the clearinghouse where it is located. When skulking a directory, DECdns refers to the directory's replica set to ensure that it finds all copies of that directory. During a lookup, DECdns may refer to the replica set in a child pointer when trying to locate a directory that does not exist in the local clearinghouse.

requester

Peer entity that requests a service.

Service user that initiates a particular action, such as requesting a service from a peer entity.

request primitive

Primitive issued by a service user to ask for a function and provide parameter values, if necessary.

responder

Defined as:

- OSI — Peer entity that receives an association request.
- FTAM — Application process that accepts an incoming request from an initiator to establish an FTAM regime; maps the incoming requests to actions performed on the real filestore.
- Virtual Terminal — Component that reacts to requests from a remote virtual terminal initiator.

responding host

Host on which a responding OSI application executes.

responding task

Task that receives and processes a request for an OSI transport connection.

response primitive

Primitive issued by a service user to confirm negotiated parameter values that the user accepts and to return attribute values that the initiating user requests.

resynchronization

Process that enables the recovery of user information lost or corrupted during transfer across an association. Sets the association back to the state it was in at a specified point in the transfer.

retransmission

Method of error recovery in which stations receiving messages acknowledge the receipt of correct messages and, on receipt of incorrect messages, either do not acknowledge or acknowledge in the negative. The lack of acknowledgment or receipt of a negative acknowledgment indicates to the sending station that it should transmit the failed message again.

RMS

See **record management services**.

root directory

Top directory in the DECdns namespace, designated by a dot (.).

ROSE

See **Remote Operations Service Element**.

round-trip delay

Total time, during communications that implement a protocol with positive acknowledgments, for a message to be transmitted, arrive at its destination, and its corresponding acknowledgment to be sent and subsequently received by the sender of the original message.

route through

Packets not destined for the local node. *See also* **forward**.

router

See **intermediate system**.

routing

Network layer function, implemented in intermediate systems, that determines the path along which data travels to its destination and the movement of that data. *See also* **decision**.

routing algorithm

See **link state algorithm** and **routing vector algorithm**.

routing domain

Collection of end systems, intermediate systems, and subnetworks that operates according to the same routing algorithm, and in which routers exchange routing information; resides within an administrative domain. Within an administrative domain, the routing domains can be DECnet-Plus routing domains. Can be divided into subdomains, or areas.

routing layer

See **Network layer**.

routing node

See **intermediate system**.

routing protocol

See **routing vector algorithm** and **link state algorithm**.

routing record

Database entry that associates a subnetwork address of an adjacent system with an NSAP address.

routing vector algorithm

DECnet Phase IV routing algorithm; routers exchange network reachability information with adjacent routers. *Contrast with* **link state algorithm**.

RPC

See **remote procedure call**.

RTSE

See **Reliable Transfer Service Element**.

SAP

See **service access point**.

SDU

See **service data unit**.

security attributes

Attributes that control access to a file on an FTAM system.

Security filter

Entity that defines an access control list (ACL) on an X.25 system that controls who can access a filter.

security information

FTAM initiator identity and, optionally, a filestore password and account. DECnet-Plus users can specify security information as part of the DECnet-Plus for OpenVMS or DECnet/OSI for Digital UNIX file specification. On DECnet-Plus for OpenVMS systems, users can also specify security information with a DCL qualifier.

segmentation

Division of a block of user data into smaller units, known as segments, for transfer across a connection.

selector

Identifier used by an OSI entity to distinguish among multiple service access points (SAPs) at which it provides services to the layer above; format is one of these: ASCII string, hexadecimal string, or null.

semiautomatic management

Type of network management in which a human system manager is not directly involved in a network management decision; rather, management software in a director makes decisions and takes actions on behalf of the system manager.

send routing message flag

Link State algorithm—Flag associated with each link state packet for each circuit on an intermediate system; when set, it indicates that the link state packet needs to be transmitted on that circuit.

send sequence number packet flag

Defined as:

- Link state algorithm — Flag associated with each link state packet for each circuit on an intermediate system; when set, it indicates that information about the packet needs to be included in a partial sequence number packet transmitted on that circuit.
- Routing vector algorithm — Flag associated with each circuit on a router; when set, it indicates that a routing vector message needs to be transmitted on that circuit.

sender

FTAM entity that writes (sends) either part of or all of the contents of a file during the file-data transfer regime.

sent packet

Packet passed from the local node's Network layer to its Data Link layer.

sequence number

Routing — Field carried in link state packets (LSPs) that allows the receiver to determine if the received LSP is newer than the one it has stored in its link state database.

sequence number packet (SNP)

Routing: Network protocol data units (PDUs) exchanged between intermediate systems that allow an intermediate system to determine if its link state database is up to date.

server

Component of a distributed application that provides a service, such as a file access or a database query, on behalf of a client.

server entity

Service provider in an asymmetric usage relationship between entities.

service

Task that an application can carry out.

Interface provided by a service element or layer for accessing one or more OSI functions.

service access point (SAP)

Point at which an entity provides a service to a user entity in the layer above; interface at which a service provider delivers services to a service user. Named according to the layer providing the services; for example, transport services are provided at a Transport SAP (TSAP) at the top of the Transport layer.

service class

Set of functional units that supports a global function, such as file transfer, for an FTAM association.

service data unit (SDU)

Data units (messages or blocks of data) passed between entities in adjacent layers on the same open system. The information is made up of a PDU received by a service provider from a service user.

service definition

International standard that describes the capabilities of a layer or the service it provides. Each layer provides a set of services at its upper boundary. Services are available to the next higher layer or, in the case of Application layer services, to the application processes. This set of services is the user's view of the functions performed by a layer.

service element

Portion of an OSI entity that provides services, such as routing, filtering, or data transmission.

service interface

Boundary at which a layer provides a service to the adjacent higher layer in the network architecture; may vary between implementations.

service parameter

Means by which a service user and a service provider exchange information.

service primitive

Message that carries the values of parameters for a specific service and that constitutes the smallest defined interaction between service users and providers. Service users issue service primitives to request services and to accept service requests from their peers on behalf of their own service users. Service providers use service primitives to pass incoming information to their service users. *See also request primitive, indication primitive, response primitive, and confirm primitive.*

service provider

Service element or layer that provides a set of services to the layer immediately above.

service specification

International standard that describes the functions and service parameters of every service of a service provider.

service user

Application program, service element, or layer that uses the services of a service provider.

session connection

Link established by the Session layer between two users of the session service.

Session Control

DNA layer that provides functions for system-dependent, process-to-process communication, name-to-address mapping, and protocol selection; works with DECdns software to translate names to addresses (for nodes and application services) and selects the appropriate protocols to be used at each layer; provides the application interface into the other DNA layers. Compare with **Session layer**.

Session Control application database

Collection of memory-resident Session Control application entities on a system. Each entity stores identifying information about an application; MAIL, CML, and FAL are examples of applications supplied by the DECnet-Plus implementations. The characteristics of user-written applications also are described in Session Control application entities. These Session Control application characteristics include the task name, image file name, and user name.

Session Control layer

See **Session Control**.

Session layer

Layer 5 in the OSI Reference Model; organizes and structures the interactions between pairs of application processes. Compare with **Session Control**.

Session service

OSI service provided by the Transport layer, as defined in International Standard ISO 8326.

set host

Feature of DECnet-Plus for OpenVMS software that allows users of one system to log in to other systems. Comparable to Berkeley UNIX `rlogin`, DECnet/OSI for Digital UNIX software's `dlogin`, and DECnet-Plus' Virtual Terminal (VT) software.

set-valued attribute

DECdns attribute that contains a set of values, rather than a single value.

SFDP

See **short format data packet**.

short format data packet (SFDP)

Short format protocol header used for data packets on point-to-point subnetworks when communicating with DECnet Phase IV nodes. *Contrast with* **long format data packet**.

simple file transfer

NIST FTAM implementation profile that requires the ability to transfer entire files using an unstructured constraint set, that is, files with an FTAM-1 or FTAM-3 document type. Also known as **NBS profile T1**, **SPAG profile A111**, and **CEN/CENELEC profile ENV 41 204**. *See also* **positional file transfer**.

simple name

One element in a DECnet-Plus full name that is delineated by dots. The rightmost simple name in a DECnet-Plus full name.

single-valued attribute

Attribute that expresses a single value rather than a set of values. For example, the `DNS$CTS` attribute expresses the creation timestamp of the DECdns namespace entry with which it is associated. *Contrast with* **attribute set**.

sink

Logical repository used to receive logged events. A logging sink can be a file or console located on an end system.

skew

Time difference between two clocks or clock values.

skulk

Process by which DECdns makes the data consistent in all replicas of a particular directory. DECdns collects all changes made to the master replica since the last skulk completed, and applies them to the replica on the server where the skulk started. It then disseminates the changes from the up-to-date replica to all other existing replicas of the directory. A skulk also performs cleanup functions that include removing soft links that have timed out, removing information about deleted directories, and adding information about newly created directories.

SNAP

See **Subnetwork Access Protocol**.

SNDCF

See **subnetwork dependent convergence function**.

SNP

See **sequence number packet**.

soft link

Pointer that provides an alternate name for an object entry or directory in the DECdns namespace.

source service access point (SSAP)

One-byte field in an LLC frame on a LAN that identifies the link service access point of the sending Data Link layer client protocol.

source task

Task that initiates a transport connection request in a task-to-task communication environment.

SPAG

See **Standards Promotion and Application Group**.

spanning tree

Logical arrangement created by bridges in an extended LAN in which all LANs are connected and there are no loops.

splitting

Process of mapping one transport connection to several network connections.

SSAP

See **source service access point**.

stack

Protocol sequence along with associated address and protocol-specific information. DECnet-Plus systems and applications have multiple stacks, which describe various sets of protocols for communicating with the node. The term **tower** is often used to mean a **stack**.

Standards Promotion and Application Group (SPAG)

Group of European OSI manufacturers that defines a set of profiles for FTAM implementations for its members and publishes these profiles in *Guide to the Use of Standards* (GUS).

static routing

Routing using manually entered information; nonadaptive routing. *Contrast with adaptive routing.*

station

Termination of a communications link of Physical and Data Link layer entities.

status

Network management entity group that contains attributes pertaining to an entity's state. Status attributes may change through direct management commands, such as ENABLE or DISABLE, or by the system itself.

storage attributes

Attributes that deal with the actual storage of a file on a local system, for example, information on accounting and file size.

subaddress

Subaddress that identifies the Internet service provider (particular subscriber line or group of lines) of an X.25 subnetwork; some number of digits of an adjacent system's X.25 subnetwork address; defined by the PSDN provider.

subdomain

See area.

subentity

See child entity.

sublayer

See subnetwork dependent sublayer and subnetwork independent sublayer.

subnetwork

Communications network (within a group of interconnected networks) that is a collection of OSI end systems and intermediate systems in which all systems use a common addressing format and that forms an autonomous whole. Examples: HDLC data link, ISO 8802-3 LAN, and X.25 packet switched network.

Subnetwork Access Protocol (SNAP)

Form of LLC frame used on LANs in which multiplexing uses a 5-byte protocol ID field; allows higher layer protocols that are not international standards to be addressed.

subnetwork address

X.25 PSDN DTE or physical address of a system on a local subnetwork.

subnetwork dependent convergence function (SNDCF)

Set of functions required to enhance the service provided by a particular subnetwork as defined by the CLNS protocol (ISO 8473).

subnetwork dependent sublayer

Sublayer of the Network layer that interfaces to the Data Link layer; masks the characteristics of the different types of subnetworks from the subnetwork independent sublayer so that they all appear the same to the subnetwork independent layer. Some of its functions are: setting up calls on dynamically established data links (DEDs), clearing calls on dynamically established data links, and determining the reachability of adjacencies.

The Data Link layer protocol is not transparent to the subnetwork-dependent sublayer. The subnetwork dependent sublayer provides protocol transparency to the subnetwork-independent sublayer; that is, it masks the characteristics of the different types of subnetworks.

subnetwork independent sublayer

Sublayer of the Network layer that performs all routing functions required to route data to its destination, for example, establishing and maintaining the routing database on each router, extracting and interpreting the route header in packets, determining the best route for packet forwarding, and performing packet forwarding based on the destination address. *See also* **decision**, **update**, and **forward**.

subordinate entity

See **child entity**.

subscriber line

Physical line between a DCE and the local exchange in a public data network.

superior entity

See **parent entity**.

SVC

See **switched virtual circuit**.

switched line

Communications link for which the physical path can vary with each use, such as the dial-up telephone network.

switched virtual circuit (SVC)

Temporary logical association between two DTEs to a PSDN and analogous to a dial-up line; is set up only when there is data to transmit and is cleared when the data transfer is complete.

Dynamically established data link that supports routing over X.25.

switching

Identifying and connecting independent transmission links to form a temporary, continuous path from the source to the destination. *Contrast with* **dedicated line**.

synchronization

Process by which a DECdts entity requests clock values from other systems, computes a new time from these values, and adjusts its system clock to the new time.

synchronization list

List of DECdts servers that a DECdts entity has discovered. The entity sends requests for clock values to the servers on the list.

synchronization points

Markers positioned in data that is being transferred, allowing an application to recover lost or corrupted data.

synchronous disconnect

Disconnect that occurs when a nontransparent task issues a call to terminate I/O operations over a transport connection without deassigning the channel. Thus, the task can use the channel for subsequent I/O operations with the same remote task or a different one.

synchronous link

Communications link using synchronous transmission, in which transmission of a block of data is preceded by a special synchronization sequence. Data is transmitted at a fixed rate with the transmitter and receiver synchronized.

synchronous transmission

Data transmission in which characters are transmitted at a fixed rate. The transmitter and receiver are synchronized, gaining greater efficiency than in asynchronous transmission. Synchronous transmissions send a predetermined group of "sync" characters ahead of a long stream of data. The sync characters enable the communicating devices to synchronize with each other in accordance with time clock at each end. *Contrast with* **asynchronous transmission**.

syntax

Descriptive technique used to convey information.

syntax transformation

Transformation of data back and forth between local data and data that is interpretable by other FTAM systems using a presentation context (abstract syntax/transfer syntax pair).

system

Information-processing unit that supports the Data Link, Network, Transport, and Session layers; single, addressable unit, such as a computer, workstation, or peripheral device. Each system has a unique Network layer address. Also called **node**.

system time

Time value that the operating system maintains according to its reading of the system's hardware clock.

tap

Entry point at which the transceiver connects to an Ethernet cable.

target system

Intended destination of messages.

target task

Task that receives and processes a transport connection request in a task-to-task communication environment.

task

Image running in the context of a process.

task specifier

Information provided to DECnet-Plus software so it can complete a transport connection to a remote task. This information includes the name of the remote node on which the target task runs and the name of the task itself.

task-to-task communication

Communication between two processes using DECnet-Plus or mailboxes. These processes may be on different systems, using different operating systems and languages.

TDF

See **time differential factor**.

Telnet

Virtual terminal protocol in TCP/IP. *See also* **virtual terminal**.

template

Network management — Named collection of module-specific attributes that can be referenced by a client of the module without knowing their individual significance.

terminal emulator

Program that acts as a transparent interface between two ports, making it appear as though a terminal on the local system is connected directly to a remote system.

terminal server

System that handles terminal operations for host nodes on a LAN; can be used to connect terminal users to nodes on the same LAN and to users on nodes located off the LAN. Offload the terminal connection and I/O responsibilities from host nodes, and reduce the number of direct terminal connections to each host, saving substantial power, packaging, and cabling expense.

terminating packet

Packet whose destination is the local node.

test access

Access right that grants DECdns users the ability to test whether an attribute has a specific value without having read access to the attribute. This right is useful for client application programmers: testing for a specific value in a set is more efficient than reading the whole set of values.

test name

Argument of the test entity directive that identifies the test to be performed.

third-party router

OSI intermediate system that implements ISO ES-IS, but does not have the same implementation of the DECnet-Plus IS-IS protocol as DEC WANrouter systems.

throughput

Measure of how much data is sent, or can be sent, between two points in a specified unit of time; often used in either of two contexts:

- Rated throughput, which refers to the bandwidth or capacity of a component.
- Real throughput, which refers to actual measured throughput.

tick

Clock timer interrupt that causes the operating system to increment the system time.

time differential factor (TDF)

Difference between Coordinated Universal Time (UTC) and the time in a particular time zone. Used by DECdts software.

time provider

Hardware device that monitors UTC time and forwards it to a DECdts server.

timestamp

See **creation timestamp (CTS)** and **binary timestamp**.

token

Mechanism for determining which side of an association can initiate certain services; the means by which peer entities agree on which of them can call certain services. Token management is the allocation and exchange of tokens between peer entities.

Mechanism for controlling the use of request primitives for token-controlled services by a specific entity.

topology

Logical arrangement of interconnected systems regardless of their physical locations.

tower

Protocol sequence along with associated address and protocol-specific information. DECnet-Plus systems and applications have multiple stacks, which describe various sets of communications protocols. The term **tower** is often used to mean a **stack**.

TP

See **Transport Protocol**.

TP0

OSI Transport Protocol Class 0 (Simple Class); the simplest OSI Transport Protocol, useful only on top of an X.25 network or other network that does not lose or damage data.

TP2

OSI Transport Protocol Class 2 (Simple Class); same as Class 0, but with the added features of multiplexing and optional flow control.

TP4

OSI Transport Protocol Class 4 (Error Detection and Recovery Class); the most powerful OSI Transport protocol, useful on top of any type of network. OSI equivalent to TCP.

TP server

DECdts server system connected to a time provider.

trace

Software function for analysis of messages that pass between peer entities; provided by Common Trace Facility (CTF), for example. *See also* **Common Trace Facility**.

traffic

Measurement of data flow, volume, and velocity over a communications link.

transceiver

Transmitter-receiver; physical device required in baseband networks that takes the digital signal from a computer or terminal and imposes it on the baseband medium; connects a host interface to a LAN, such as Ethernet.

transfer syntax

Machine-independent form in which data passes between peer entities.

Set of rules used in the formal specification of data that embodies a specific representation of that data, allowing its transfer between open systems. *See also* **syntax transformation**.

transient information

Network management information carried in an operation; is meaningful only while the operation is being performed.

transit packet

Packet arriving from a source node at an intermediate system that is destined for another node. *See also* **route through**.

transport address

Identifier that specifies the location of a transport service access point (TSAP). *See also* **transport service access point**.

transport connection

Link established by the Transport layer between two users of the transport service.

Transport layer

Layer 4 in the OSI Reference Model; provides reliable, transparent transfer of data between end systems, with error recovery and flow control. Important protocol classes supported by this layer are: Class 0 (TP0), Class 2 (TP2), and Class 4 (TP4). *See also* **TP0**, **TP2**, and **TP4**.

Transport Protocol (TP)

OSI protocol that provides reliable end-to-end communication between programs running on different systems. *See also* **TP0**, **TP2**, and **TP4**.

transport protocol data unit (TPDU)

Unit of data, processed by a transport protocol, with transport protocol control information and, possibly, also user data.

transport selector

Field within the domain specific port (DSP) part of an NSAP that indicates to the Routing layer which transport is to receive the data. The selector value consists of two hexadecimal digits (from 0 to F). The DECnet-Plus values are 20 (for NSP transport) and 21 (for OSI transport).

Also called the **NSAP selector**.

Transport service

OSI service that provides a universal communications interface that is independent of the low-level communications medium and protocols; service to high-level protocols at the upper boundary of the Transport layer, as defined in International Standard ISO 8072.

transport service access point (TSAP)

Unique reference that identifies a transport user. When making a transport connection, the transport user provides its TSAP. Messages sent over a transport connection include the TSAP of the transport user to which they are addressed. The transport service uses this TSAP information to route messages to their proper destination. A transport user must be associated with a TSAP in order to use the transport service.

transport service provider (transport service, TS Provider)

Software that provides a transport service. Example: OSI transport.

transport service user (transport user, TS user)

Software running above the Transport layer and using the transport service to communicate with other software running above the Transport layer, possibly on another host.

transport subaddress

Subaddress that identifies the transport service provider of an X.25 network; forms the last two digits of a target system's X.25 destination address.

transport template

Entity that identifies the network service to be used by an OSI transport connection, and the transport characteristics that apply to that connection.

traversal sequence

Set of rules, defined by the FTAM standard, for moving between nodes within a file, beginning with its root node.

traversing

FTAM process of moving to the next node in a sequence of nodes and, optionally, accessing information associated with that node.

tributary

One of multiple points on a multipoint line. *See also* **polling**.

tributary address

Address that a control station uses to poll a tributary. *See also* **polling**.

tributary station

Station on a multipoint line that is not a control station.

true name

Name of an entity that can be directly mapped in the namespace (without resorting to backtranslation or soft links) to an agent address.

TSAP

See **transport service access point**.

TT Device

Device that processes system services relating to user data, which are issued by the X.29 program, and passes them to the NV device.

UA (User Agent)

OSI application process that represents a human user or organization in the X.400 message handling system. Creates, submits, and takes delivery of messages on the user's behalf.

UDP (User Datagram Protocol)

TCP/IP transport protocol; a connectionless transport protocol required by RIP and SNMP. TCP/IP equivalent of OSI CLTP.

UFS

See **ULTRIX File System**.

UI frame

Unnumbered, HDLC, information frame, used to carry data not subject to flow control or error recovery.

ULTRIX File System (UFS)

Set of UFS file-management procedures called by programs to process files.

unconfirmed event report

Event report for which the event sink does not return a response to the request, and the agent might discard the event after transmitting it.

unconfirmed service

Service with nonnegotiable parameters whose functioning is fully controlled by the requesting entity.

unique identifier (UID)

Attribute of all DECdns clearinghouses, directories, soft links, child pointers, and object entries that contains a unique value reflecting the date and time that an entity and a named entity with counters was created. The timestamp consists of two parts — a time portion and a portion with the system identifier of the node on which the name was created. This guarantees uniqueness among timestamps generated on different nodes.

universal time coordination

See Coordinated Universal Time.

unreachable node

Node to which the Network layer has determined that the path exceeds the maximum hops of the network, or the intermediate system either cannot recognize the destination address or access the destination node by a usable path.

unstructured file view

Access context for a file that recognizes only the file-contents data units in a file-access data unit (FADU).

update

Routing process that periodically updates routing information to reflect topology changes, such as circuits going on or off, and changes that network managers make to routing attributes. Level 1 routers send information about adjacent nodes to all other routers in an area in level 1 LSPs, and level 2 routers send information about adjacent areas to all other level 2 routers in level 2 LSPs.

update propagation

Immediate attempt by DECdns software to apply a change to all replicas of the directory in which the change was just made. An update propagation delivers changes in a more efficient and timely way than a skulk, which is the periodic distribution of a whole collection of changes.

update timestamp (UTS)

Attribute (named `DNS$UTS`) that identifies the time when the most recent change was made to any attribute of a particular name in the namespace; for directories, reflects changes made only to attributes that apply to the directory as a whole (not to one of its replicas).

upline dumping

Transferring a copy of a memory image from a target node to a load host node. Some systems, such as DEC WANrouter systems and DECserver terminal servers, automatically upline dump their image upon failure.

user

Defined as:

- Service user. An OSI entity that lies directly on top of another OSI entity and that directly uses its services.
- Person who uses a command or programming interface to access services provided by an entity of the Application layer.
- DECdns — Application or person.

user element

Part of an application entity that invokes the service primitives provided by one or more of the protocols in the Application layer and, optionally, some provided by the Presentation layer.

user entity

Part of an application process that communicates with the OSI protocols.

user interface layers

Device and style layers of the director.

UTC

See **Coordinated Universal Time**.

UTS

See **update timestamp**.

virtual circuit

Independent logical path between two entities for the purpose of exchanging data. Association between two DTEs connected to a PSDN that makes it seem as if a specific circuit were dedicated to them throughout the transmission. In reality, only a logical connection is established: actual, physical circuits are allocated according to criteria such as route availability and network traffic.

virtual file

Unambiguously named collection of structural information. The information in a virtual file includes values for the file attributes that describe a file's properties (such as its file name, size, and file-access structure) and descriptions of a file's file-access data units (FADUs). Used by FTAM software.

virtual filestore

Idealized filestore defined by the FTAM virtual filestore model to serve as an intermediary between real filestores.

virtual filestore model

Common model that describes files (alone or in organized collections) and the possible actions that FTAM software can perform on those files.

virtual terminal

Abstract representation of a real terminal, complete with terminal operations such as reading text from the keyboard, writing text to the screen, and moving the cursor. Using this model, any terminal connected to a system can access any other system regardless of the types of machines in use. *See also* **Virtual Terminal software**.

virtual terminal association

Communication established between two virtual terminal systems, one initiator and one responder. The association allows the two systems to exchange request for information and their corresponding responses. Performed by DECnet-Plus Virtual Terminal software. *See also* **Virtual Terminal software**.

Virtual Terminal software

DECnet-Plus software that allows users and applications of one host to log in to a remote host and then interact as normal terminal users of that host.

- DECnet-Plus Virtual Terminal (VT) software — Supports the ISO Virtual Terminal Protocol for remote logins to other OSI-virtual-terminal-compliant systems, including multivendor systems. Comparable to TCP/IP's Telnet.
- DECnet-Plus for OpenVMS set host feature — Supports remote logins to any other DECnet or DECnet-Plus system.
- DECnet-Plus for Digital UNIX dlogin feature — Supports remote logins to any other DECnet or DECnet-Plus system. Comparable to Berkeley UNIX rlogin.

VisibleString

FTAM string composed of a definable character set that includes alphanumeric characters, punctuation, and graphics.

VMScluster alias

See **cluster alias**.

VT

See **Virtual Terminal software**.

WAN

See **wide area network**.

wide area network (WAN)

Data communications network that covers a wide geographical area, such as a country. *Contrast with* **local area network**.

wide area network device driver (WANDD) software

DECnet-Plus software that links a hardware device and layered software.

wildcard name

DECdns and network management: Name that allows the looking up of groups of entries that have common name characteristics.

window

Ordered set of consecutive data packets authorized to cross the DTE/DCE interface of the logical channel used for a switched virtual circuit (SVC) or permanent virtual circuit (PVC) and for each direction of transmission.

window size

Maximum number of packets that are allowed to be transmitted without an acknowledgment in a PSDN.

write access

Access right that grants users the ability to change DECdns data.

X Recommendations

CCITT documents that describe data communication network standards, for example, the X.25 packet switching standard, X.400 message handling system, and X.500 directory services.

X.3

CCITT recommendation that describes the operation of a packet assembler /disassembler (PAD) and PAD parameters.

X.21

CCITT recommendation that defines the physical and electrical interface between data terminal equipment (DTE) and data circuit-terminating equipment (DCE) of a PSDN; includes OSI layers 1–3 functions.

X.25

CCITT recommendation that defines the interface between the data terminal equipment (DTE) and the data circuit-terminating equipment (DCE) of a PSDN; provides an OSI Network service for PSDNs. *See also* **X.25 Protocol**.

X25 Access module

Defined as:

- X.25 module that defines the interface between the X.25 protocol and applications.
- Network management module that manages the entities that constitute the X25 Access module; interfaces with the X25 Protocol, X25 Client, and X25 Server modules to provide X.25 services and functions as described in the DNA X.25 Access architecture; resides in the DNA Application layer.

X.25 address

Address of a specific DTE that allows network access on X.25 networks.

X25 Client module

Defined as:

- X.25 module that defines how an Access system operates.
- Network management module that manages the X25 Client entity; interfaces with the X25 Access module to establish communications with its X.25 Server system over a DNA Session Control connection using the X.25 GAP protocol; resides in the DNA Application layer.

X.25 gateway access

Facility that allows a user of a DECnet-Plus system not directly connected to an X.25 network to access facilities of that network through an intermediary gateway node.

X.25 line

Line supplied by a PSDN; also called **DTE**.

X.25 network

PSDN that can be accessed by using an interface that complies with the X.25 recommendation.

X.25 Protocol

Connection-oriented network interface protocol; negotiates services for a PSDN.

X25 Protocol module

Defined as:

- X.25 module that defines the X.25 level 3 protocol that is used to exchange packets between a DTE and a DCE; defines the DTEs, PVCs, and closed user groups recognized by an X.25 system.
- Network management module that manages the entities that constitute the X25 Protocol module; provides the X.25 packet level interface into a PSDN; resides in the DNA Network layer.

X25 Relay module

Network management module that manages the entities that constitute the X25 Relay module; interfaces with the X25 Access module to receive an incoming switched virtual call and then make an outgoing call back out through the X25 Access module; resides in the DNA Application layer.

X.25 security

X.25 utility that allows control of use of X.25 systems and protects X.25 systems from unauthorized use.

X25 Server module

Defined as:

- X.25 module that defines how a gateway system communicates with an Access system.
- Network management module that manages the entities that constitute the X25 Server module; interfaces with the X25 Access module to listen for incoming calls for its X.25 client systems, and to make outgoing calls on behalf of its X.25 clients; resides in the DNA Application layer.

X.28

CCITT recommendation that specifies how to control the packet assembler /disassembler (PAD) facilities that a user can select from a terminal; defines procedures for setting up and clearing a link between the PAD and the terminal and for setting up and clearing virtual circuits.

X.29

CCITT recommendation that specifies how to control the PAD facilities that the user can select from the remote packet-mode DTE; defines the procedures for exchanging PAD control information and PAD messages.

X.29 terminal

Remote virtual terminal.

X.75

CCITT recommendation that specifies procedures for communicating between PSDNs.

X.121

CCITT recommendation that defines the format of a DTE address.

X.400

Series of OSI standards, defined by CCITT and approved by it, that specify how networks exchange electronic mail messages.

XID frame

Frame in HDLC used to exchange operational parameters between participating stations.

X/Open Transport Interface (XTI)

Transport service interface, which is independent of the transport provider, that is between the transport user (a networking application or Session layer protocol) and the transport provider; IEEE 1003.8 project.

XTI

See **X/Open Transport Interface**.

G.2 Acronyms

Table 1 shows DECnet-Plus acronyms and other acronyms related to open networking.

Table 1 Acronyms

Acronym	Meaning
ACE	access control entry
ACL	access control list
ACS	access control string
ACSE	Association Control Service Element
AD	ISO Addendum
ADMD	Administration Management Domain
AEP	Applications Environment Profile
AFI	authority and format identifier
AFNOR	French national standards body
AM	active monitor
ANSI	American National Standards Institute
AOW	Asian and Oceania Workshop
APIA	Application Program Interface Association (X.400)
API	application programming interface
ASE	application service element
ASN	Abstract Syntax Notation
ASN.1	Abstract Syntax Notation One
AUI	attachment unit interface
BCUG	bilateral closed user group
BEA	broadcast end-node adjacency
BER	basic encoding rules
BRA	broadcast router adjacency
BRI	Basic Rate Interface (of ISDN)
BSI	British Standards Institute
CALS	Computer-aided Acquisition and Logistic Support
CCITT	International Telegraph and Telephone Consultative Committee
CCR	Commitment, Concurrency, and Recovery
CD	Committee draft (ISO)
CEC	Commission of European Communities
CEN	European Committee for Standardization
CENELEC	European Committee for Electrotechnical Standardization
CEN/CENELEC	Joint European Standards Institution
CBEMA	Computer Business Equipment Manufacturer's Association
CEPT	European Conference of PTTs

(continued on next page)

Table 1 (Cont.) Acronyms

Acronym	Meaning
CL	connectionless mode
CLNP	Connectionless Network Protocol
CLNS	Connectionless-Mode Network Service
CLTP	Connectionless Transport Protocol
CMISE	Common Management Information Service Element
CMIP	Common Management Information Protocol
CONS	Connection-Oriented Network Service
COS	Corporation for Open Systems
COSINE	Cooperation for Open Systems Interconnection Networking in Europe
CSMA-CD	Carrier Sense, Multiple Access with Collision Detection
CT	conformance testing
CTS	creation timestamp
CTS-WAN	Conformance Testing Services — Wide Area Networks (now OSTC)
CUG	closed user group
DA	dynamic assignment
DAD	ISO Draft Addendum
DAF	Distributed Applications Framework
DAM	ISO Draft Amendment
DAP	Data Access Protocol
DCB	data-chain buffer
DCE	data circuit-terminating equipment
DCM	dynamic connection management
DDCMP	Digital Data Communications Message Protocol
DED	dynamically established data link
DIS	ISO Draft International Standard (DP accepted, second technical ballot)
DISP	ISO Draft International Standardized Profile
DLM	data link mapping
DNA	Digital Network Architecture
DNIC	data network identification code
DP	ISO Draft Proposal (has started first technical ballot)
DPDU	data link protocol data unit
DSAP	destination service access point
DSP	domain-specific part
DTE	data terminal equipment
EC	European Commission
ECMA	European Computer Manufacturer's Association
EDI	electronic data interchange format (X.12)

(continued on next page)

Table 1 (Cont.) Acronyms

Acronym	Meaning
EG	Expert Group
EMA	Enterprise Management Architecture
EN	European Norm (profiles from CEN/CENELEC)
ENV	Draft European Standard
ES	end system
ES-IS	end system to intermediate system protocol
ESPRIT	Series of European Cooperative technical development projects
ETCOM	European Testing and Certification for Office and Manufacturing Protocols
ETG	EWOS Technical Guide
ETSI	European Telecommunication Standards Institute
EVD	event dispatcher
EWOS	European Workshop for Open Systems
FADU	file-access data unit
FAL	file access listener
FCS	frame checking sequence
FDDI	Fiber Distributed Data Interface
FDT	function dispatch table
FIPS	Federal Information Processing Standard
FIMS	ISO Project for Forms Information Management System
FNC	Federal Networking Council
FOD	Office Document Format
FPM	FTAM protocol machine
FTAM	File Transfer, Access, and Management
GAP	Gateway Access Protocol
GDMO	Guidelines for the Development of Managed Objects
GOSIP	Government OSI Profile
IDI	initial domain identifier
IDP	initial domain part
IEC	International Electrotechnical Commission
IEEE	Institute of Electrical and Electronics Engineers
IEEE/CS	IEEE Computer Society
INTAP	Interoperability Technology Association for Information Processing, Japan
IONL	Internal Organization of the Network Layer
IPC	Interprocess Communications (function in IEEE 1003.4)
IPSIT	International Public Service Information Technology
IRDS	Information Resource Directory Service

(continued on next page)

Table 1 (Cont.) Acronyms

Acronym	Meaning
IS	(1) ISO International Standard; (2) Intermediate system
ISDN	Integrated Services Digital Network
IS-IS	intermediate system to intermediate system protocol
ISO	International Organization for Standardization
ISODE	ISO Development Environment
ISP	International Standardized Profile
I.T.	Information Technology
ITTF	Information Technology Task Force (ISO/IEC)
ITU	International Telecommunications Union
LAN	local area network
LAP	Link Access Protocol
LAPB	Link Access Protocol Balanced
LCN	logical channel number
LES	Layered Environment Services
LFDP	long-format data packet
LIN	LAN interworking network
LLC	logical link control
LSAP	link service access point
LSP	link state packet
MAC	media access control
MAU	media attachment unit
MEN protocol	Management Event Notification Protocol
MHS	message-handling system
MICE protocol	Management Information Control and Exchange protocol
MMS	Manufacturing Message Systems
MIA/NTT	Multivendor Integration Architecture (Nippon Telephone and Telegraph)
MOM	Maintenance Operations Module
MOP	Maintenance Operations Protocol
MOTIS	Message Oriented Text Interchange System
MTA	message transfer agent (X.400)
MTU	maximum transmission unit
NAS	Network Applications Support
NBS	National Bureau of Standards (now NIST)
NCC	National Computing Centre, Inc.
NCF	Network Computing Forum
NCL	Network Control Language
NCP	Network Control Program

(continued on next page)

Table 1 (Cont.) Acronyms

Acronym	Meaning
NET	network entity title
NICE Protocol	Network Information and Control Exchange Protocol
NIST	National Institute of Standards and Technology (formerly NBS)
NMF	Network Management Forum (OSI/NMF)
NM	network management
NNPC	Network New Products Committee
NP	new project
NPDU	network protocol data unit
NSAP	network service access point
NSCTS	namespace creation timestamp
NSP	Network Services Protocol
NWI	ISO New Work Item
ODA	Office Document Architecture (ISO Standard 8613)
ODIF	Office Document Interchange Format (part of ODA)
ODP	open distributed processing
OIW	OSI Implementors Workshop (NIST)
OMG	Object Management Group
OSAK	OSI Applications Kernel
OSF	Open Software Foundation
OSI	Open Systems Interconnection
OSTC	Open Systems Testing Consortium (formerly CTS-WAN)
OTF	Open Token Foundation
PAD	packet assembler/disassembler
PAGODA	Profile Alignment Group on ODA (consists of EWOS EG on ODA, NIST ODA/ODIF SIG, POSI)
PAR	Proposal and Request (IEEE)
PBX	Public Branch Exchange
PCI	protocol control information
PCTE	Portable Common Tool Environment (ECMA and ESPRIT projects)
PDAD	ISO Proposed Draft Addendum
PDISP	ISO Proposed Draft International Standardized Profile
PDES	Product Data Exchange Specification
PDU	protocol data unit
PICS	Protocol Implementation Conformance Statement
PODA	European Group on ODA
POSI	Promoting Conference for OSI (Japan)
PRMD	Private Management Domain
PSE	packet switching exchange

(continued on next page)

Table 1 (Cont.) Acronyms

Acronym	Meaning
PSDN	packet switching data network
PSTN	public switched telephone network
PTT	Postal, Telephone and Telegraph Authority
PVC	permanent virtual circuit
RD	redirect packet
RDA	remote database access
RISC	Reduced Instruction Set Computer
ROSE	Remote Operations Service Element
RPC	remote procedure call
RPOA	Recognized Private Operating Agency
RTS	Reliable Transport Service
RTSE	Reliable Transfer Service Element
SAB	Standards Activities Board (IEEE/CS)
SAP	service access point
SASE	Specific Application Service Element
SC	subcommittee
SCC	Standards Coordinating Committee (IEEE/CS)
SDO	Standards Development Organization
SDU	service data unit
SFDP	short format data packet
SG	subgroup
SGFS	Special Group on Functional Standardization (part of JTC1)
SGML	Standard Generalized Mark-up Language (ISO 8879)
SIG	special interest group
SIGMA	Software Industrialized Generator and Maintenance Aids (Japan)
SILS	Standard for Interoperable LAN Security
SMDS	Switched Multi-Megabit Data Service
SNAP	Subnetwork Access Protocol
SNDCF	subnetwork dependent convergence function
SNP	sequence number packet
SONET	synchronous optical network
SPAG	Standards Promotion and Application Group
SSAP	session service access point
SVC	switched virtual circuit
SWG	Special Working Group
T1	Standard for High Bandwidth WAN Connections
TC	Technical Committee
TCCC	Technical Committee on Computer Communications (IEEE)

(continued on next page)

Table 1 (Cont.) Acronyms

Acronym	Meaning
TCOS	Technical Committee on Operating Systems (IEEE)
TDF	time differential factor
TFS	Transparent File Service (IEEE 1003.8 Project)
TG	Technical Group
TOP	Technical Office Protocols
TP	Transport Protocol
TPDU	transport protocol data unit
TR	Technical Report
TSAP	transport service access point
TSG	Technical Study Group
TS provider	Transport Service provider
TS user	Transport Service user
UDP	User Datagram Protocol
UID	unique identifier
ULA	upper layer architecture
ULCT	upper layer conformance testing
UTC	Coordinated Universal Time
UTP	unshielded twisted pair
UTS	update timestamp
VSAT	very small aperture terminal (satellite network)
VT	Virtual Terminal (OSI Protocol)
VTAM	Virtual Telecommunications Access Method
WAN	wide area network
WD	Working Draft
WG	Working Group
XTI	X/Open Transport Interface (network stack independent transport)

Table 2 ISO and IEC Acronyms in Transition

Was	Will Become	Meaning
PDAD	PDAM	Proposed Draft Amendment
DAD	DAM	Draft Amendment
AD	AM	Amendment
NWI	NP	New Project
WD	CD	Committee Draft

A

- Abbreviated names, 5–2
- Access
 - network, 5–2
 - proxy, 5–4
 - remote file, 5–3
- Access control
 - null string, 5–4
 - on remote files, 5–4
- Access control string
 - null, 5–4
- Account
 - default DECnet-Plus for OpenVMS, 5–4
 - proxy, 5–4
- ACSE protocol, 2–14
- administrative domain, 3–19
- APPEND command
 - using over the network, 5–6
- application
 - application service element
 - defined, 3–6
 - developing, 1–9
 - distributed overview, 1–10
 - Session Control layer
 - defined, 2–5, 3–10
 - Digital-supplied, 2–5 to 2–6
- Application layer
 - ACSE protocol, 3–8
 - explained, 3–8
 - functions of, 3–2
 - protocols, 3–8
- asynchronous communication
 - dynamic links, 3–25
 - static links, 3–25
 - with DDCMP, 3–25

B

- backward compatibility, 2–8
 - management of Phase IV nodes supported, 2–18

C

- CCITT, 4–1
 - standards related to X.25, 4–7
 - X.25 recommendation, 4–1
 - optional PSDN facilities, 4–12
- CLNS, 3–16, 4–8
 - compared with CONS, 3–16
 - defined, 3–16
 - OSI Transport over ISO 8802-3, 2–4
 - OSI Transport over X.25, 2–4
 - supported by Network layer, 3–16
 - supported by NSP Transport Service, 3–13
 - with Inactive Network Layer Protocol, 3–17
 - with Internet/ES-IS, 3–17
- CMIP
 - access provided by CML, 2–20
 - overview, 1–4, 3–29
 - remote network management, 3–28
- CMISE Application Programming Interface, 2–6
- CML, 1–4
 - for remote management of local system, 2–5
 - overview, 2–20
- Common Trace Facility (CTF)
 - overview, 2–18
 - problem solving, 2–18
 - supported software, 2–18
- Components
 - DECdts, 1–8, 2–10
- Configuration
 - logical names, 5–2
- Configuring DECnet-Plus, 3–29
- CONS, 4–8
 - compared with CLNS, 3–16
 - defined, 3–17
 - OSI Transport over ISO 8802-3, 2–4
 - OSI Transport over X.25, 2–4
 - over X.25, 3–17
 - supported by Network layer, 3–16
- Conversation
 - over the network, 5–10
- COPY command
 - using for remote files, 5–5
- Copying
 - files over the network, 5–5

CSMA-CD
 LAN overview, 3-21
CUG, 4-12

D

DAP
 used by FAL, 2-5
data flow, 3-3
Data Link layer, 3-20
 functions of, 3-2
 support for CSMA-CD, 3-21
 supported protocols, 3-20
Database
 accessing when public, 5-5
DCE, 4-1
DCL
 commands for FTAM operations, 2-15
 remote file handling commands, 5-1
DCL command, 5-3 to 5-10
DDCMP
 asynchronous links, 3-25
 converting to HDLC, 3-26
 overview, 3-24
 synchronous links, 3-25
DECdns
 choosing and dependencies, 1-8
DECdts
 defined, 1-8, 2-10
DECnet Phase V, 1-2
DECnet-Plus
 advantages, 1-2
 compatibility with Phase IV, 2-8
 configuring, 3-29
 data flow, 3-3
 licenses, 1-6
DECnet-Plus for OpenVMS
 default account, 5-4
 general user, 5-1 to 5-10
decnet_dns_register
 overview, 2-19
decnet_loc_register
 overview, 2-19
decnet_migrate
 overview, 2-19
decnet_register
 tool, 2-19
Default
 DECnet-Plus for OpenVMS account, 5-4
DEFINE command
 using with public directories, 5-5
dialogues, 3-5
 association, 3-5
 connections, 3-5
Digital network architecture (DNA), 1-2
Digital UNIX or ULTRIX
 connecting to from an OpenVMS node, 6-2

director
 defined, 3-27
Directory
 accessing when public, 5-5
 displaying remote, 5-5
DIRECTORY command
 using over network, 5-5
distributed network management
 concepts, 3-27
DNA
 layers
 functions, 3-2
 Reference Model, 1-1, 3-4
DNA CMIP
 implemented in DECnet-Plus network
 management, 3-27
DNIC, 4-11
DNS/BIND
 choosing and dependencies, 1-8
DTE, 4-1
 address prefix, 4-11
 character-mode, 4-3
 packet-mode, 4-3
DUMP/RECORDS command
 using over the network, 5-8

E

EDIT command
 for remote file, 5-7
Electronic mail
 See Mail utility
end node, 2-3
end system, 2-3
 defined, 3-18
 end-system-only configuration, 2-8
Enterprise Management Architecture (EMA),
 2-14
entity
 defined, 3-6, 3-27
 network management, 3-27
error
 detection and recovery
 by OSI transport service, 3-14
error detection, 4-6
ES-IS Routing Protocol, 3-15
 defined, 2-8
 illustrated, 3-18
 on LAN, 2-8
Event Dispatcher (EVD)
 overview, 2-19
expedited data, 3-15

F

FAL

- nonzero application, 3-11
- supplied by DECnet-Plus for OpenVMS, 2-5, 3-10

File

- accessing remote, 5-3
- controlling access over the network, 5-6
- copying remote, 5-5
- displaying contents over the network, 5-6
- displaying list of remote files, 5-5
- dumping remote, 5-8
- printing remote, 5-6
- queuing for printing at remote node, 5-6
- specifying remote, 5-3
- specifying remote OpenVMS cluster, 5-3
- transfers over the network, 5-5

File Definition Language

- See FDL

File handling

- network operations, 5-4

File operations, 5-4

File specification

- for remote files, 5-3

flow control

- by OSI transport service, 3-14
- during OSI transport connections, 3-15

FMS, 2-14

FTAM, 2-2

- introduction, 2-15
- journaling feature, 2-15
- using DCL commands, 2-15

FTAM gateway

- overview, 2-16

G

General user

- of network, 5-1 to 5-10

GOSIP

- compliance, 2-1

H

hardware

- device interface for, 2-12

HDLC

- converting from DDCMP, 3-26
- LAPB, 3-24
- overview, 3-23, 3-24
- supported in Data Link layer, 3-20

Heterogeneous network

- remote file operations, 6-1

Host-based routing, 1-3, 2-4, 3-23

I

Inactive Network Layer Protocol, 3-15

initialization script, 3-29, 3-30

Input/output operations

- over network, 5-1

installation

- dependencies and related products, 1-6 to 1-9

intermediate system

- defined, 3-18

Internet

- end-to-end connectionless standard, 3-18

\$IPC

- overview, 2-6 to 2-7
- supported by Session Control, 3-11

IS-IS Routing Protocol, 3-18

- illustrated, 3-18

ISO

8802-3 LAN

- compatible with Ethernet, 3-21
- overview, 3-21
- running OSI Transport class 4 over, 2-4
- running OSI Transport classes 0, 2, and 4 over, 2-4
- support implemented in DECnet-Plus for OpenVMS base system, 2-3

CLNS, 1-5

CONS, 1-5

Internetwork Protocol, 1-5, 3-15

standards

- in DECnet-Plus for OpenVMS, 2-1
- in DECnet-Plus for OpenVMS, 1-4 to 1-6
- OSI Transport Protocol: protocol standard, 3-12
- OSI Transport Protocol: service definition, 3-12
- Presentation layer, 3-8
- Session layer, 3-8
- TP 2, 3-12

L

LAN

- configuration of, 2-8
- OSI transport connection over, 3-15

LAPB, 3-17

- overview, 3-24

LAT/VT gateway

- overview, 2-17

layer

- functions, 3-2

LCGN, 4-10

LCN, 4-9

- for different circuit types, 4-10
- line between DTE and DCE, 4-9

- Link state routing, 1–3
- LLC2, 3–18
- Local namespace
 - choosing and dependencies, 1–7
 - defined, 2–8
 - introduction, 2–8
- Local root, 5–2
- logical link, 3–11
- Logical name
 - using with public directories, 5–5

M

- MAIL command
 - using over the network, 5–9
- MAIL object
 - See Mail utility
- Mail utility
 - network operations, 5–2
- MAIL utility
 - default access account requirement, 5–9
 - network operations, 5–9
- MERGE command
 - using over the network, 5–8
- MIRROR application, 2–5
- MOP
 - functions of, 3–29
- MS-DOS node, 6–5
- Multihomed system
 - defined, 3–23
- multipoint connections
 - supported by WANDD, 3–24, 3–26

N

- Name
 - logical
 - See Logical name
- Name services
 - dependencies, 1–7
 - node name-to-address mapping, 1–7
- Names
 - abbreviated, 5–2
 - alternate, 5–2
- Namespace
 - mix of local and distributed on network, 2–9
- NCL script files, 3–30
- NCP
 - for management of Phase IV nodes, 2–18
 - for managing Phase IV nodes, 3–27
- net\$configure.com
 - FAST DECnet-Plus configuration, 3–29
- NET\$CONFIGURE.COM
 - ADVANCED DECnet-Plus configuration, 3–30
 - BASIC DECnet-Plus configuration, 3–30

- Network
 - access, 5–2
 - network connection
 - defined, 3–18
 - supporting OSI transport connection, 3–15
 - Network Control Language (NCL), 2–14
 - Network layer, 3–15
 - functions of, 3–2
 - support for CSMA-CD, 3–17
 - supporting CLNS, 3–16
 - supporting CONS, 3–16
 - network management
 - agent, 3–27
 - benefits, 2–8
 - command-line interface to, 2–17
 - concepts, 3–27
 - configuration procedure, 3–29, 3–30
 - director, 3–27
 - distributed, 3–27
 - entity, 3–27
 - initialization script, 3–29, 3–30
 - of Phase IV nodes, 2–18
 - service element, 3–27
 - with NCL, 2–17
 - Network operations
 - for the general user, 5–1 to 5–10
 - using Mail utility, 5–2
 - using Phone utility, 5–2
 - Network service
 - CLNS, 3–16
 - CONS, 3–17
 - defined, 3–15
 - explained, 3–15
 - service conversion, 3–16
 - Node synonyms, 5–2
 - node-name-to-address mapping
 - by Session Control layer, 3–10
 - Non-OpenVMS system
 - specifying remote files on, 5–3
 - normal data transfer, 3–15
 - NSDU
 - made from TPDU, 3–14
 - NSP
 - for Phase IV interoperability, 3–10
 - introduction, 2–8
 - in base system, 2–3
 - supported by Transport layer, 3–11
 - Transport Service types, 3–13
 - Null
 - access control string, 5–4

 - Open System Interconnection (OSI), 1–2
 - OpenVMS Cluster, 2–5
 - OpenVMS cluster node
 - alias node identifier, 1–11

- OpenVMS Cluster node
 - configuration, 1–10
- OpenVMS Mail application, 2–5
- OpenVMS Phone application, 2–5
- OpenVMS to MS-DOS network operation, 6–5
- OpenVMS to Digital UNIX or ULTRIX network operation, 6–2
- OpenVMS to RSX (using RMS-based FAL) network operation, 6–8
- OpenVMS VPM application, 2–5
- OSAK, 2–2
 - interface, 2–12
 - programming interface, 2–12
 - software
 - concepts, 2–12
 - protocol machine, 2–14
 - trace utility, 2–14
- OSAK API
 - and OSI reference model, 2–13
 - asynchronous event notification, 2–13
 - parameter block, 2–13
 - passing parameters, 2–13
 - routines, 2–13
 - uses of, 2–13
- OSI
 - See Open System Interconnection
 - application development, 1–9
 - applications
 - OSAK software required, 2–12
 - compliance, 2–1
 - layers
 - functions, 3–2
 - protocol stack management, 2–14
 - Reference Model, 3–1
 - standards
 - defined, 3–1
 - related to X.25, 4–8
- OSI Transport Protocol
 - classes, 3–12
 - supported by Transport layer, 3–11
 - TP 0, 3–12
 - TP 2, 3–12
 - TP 4, 3–12
- OSI transport service
 - functions of
 - concatenation of TPDU's, 3–14
 - error detection and recovery, 3–14
 - flow control, 3–14
 - mapping OSI transport connections to transport users, 3–13
 - recombining data, 3–13
- OSI transport template, 3–14
- transport connection, 3–14
 - flow control during, 3–15
 - initiating host, 3–14
 - kinds of data, 3–15
 - releasing, 3–15
 - responding host, 3–14

- OSI transport service
 - transport connection (cont'd)
 - setting up, 3–14
 - supported by network connection, 3–15
 - using, 3–15
 - transport connection types, 3–15
 - transport user, 3–14
 - initiating, 3–14
 - responding, 3–14
 - TSAP, 3–13
- OSI Transport Service
 - error detection and recovery
 - in Class 4, 3–12
 - explained, 3–13
 - flow control
 - in class 2, 3–12
 - in class 4, 3–12
 - functions of, 3–13
 - multiplexing, 3–13
 - in base system, 2–3
 - introduction, 2–7
 - multiplexing
 - in class 2, 3–12
 - in class 4, 3–12
 - provided by Transport layer, 3–13

P

- packet, 3–18, 4–2
 - control, 4–6
 - formatting of, 4–6
 - level, 4–6
 - routes of, 4–2
 - sequencing of, 4–6
- packet switching
 - acknowledgment, 4–11
 - character-mode DTE, 4–3
 - control packet, 4–6
 - DTE address, 4–11
 - DNIC, 4–11
 - national number, 4–11
 - subaddress, 4–11
 - DTE parameters, 4–12
 - LCGN, 4–10
 - LCN, 4–9
 - for different circuit types, 4–10
 - overview, 4–1
 - packet
 - call request, 4–10, 4–11
 - control, 4–6
 - controlling flow of, 4–11
 - packet assembly/disassembly, 4–3
 - packet header, 4–6
 - LCN, 4–6
 - packet type identifier, 4–6
 - packet-mode DTE, 4–3
 - PAD
 - optional facilities, 4–12

- packet switching
 - PAD (cont'd)
 - parameter, 4-12
 - PAD functions, 4-3
 - window size, 4-11
 - X.121 standard, 4-7
 - X.28 standard, 4-7
 - X.29 standard, 4-7
 - X.3 standard, 4-7
 - X.75 standard, 4-7
 - PADs, 4-3
 - directly connected, 4-4
 - for public use, 4-4
 - host-based, 4-4
 - multiple terminal connection, 4-4
 - provided by PSDN, 4-4
 - public vs private PSDNs, 4-2
 - use of modems, 4-4
 - Password
 - avoiding use in file specification, 5-4
 - PDU
 - defined, 3-4
 - generated by service primitives, 3-6
 - Phase IV
 - compatibility with DECnet-Plus, 2-8
 - node
 - remote management of, 2-18, 3-27
 - PHONE command
 - using over the network, 5-9, 5-10
 - Phone utility
 - network operations, 5-2
 - PHONE utility
 - network operations, 5-9
 - PHONE Utility
 - network operations, 5-10
 - Physical layer, 3-26
 - functions of, 3-2
 - support for CSMA/CD, 3-26
 - Polycenter Framework Management System (FMS), 2-14
 - port
 - Session layer, 3-11
 - Transport layer, 3-11
 - PPCI
 - traced with OSAK trace utility, 2-15
 - Presentation layer
 - functions of, 3-2
 - provided services, 3-8
 - standards, 3-8
 - Printing
 - files over the network, 5-6
 - PRINT/REMOTE command
 - using for remote files, 5-6
 - Process
 - remote, 5-4
 - programming interface
 - DNA
 - in DECnet-Plus for OpenVMS base system (cont'd)
 - supported by Session Control, 3-11
 - OSI
 - supported by Session layer, 3-11
 - Protection
 - of remote files, 5-4
 - protocol
 - defined, 3-4
 - supported by Data Link layer, 3-20
 - protocol machines, 2-14
 - defined, 3-4
 - protocol tower
 - built by Session Control, 3-9
 - DNA\$Towers attribute, 3-10
 - information used by Session Control, 2-6
 - protocol tower set, 3-9
 - Proxy
 - account, 5-4
 - PSDN, 4-1
 - concepts, 4-1
 - data transmission speed
 - dial-up line, 4-9
 - leased line, 4-9
 - dial-up line connection, 4-9
 - flow control on, 4-11
 - leased line connection, 4-9
 - logical channels, 4-9
 - making a call, 4-9
 - physical connection, 4-9
 - private network, 4-13
 - public network, 4-13
 - switched (telephone) line, 4-9
 - testing procedure, 4-13
 - virtual circuits
 - between logical channels, 4-9
 - PVC, 4-10
 - SVC, 4-10
 - PSE, 4-1
 - Public database
 - accessing, 5-5
 - Public directories
 - accessing, 5-5
 - PVC, 4-10
- Q**
-
- \$QIO
 - overview, 2-7
 - supported by Session Control, 3-11
 - Queuing remote file
 - for printing, 5-6
 - Quotation marks
 - in remote file specifications, 5-3

R

Remote file

- See also Remote file access
- copying, 5-5
- displaying contents, 5-8
- printing, 5-6
- specifying, 5-3
- specifying on non-OpenVMS systems, 5-3

Remote file access, 5-3

- controls, 5-4

Remote file operation

- general DECnet restrictions, 6-1
- OpenVMS to MS-DOS, 6-5
- OpenVMS to Digital UNIX or ULTRIX, 6-2
- OpenVMS to RSX (using RMS-based FAL), 6-8

Remote file operations

- heterogeneous network, 6-1

Remote process, 5-4

retransmission

- by CSMA/CD, 3-21
- by OSI transport service, 3-3, 3-14
- timer, 3-14

RMS

- defined, 2-15

Root

- local, 5-2

ROSE protocol, 2-14

routing

- adaptive benefits, 1-9
- defined, 3-18
- introduction, 2-8
- requirements, 1-8, 2-3
- routing circuit, 3-19
- routing port, 3-19
- setting up interphase routing, 2-19

Routing

- host-based, 1-3, 2-4, 3-23
- link state, 1-3

RSX node, 6-8

S

SAP

- defined, 3-6
- selector, 3-7

Save set

- in network operations, 5-8

SDU

- defined, 3-4

service

- application service element
 - defined, 3-6
- defined, 3-6
- provider
 - defined, 3-6
 - originates primitives, 3-6

service (cont'd)

- service primitives, 3-6
- user

- connections established between, 3-5
- defined, 3-6
- originates primitives, 3-6

Session Control layer, 3-9

- address selection, 3-10
- application
 - Digital-supplied, 2-5 to 2-6
- defined, 2-5
- node-name-to-address mapping, 3-10

Session layer

- functions of, 3-2
- provided services, 3-8
- standards, 3-8

SET HOST command, 5-3

Short forms of names, 5-2

SPCI

- traced with OSAK trace utility, 2-15

SSDU

- traced with OSAK trace utility, 2-15

standards

- compliance with CCITT, 4-7
- compliance with OSI, 4-8
- optional PSDN facilities, 4-12
- Presentation layer, 3-8
- protocol specification, 3-4
- service definition, 3-4
- Session layer, 3-8

Startup logical names, 5-2

Startup/configuration logical names, 5-2

subnetwork

- broadcast, 3-15
- connected by Network layer, 3-15
- defined, 3-18
- in OSI administrative domain, 3-19
- nonbroadcast, 3-15
 - permanent link, 3-15
- OSI transport connection over, 3-15
- routing requirements, 2-3
- with X.25 communications, 2-4, 4-8

Subnetwork

- OSI transport connection using CLNS with
 - Internet/ES-IS, 3-17

SVC, 4-10

synchronous communication

- with DDCMP, 3-25

Synonyms

- node, 5-2

T

Task application, 2-5

telnet command

- accessing remote OSI system, 2-17

- Telnet/VT gateway
 - introduction, 2-17
- template
 - OSI Transport, 3-14
- Time Service, 1-8, 2-10
- tool
 - overview, 2-17 to 2-20
- TPDU
 - acknowledge TPDU, 3-14
 - concatenating, 3-14
 - OSI transport connection request, 3-14
 - OSI transport disconnect, 3-15
- Transferring
 - files over the network, 5-5
- Transport layer, 3-11
 - functions of, 3-2
- TSAP
 - defined, 3-13
- TSDU
 - traced with OSAK trace utility, 2-15
- TYPE command
 - using over network, 5-5

V

- virtual circuits, 4-6
 - controlled by PADs, 4-3
 - PVC, 4-10
 - SVC, 4-10
 - types, 4-10
- Virtual Terminal
 - gateways overview, 2-16
 - overview, 2-16
- VT/LAT gateway
 - overview, 2-17
- VT/Telnet gateway
 - introduction, 2-17

W

- WANDD
 - overview, 2-12
 - support of tributary multipoint connections, 3-24, 3-26
- Wildcard character
 - in file specifications for network copying operations, 5-6

X

- X.121
 - CCITT standard, 4-7
- X.25
 - CCITT recommendation, 4-1
 - defined, 4-1
 - electrical connection, 4-7
 - flag character, 4-6
 - frame check sequence, 4-6
- X.25 (cont'd)
 - frame header, 4-6
 - gateway, 3-17
 - network defined, 4-1
 - OSI transport connection over, 3-15
 - point-to-point link
 - OSI transport connection over, 3-15
 - protocol level
 - packet level, 4-6
 - physical level, 4-7
 - related CCITT standards, 4-7
 - related OSI standards, 4-8
 - software overview, 2-10
- X.25 protocol level
 - frame level, 4-6
- X.28
 - CCITT standard, 4-7
- X.29
 - CCITT standard, 4-7
- X.3
 - CCITT standard, 4-7
- X.75
 - CCITT standard, 4-7