Software
Product
Description

PRODUCT NAME: **VAX DEC/MAP, Version 3.0** **SPD 27.66.03**

DESCRIPTION

VAX DEC/MAP is a network communication product. It implements the Network through Application Layers of the Open Systems Interconnection (OSI) reference model. When combined with prerequisite hardware, VAX DEC/MAP will interoperate with other systems supporting the Manufacturing Automation Protocol (MAP), Version 3.0 specification.

VAX DEC/MAP is a VMS layered product.

The MAP network architecture specifies the protocols and services for the Physical, Data link, Network, Transport, Session, Presentation and Application Layers.

The Physical and Data link layers are implemented in the prerequisite hardware, which are a fully integrated communications options not requiring any additional or non-Digital hardware.

VAX DEC/MAP will connect Q-bus VAX processors to a MAP network using the same IEEE 802.4 token bus hardware (DTQNA) developed for VAX DEC/MAP V2.0. VAX DEC/MAP V3.0 will also connect VAX UNIBUS, BI-Bus, and Q-bus processors to IEEE 802.3 networks. (Ethernet is not supported.)

VAX DEC/MAP software implements the Network Layer through Applications Layers, layers 3 through 7.

VAX DEC/MAP provides the network and transport layers through the use of the VAX OSI Transport Service (VOTS) V2.0 product which is packaged with this product.

VAX DEC/MAP also includes the session and presentation layers.

The Application Layer of VAX DEC/MAP provides file transfer, access, and management (FTAM - through the use of the VAX FTAM V1.1 product which is packaged with this product), an application programming interface (MAPI - ACSE/Presentation Services), Network Management, and directory services (MDS).

MMFS is no longer part of the MAP specification and will not be included in this product. Manufacturing Message Specification (MMS) is not included in this product.

The conformance of VAX DEC/MAP software to OSI and MAP standards is specified in Appendix A attached to this SPD.

SOFTWARE CAPABILITIES

VAX DEC/MAP allows the VAX it is running on to communicate via an IEEE 802.3 or IEEE 802.4 Local Area Network (LAN).

The user of VAX DEC/MAP is provided with several application interfaces for communicating with other systems on the network, and a network management facility for control of the local node.

Network Management

VAX DEC/MAP Network Management is designed to provide local network management for the VAX DEC/MAP host. The VAX DEC/MAP Network Management functions consist of management for layers 1 through 7, directory services, tracing, logging, and local events. The user interface is a command line interface provided by the MAPCL program.

VAX DEC/MAP Network Management has the following functions:

- Commands to monitor and modify information within the databases.
- Commands to control the VAX DEC/MAP host's local network capabilities such as enable/disable transport and transport connection.
- Commands to create access control information, which can be used by VOTS to restrict access to transport users passively associated with Transport Service Access Points (TSAP).
- Local event logging mechanisms for Application, Presentation, Session, Transport, Network, Data Link, and Physical Layers, and also for Network Access Points (NAP) and Subnetwork Access Points (SNAP).
- A software loopback test at the Transport Layer that enables users to loop data from one VAX DEC/MAP host to another in the VAX DEC/MAP network. Also provides a verification test that can be used to show the operability of a VAX DEC/MAP host.

November 1989
AE-JC15D-TE

- Access to VAX DEC/MAP Directory Service (MDS) for the storage and management of directory objects.
- Tracing facilities for Presentation, Session, Transport, and Network Layer data units. There are two modes of trace operation: Real-time tracing (collecting and analyzing data directly) and Non-interactive tracing (collecting data and analyzing it later).
- A symbol substitution facility that allows the user to create, delete, use and review logical strings in place of longer or complex strings found in a command.
- A logfile facility that allows the user to create a network management command file to store the journal of subsequent commands. Such logfiles are to be created such that they can subsequently be used as scripts as invoked by the DO command.

VAX DEC/MAP Application Layer

The VAX DEC/MAP Application Layer provides the user interface to the communication services provided by the MAP network. It consists of File Transfer, Access and Management (FTAM), Association Control Service Elements (ACSE), and VAX DEC/MAP Directory Service (MDS) via the VAX DEC/MAP Application Programming Interface (MAPI).

File Transfer, Access and Management (FTAM)

The VAX FTAM V1.1 product is packaged with VAX DEC/MAP to provide the ISO FTAM functions. VAX FTAM provides users the ability to perform file transfer and limited file management between a VMS system and any other vendor's system which supports a compatible version of FTAM. It supports FTAM access as both an 'Initiator' and 'Responder'. The initiator component allows users on the local system to request file transfer to or from a file on a remote FTAM system. The responder component allows a user or application on a remote FTAM system to access files on the local VMS system. The VAX FTAM V1.1 product conforms to the MAP V3.0 specification of FTAM.

The user accesses FTAM services through the DCL command line interface which supports basic OSI file transfer and limited management capabilities. Utilities to support the APPEND, COPY, DELETE, DIRECTORY, RENAME functions are provided for FTAM files. These functions are implemented in a manner consistent with the standard VMS utilities. The "/OSI" qualifier must be used with the DCL commands to invoke FTAM as shown below:

```
$ APPEND/OSI
$ COPY/OSI
$ DELETE/OSI
$ DIRECTORY/OSI
$ RENAME/OSI
```

FTAM can access and transfer files containing both binary and ASCII data supporting FTAM-1, FTAM-2 and FTAM-3 document types. For FTAM-1 document types, the string significance parameter values supported are fixed, variable and none. For FTAM-2 document types, the string significance parameter value supported is none. For FTAM-3 document types, the string significance parameter values supported are fixed and none.

VAX DEC/MAP Application Programming Interface (MAPI)

The MAPI Applications Programming Interface is provided to give the user direct access to the communication interfaces of ISO ACSE (Association Control Service Elements) and limited access to ISO Presentation. This is the primary interface used by customers to develop their own applications on the VAX DEC/MAP system. It replaces the CASE interface available in previous versions of VAX DEC/MAP.

The interface provides for peer-to-peer communications between application processes running in MAP network systems. It allows for:

- Establishing an association with a peer application
- Accepting or Rejecting an association request
- Exchanging data with a peer application
- Releasing or Aborting an association

Both synchronous and asynchronous calls are supported.

VAX DEC/MAP Directory Services (MDS)

The VAX DEC/MAP Directory Service (MDS) provides for the storage, access and management of directory objects. MDS implements a local directory service and a Directory User Agent (DUA) so that directory functions are available both in a local context and when the VAX DEC/MAP system is part of a centralized or distributed directory configuration.

Full functionality of MDS may be accessed via the MAPCL command-line interface. Additionally, the directory lookup (name-to-address resolution) capability is also made available via a synchronous programming interface routine.

DUA bootstrap information (addressing information about the available DSAs on the MAP network) is entered at installation time. However, this information may be updated at a later stage using MAPCL with the "prohibit remote" service option.

Protocol Layers

Association Control Service Element (ACSE)

The ACSE provides services for establishing and releasing application associations. These services are intended to be applicable to a wide range of application process communication requirements. ACSE services are accessed via the MAPI interface.

Presentation

The purpose of the Presentation layer is to negotiate transfer syntaxes, and perform local syntax to transfer syntax conversion, for the exchanging of data. VAX DEC/MAP supports only the built-in and null syntax handlers; applications must perform their syntax conversion at the application level.

The Presentation layer can be accessed via the MAPI interface.

Session

The Session layer supports an orderly dialogue between presentation entities using full-duplex functionality, checkpointing and resynchronizing capabilities. The MAPI, FTAM, and MDS services make use of VAX DEC/MAP Session services. VAX DEC/MAP does not have an exposed Session-level programming interface.

Transport and Network Layer Implementation (VOTS)

The VOTS implementation is comprised of the OSI Transport and Network layer. The Transport layer is the fourth layer in the OSI seven-layer Basic Reference Model. It provides reliable end-to-end transfer of data between cooperating applications on open systems. The upper layers use the services of VOTS and the VOTS Programming interface is available for direct user program access. Note that MAPI and FTAM services will not negotiate the use of Transport EXPEDITED data. It is only available via applications written to the VOTS programming interface.

The Network layer is the third layer in the seven-layer OSI Basic Reference Model. The VOTS implementation includes the Internet Protocol (ISO 8473). The Internet Protocol allows for communication between Transport layer entities on the same or different subnetworks. This implementation only has the capabilities of an End System, that is, it cannot forward data received from another system; it can only send its own data units.

The VAX DEC/MAP transport/internet implementation may be connected to one or more ISO 8802-3 (IEEE 802.3) local-area subnetworks. It may also be connected to an ISO 8802-4 (IEEE 802.4) local-area subnetwork and allows concurrent access to both types of subnetworks. Note that VAX DEC/MAP does not support X.25 packet-switched subnetworks.

Data Link And Physical Layer

Implementation of these layers is provided by the requisite hardware supplied by Digital Equipment Corporation.

VAX DEC/MAP Installation Verification Procedure (IVP)

The VAX DEC/MAP IVP performs a series of tests to verify the proper installation and functionality of the VOTS, FTAM, and other VAX DEC/MAP components.

The Transport IVP allows for setting up both local (LOOPBACK) and remote (to another VMS system running VAX DEC/MAP) Transport connections in order to test the configuration of the Transport implementation.

The FTAM IVP exercises the COPY/OSI, RENAME/OSI, DELETE/OSI, and DIR/OSI commands on the local FTAM responder of the VAX DEC/MAP system.

The MAPI IVP conducts a series of associations, data transfers, and orderly releases in local loopback mode.

The MAPCL IVP verifies access to various databases.

INSTALLATION

Only experienced customers should attempt installation of this product. Digital recommends that all other customers purchase Digital's Installation Services. These services provide for installation of the software product by an experienced Digital Software Specialist.

Customer Responsibilities

Before installation of the software, the customer must:

- Previously have installed all prerequisite software.
- Obtain, install and demonstrate as operational any modems and other equipment and facilities necessary to interface to Digital's communication equipment.
- If this is a second Digital VAX DEC/MAP installation, designate and provide access to a VAX DEC/MAP host that has previously been installed by Digital on this certified channel for subsequent verification of installation and connectivity in a multi node network.

Prior to the product being placed on a DEC Service contract, the customer must ensure that a channel certification for the Digital node has been completed.

Delays caused by any failure to meet these responsibilities will be charged at the then prevailing rate for time and materials.

HARDWARE REQUIREMENTS

VAX, MicroVAX or VAXstation configuration as specified in the System Support Addendum (SSA 27.66.03-x).

SOFTWARE REQUIREMENTS

VMS Operating System.

Refer to the System Support Addendum (SSA 27.66.03-x) for availability and versions of required software.

ORDERING INFORMATION

Software Licenses: QL-VFZ**-**
Software and Documentation: QA-VFZAA-H*
Documentation Only: QA-VFZAA-GZ
Software Product Services: QT-VFZA*-**

* Denotes variant fields. For additional information on available licenses, services and media, refer to the appropriate price book.

SOFTWARE LICENSING

This software is furnished under the licensing provisions of Digital Equipment Corporation's Standard Terms and Conditions. For more information about Digital's licensing terms and policies, contact your local Digital office.

LICENSE MANAGEMENT FACILITY SUPPORT

This layered product supports the VMS License Management Facility. License units for this product are allocated on a CPU-capacity basis.

For more information on the License Management Facility, refer to the VMS Operating System Software Product Description (SPD 25.01.xx) or the VMS Operating System documentation set. For more information on Digital's licensing policies, contact your local Digital office.

SOFTWARE PRODUCT SERVICES

A variety of service options are available from Digital. For more information, contact your local Digital office.

SOFTWARE WARRANTY

Warranty for this software product is provided by Digital with the purchase of a license for the product as defined in the Software Warranty Addendum of this SPD.

WARRANTY LIMITATIONS

Digital has produced this product as a MAP Version 3.0 variant and warrants the functionality and proper operation with other Digital VAX DEC/MAP systems. Digital has produced this product according to Protocol Specifications produced by the International Organization for Standardization (ISO) and the Manufacturing Automation Protocol (MAP) Version 3 specification as defined in the Conformance to Standards appendices of this SPD.

Digital warrants this product to conform to these Protocol specifications, as described by these Standards.

To minimize the risk of interworking problems, Digital has tested this product against selected other MAP V3.0 implementations. However, Digital cannot assume responsibility for interworking problems due to:

- Other vendors' non-conformance to the International Standards that VAX DEC/MAP implements
- Ambiguities in the International Standards implemented by VAX DEC/MAP
- Flaws in the protocol design appearing in the International Standards implemented by VAX DEC/MAP

Assistance requested of Digital for problems resulting from any of the above listed errors will result in all associated service calls being billed at the prevailing per call rates.

SERVICE STANDARDS

The internal components of VAX DEC/MAP have been implemented taking into account the following International Standards for Service Definitions.

Application Layer Service Standards

ISO 8571 Part 3: Information Processing Systems - Open Systems Interconnection - File Transfer, Access, and Management (FTAM).

ISO 8649: Information Processing Systems - Open Systems Interconnection - Service Specification for the Association Control Service Element.

Presentation Layer Service Standard

ISO 8822: Information Processing Systems - Open Systems Interconnection - Connection Oriented Presentation Service Definition.

Session Layer Service Standard

ISO 8326: Information Processing Systems - Open Systems Interconnnection - Basic Connection Oriented Session Service Definition, including Addendum 2 incorporating Session Unlimited User Data.

Transport Layer Service Standard

ISO 8072: Information Processing Systems - Open Systems Interconnection - Transport Service Definition.

Network Layer Service Standards

ISO DIS 8348: Information Processing Systems - Data Communications - Network Service Definition.

ISO DIS 8348/DAD1: Information Processing Systems - Data Communications - Network Service Definition - Addendum 1: Connectionless Mode Transmission.

ISO DIS 8348/DAD2: Information Processing Systems - Data Communications - Network Service Definition - Addendum 2: Network Layer Addressing.

APPENDIX A

Conformance To Standards

This appendix defines the conformance of VAX DEC/MAP product to International Organization for Standardization (ISO) international specifications for protocol standards. It is organized by layers of the OSI model application (covering FTAM, Directory Service, and ACSE), presentation, session, transport, and network.

Relevant ISO service standards are also listed in the latter portion of this appendix. However, VAX DEC/MAP does not claim conformance to any service standards; they have been included for completeness.

Protocol Standards

Application Layer Standards

ISO 8571 Part 4: Information Processing Systems - Open Systems Interconnection - File Transfer, Access, and Management (FTAM).

Conformance statement as required in ISO 8571-4 (Clause 22.1, statement requirements).

1.1 Supported Service Classes

VAX FTAM V1.1 supports the following services classes:

TABLE 1: Supported Service Classes

Service Class	**Description (as detailed in 8571)**
File Access	Allows access to and manipulation of file access data units within files.
Transfer	Allows transferring whole files or parts of files; optionally, allows creating and deleting files and reading file attributes.
Management	Allows reading file attributes and creating and deleting files independently of file-transfer capabilities; optionally allows changing file attributes.
Transfer & Management	Allows transferring whole files or parts of files; allows creating and deleting files and reading file attributes; optionally allows changing file attributes.
Unconstrained	Allows negotiating for available functions based on functional units rather than service classes; permits support of nonstandard applications.

1.2 Supported Functional Units

The VAX FTAM V1.1 implementation supports the following functional units:

TABLE 2: Supported Functional Units

Functional Unit	**Description (as detailed in 8571)**
Kernel	Supports basic file services for the establishment of the FTAM and file selection regimes.
Read	Supports transfer of data from the responder to the initiator.
Write	Supports transfer of data from the initiator to the responder.
Limited file management	Supports file creation, deletion and reading of file attributes.
Grouping	Allows grouping of service primitives in a single exchange, thus allowing the efficient establishment of nested regimes.
File access	Allows location and deletion of individual file access data units. Optionally, when combined with read and write functional units, it allows reading and inserting file access data units within files.
Enhanced file management	Supports the changing of file attributes.

1.3 Supported Values of Recovery-Mode Parameters

VAX FTAM V1.1 supports an FTAM quality of service of zero, which indicates that the recovery/restart error-recovery have not been unimplemented.

1.4. Capability of System to Act As Initiator and/or Responder

The VAX FTAM V1.1 implementation acts as either or both initiator and responder.

1.5 Capability of System to Act As Sender and/or Receiver

The VAX FTAM V1.1 implementation acts as either or both sender and receiver.

1.6 Supported Optional Attribute Group(s)

VAX FTAM V1.1 supports the following optional attribute group:

Attribute Group	**Description (as detailed in 8571)**
Storage attribute group	Comprises attributes that deal with storing a file on the local system

1.7 Supported Optional Components of Supported Attributes

The only supported attribute with any optional component is the contents type attribute for which VAX FTAM supports only document-type names.

1.8 Range of Values Supported for Each Supported Attribute

This section documents the range of values that the VAX FTAM responder supports for supported file attributes:

TABLE 3: Value Ranges of File Attributes Supported by Responder

Attribute	**Supported Value Range**
File name	A single element that contains one or more of the components of RMS file designations, as follows: (device:[directory]file-name.file-ext;version). Except for the file name or file name delimiter(.), all the components are optional.
Permitted actions	Actions available: read, insert, replace, extend, erase, read attribute, change attribute, and delete file FADU-ID groups available: traversal
Contents type	Document-type name (FTAM-1, FTAM-2, or FTAM-3)
Storage account	"No value available"
Date and time of creation	RMS creation date
Date and time of last modification	RMS revision date
Date and time of last read access	"No value available"
Date and time of last attribute modification	"No value available"
Identity of creator	"No value available"
Identity of last modifier	"No value available"
Identity of last reader	"No value available"
Identity of last attribute modifier	"No value available"
File availability	"Immediately available"
File size	Unbounded integer representing a whole number of bytes
Future file size	Unbounded integer representing a whole number of bytes

1.9 Supported Document Types

The VAX FTAM V1.1 product supports the following ISO 8571 document types:

TABLE 4: Document Types Supported by VAX FTAM V1.1

Name	**File Type**
FTAM-1	Unstructured text
FTAM-2	Sequential text
FTAM-3	Unstructured binary

1.10 Hierarchical File Model Support

VAX FTAM V1.1 supports a subset of the hierarchical file model that is defined by the constraint sets used by the supported document types. Those constraint sets and the associated document types are identified in Table 5.

TABLE 5: Supported Constraint Sets

Descriptor	**Identifier**	**Associated Document Type(s)**
FTAM unstructured	{ISO standard 8571 constraint-set (4) unstructured (1)}	FTAM-1 FTAM-3
FTAM sequential flat	{ISO standard 8571 constraint-set (4) sequential flat (2)}	FTAM-2

Table 6 lists the available file store actions and access contexts for each constraint set.

TABLE 6: Supported Filestore Actions and Access Contexts

Descriptor	**Optional Actions**	**Access Contexts**
FTAM unstructured	Read, replace, extend, erase	Unstructured all DU (UA)
FTAM sequential flat	Read, insert, erase, locate	Unstructured all DU (UA), Flat all DU (FA)

1.11 Changes to Files Between FTAM Protocol Exchanges

This section documents the circumstances, if any, where changes occur in a file's existence, contents, or attributes between accesses using the FTAM protocol. A file in the VMS real filestore is controlled by the VMS Record Management Services (RMS). Therefore, any VMS user possessing access to a file's storage account or privileges to write or delete a file can potentially alter the file.

1.12 Unavailable Changes to Files by FTAM Protocol Exchanges

This section documents the circumstances, if any, where modifications by FTAM protocol exchanges to a file's contents or attributes are subsequently unavailable for use. Ideally, all FTAM-generated modifications on an RMS file occur, and subsequently will be available for use. Potentially, however, a problem such as power loss or a disk-drive failure can prevent modificiations on a target file.

1.13 Support for Parameters

This section documents what optional parameters are supported and the range of values for all supported parameters. This discussion defines parameter support as the ability of the initiator and/or responder to pass or accept values for a parameter. Parameter support does not necessarily imply that the VAX FTAM responder support the corresponding attributes or functions.

Table 7 lists the supported protocol parameter(s) for each service. For each parameter, the table indicates whether support is mandatory or optional. For each optional parameter, the table indicates whether the VAX FTAM implementation supports the parameter for the request and/or the response service primitives. The table also summarizes the acceptable range of values for each supported parameter. Generally, the VAX FTAM minimum ranges, which are the same as for equivalent VMS data types, meet or exceed the minimum ranges specified in ISO 8571. Where only a specific value or a limited range of values is acceptable, the table lists the specific value or range.

Support of a parameter on the request primitive means that the VAX FTAM initiator can pass on a value within the acceptable range. Support of a parameter on the response service primitive means that the VAX FTAM responder can return a value within the acceptable range. However, the ability to return a value does not necessarily imply support for the underlying attribute or function. For example, for the recovery mode parameter, the responder can return a value of zero, which indicates that recovery mode is unsupported.

TABLE 7: Protocol Parameters

Parameter	*ISO specif- ication	Parameter support for service primitives		Range of acceptable values
		Request	Response	
F-INITIALIZE Supported Parameters				
State result	m	-		Integer (0 or 1)
Action result	m	-		Integer
Protocol ID	m			Bit string (bit 0 set to 1)
Implementation ID	o	n	yes+	String
Presentation context management	m			Boolean
Application context name	o	yes	yes	Object identifier, "ISO FTAM"
Service class	m			Bit string
Functional units	m			Bit string
Attribute groups	m			Bit string
Shared ASE information	o	no	no	-
FTAM quality of service	m			Integer (0)
Communications quality of service	m			Information as defined in ISO 8823
Contents type	o	yes	yes	Sequence of choice between document types and abstract syntax names
Initiator identity	o	yes	-	String
Account	o	yes	-	String
Filestore password	o	yes	-	String
Diagnostic	o	-	yes	Sequence of information as defined in ISO 8571-3

- \+ The VAX FTAM responder can accept a value, but the VAX FTAM initiator does not generate one.
- * Requirement key: m = Mandatory parameter; o = Optional parameter
- ! Support key: yes = Optional parameter supported; no = Optional parameter unsupported; blank space ( ) = Mandatory parameter (supported); - = Not applicable.

TABLE 7 (Cont.): Protocol Parameters

Parameter	*ISO specif- ication	Parameter support for service primitives		Range of acceptable values
		Request	Response	
F-TERMINATE Supported Parameters				
Shared ASE information	o	no	no	-
Charging	o	no	no	-
F-U-ABORT Supported Parameters				
Action result	m	-		Integer
Diagnostic	o	-	yes	Sequence of information as defined in ISO 8571-3
F-SELECT Supported Parameters				
State result	m	-		Integer (0 or 1)
Action result	m	-		Integer
Attributes	m			Sequence of information as defined in ISO 8571-3
Requested access	m		-	Bit string
Access password	o	yes	yes	Sequence of information as defined in ISO 8571-3
Concurrency control	o	yes	yes	Sequence of information of defined as ISO 8571-3
Shared ASE information	o	no	no	-
Account	o	yes	-	String
Diagnostic	o	-	yes	Sequence of information as defined in ISO 8571-3

* Requirement key: m = Mandatory parameter; o = Optional parameter.

! Support key: yes = Optional parameter; no = Optional parameter unsupported; blank space ( ) = Mandatory parameter (supported); - = Not applicable.

TABLE 7 (Cont.): Protocol Parameters

Parameter	*ISO specification	Parameter support for service primitives		Range of acceptable values
		Request	Response	
F-DESELECT Supported Parameters				
Action result	m	-		Integer
Charging	o	-	no	-
Diagnostic	o	-	yes	Sequence of information as defined in ISO 8571-3
Shared ASE Information	o	no	no	-
F-CREATE Supported Parameters				
State result	m	-		Integer (0 or 1)
Action result	m	-		Integer
Override	m		-	Integer
Initial attributes	m			Sequence of information as defined in ISO 8571-3
Create password	o	yes	-	Sequence of information as defined in ISO 8571-3 of strings
Requested access	m	-		Bit string
Access passwords	o	yes	-	Sequence of information as defined in ISO 8571-3 of strings
Concurrency control	o	yes	-	Sequence of information as defined in ISO 8571-3
Shared ASE information	o	no	-	-
Account	o	yes	-	String
Diagnostic	o	-	yes	Sequence of information as defined in ISO 8571-3

* Requirement key: m = Mandatory parameter; o = Optional parameter.

! Support key: yes - Optional parameter supported; no = Optional parameter unsupported; blank space ( ) = Mandatory parameter (supported); - = Not applicable

TABLE 7 (Cont.): Protocol Parameters

Parameter	*ISO specification	Parameter support for service primitives		Range of acceptable values
		Request	Response	
F-DELETE Supported Parameters				
Action result	m	-		Integer
Shared ASE information	o	no	no	-
Charging	o	-	no	-
Diagnostic	o	-	yes	Sequence of information as defined in ISO 8571-3
F-READ-ATTRIBUTE Supported Parameters				
Action result	m	-		Integer
Attribute names	m		-	Bit string
Attributes	o	-	yes	Sequence of information as defined in ISO 8571-3
Diagnostic	o	-	yes	Sequence of information as defined in ISO 8571-3
F-OPEN Supported Parameters				
State result	m	-		Integer (0 or1)
Action result	m	-		Integer
Processing mode	m	-		Bit string
Contents type	m			Sequence of information as defined in ISO 8571-3
Concurrency Control	o	yes	yes	Sequence of information as defined in ISO 8571-3
Shared ASE information	o	no	no	-

* Requirement key: m = Mandatory parameter; o = Optional parameter.

! Support key: yes = Optional parameter supported; no = Optional parameter unsupported; blank space ( ) = Mandatory parameter (supported); - = Not applicable.

TABLE 7 (Cont.): Protocol Parameters

Parameter	***ISO specif-ication**	**Parameter support for service primitives**		**Range of acceptable values**
		Request	**Response**	
F-OPEN Supported Parameters				
Enable FADU locking	m		-	Boolean
Diagnostic	o	-	yes	Sequence of information as defined in ISO 8571-3
(Additional Parameters in the Internal File Service)				
Checkpoint	o	no	no	-
Activity Identifier	o	no	-	-
Recovery mode	m	-	-	
F-CLOSE Supported Parameters				
Action result	m	-		Integer
Diagnostic	o	-	yes	Sequence of information as defined in ISO 8571-3
Shared ASE information	o	no	no	-
F-BEGIN-GROUP Supported Parameters				
Threshold	m		-	Integer
F-END-GROUP Supported Parameters				
No parameters specified				
F-READ Supported Parameters				
Bulk data transfer specification (subparameters):				
FADU identity	m		-	Sequence of information as defined in ISO 8571-3
Access context	m		-	Integer
FADU lock	o	yes	-	Boolean

- * Requirement key: m = Mandatory parameter; o = Optional parameter.
- ! Support key: yes = Optional parameter supported; no = Optional parameter unsupported; blank space ( ) = Mandatory parameter (supported); - = Not applicable.

TABLE 7 (Cont.): Protocol Parameters

Parameter	*ISO specification	Parameter support for service primitives		Range of acceptable values
		Request	Response	
		F-WRITE Supported Parameters		
	Bulk data transfer specification (subparameters):			
FADU operation	m		-	Integer
FADU identity	m		-	Sequence of information as defined in ISO 8571-3
FADU lock	o	yes	-	Boolean
		F-DATA-END Supported Parameters		
Action result	m		-	Integer
Diagnostic	o	-	yes	Sequence of information as defined in ISO 8571-3
		F-TRANSFER-END Supported Parameters		
Action result	m	-		Integer
Shared ASE information	o	no	no	-
Diagnostic	o	-	yes	Sequence of information as defined in ISO 8571-3
		F-CANCEL Supported Parameters		
Action result	m	-		Integer
Diagnostic	o	-	yes	Sequence of information as defined in ISO 8571-3

- * Requirement key: m = Mandatory parameter; o = Optional parameter.
- ! Support key: yes = Optional parameter supported; no = Optional parameter unsupported; blank space ( ) = Mandatory parameter (supported); - = Not applicable.

1.14 Character Set(s) Supported for FTAM-PCI and for FADU Structuring Information

The VAX FTAM V1.1 implementation supports the IA5 and 8859/1 character sets for FTAM-PCI and for FADU structuring information.

1.15 Support for the Private Use Attribute

The private use attribute is unsupported.

1.16 Diagnostic Codes Usage

This section documents the diagnostic codes used. The VAX FTAM V1.1 implementation uses the FTAM diagnostic codes defined in ISO 8571-3, as restricted by the NBS agreements. The VAX FTAM implementation attempts to use specific rather than general diagnostic codes.

Directory Service

1.1 Directory Service Overview

Implements a local directory service and a Directory User Agent (DUA) so that directory functions are available both in a local context and when the VAX DEC/MAP system is part of a centralized or distributed directory configuration.

The MDS DUA implements the Directory Access Protocol as specified by the International Organization for Standardization (ISO) in standard 9594.

This standard defines the abstract service between a DUA and the Directory. The DAP protocol is mapped onto the Association Control Service Element (ACSE) and the Remote Operations Service Element (ROSE).

The DEC/MAP Directory Service does not provide a Directory System Agent (DSA) and the Directory System Protocol (DSP) is not implemented by MDS.

1.2 Directory Access Protocol (DAP)

The Directory Access Protocol is used to realize the Directory Abstract Service. MDS comprises the READ and MODIFY application-service-elements in addition to ACSE and ROSE. Use of these ASEs is specified by the Directory Access Application Context.

Use of Underlying Services

The ASEs that MDS supports are users of RO-INVOKE, RO-RESULT, RO-ERROR, RO-REJECT-U, and RO-REJECT-P services of ROSE. The MDS DUA operates in a synchronous manner resporting success or failure depending on whether the operation produced a result or an error.

The Directory Bind and Directory Unbind operations are mapped onto the A-ASSOCIATE and A-RELEASE services of ACSE in normal mode.

1.3 Directory Protocol Abstract Syntax

Abstract Syntaxes

The MDS ASEs share a single abstract syntax, the Directory Access Abstract Syntax defined as {joint-iso-ccitt (2) ds (5) abstract-syntax (9) id-as-directoryAccessAS (1)}. MDS requires that this abstract syntax be encoded according to ISO 8825 ASN.1 Basic Encoding Rules.

Directory Application Service Elements

The READ ASE supports the abstract-operations of the Read Port of the Directory. The MDS DUA supports the invocation of the Read and Compare operations through the READ ASE.

The MODIFY ASE supports the abstract operations available from the Modify Port of the Directory. The MDS DUA provides support for the invocation of the AddEntry, RemoveEntry, and ModifyEntry operations.

MDS does not provide support for the List and Search abstract operations available through the Search Port of the Directory. However, MDS does support list and search functionality on the local directory database.

Directory Application Contexts

The MDS DUA uses the Directory Access Application Context defined as {joint-iso-ccitt (2) ds (5) applicationContext (3) id-ac-directoryAccessAC (1)}. This application context allows the MDS DUA to access the READ and MODIFY ASEs.

Errors

MDS provides support for the notification of the user of the directory service of the following abstract errors: Abandoned, AttributeError, NameError, Referral, SecurityError, ServiceError, UpdateError, DSAReferral, and AbandonFailed.

1.4 Mapping Onto Used Services

The MDS DUA maps the Abstract-Bind operation on the A-ASSOCIATE service of the ACSE in normal mode. The application context proposed is the Directory Access Application Context. The MDS DUA as the initiator of the association supplies the Presentation Context Definition List which contains the ACSE abstract syntax ({2 2 1 0 1}) and the DAP apbstract syntax ({2 5 9 1}). The A-ASSOCIATE specifies the use of the Kernel and Duplex functional units of the Session layer.

The MDS DUA provides its Presentation Address and Application Entity Title (optionally) at association establishment time. The MDS defaults for the DUA calling address and calling application entity title are modifiable via MAPCL by the user.

The Abstract-Unbind operation is mapped onto the A-RELEASE service of the ACSE.

1.5 Supported Object Classes and Attributes

MDS supports the DSA object class ({2 5 4 13}) and the Titled Application Entity object class ({1 3 9999 41 0 3}).

As per the MAP specification, the Common Name, Object Class, and Presentation Address attributes are supported. Additionally, MDS provides support for the Country Name, Organization Name, Organizational Unit Name, Application Process Title, Application Entity Qualifier, and Supported Application Context attributes.

The sequence of the Country Name, Organization Name, Organizational Unit Name and Common Name form the MDS Application Simple Name of an application entity.

Under the Titled Application Entity object class, supported attributes are Presentation Address, Common Name, Application Process Title (optional), and Application Entity Qualifier (optional). This is in accordance with the MAP specification for the recommended object class.

Under the DSA object class, supported attributes are Presentation Address, Common Name, and Supported Application Context (optional).

1.6 Conformance

The MDS DUA implementation claims conformance pursuant to the statements listed hereunder:

Conformance Statements

a. The MDS DUA is capable of invoking the Read, Compare, AddEntry, RemoveEntry, and ModifyEntry operations as specified by ISO 9594.

b. The MDS DUA does not support either simple or strong security levels.

c. The MDS DUA has the capability of supporting the Directory Access Application Context as defined by its abstract syntax for the operations listed in (a).

d. The MDS DUA conforms to the mapping onto used services defined by IS 9594.

Association Control Service element (ACSE) Standards

ISO 8650: Information Processing Systems - Open Systems Interconnection - Protocol Specification for the Association Control Service element (ACSE).

With reference to section 10 (Conformance) of ISO 8650, VAX DEC/MAP:

- supports normal mode.
- supports initiation and reception of an association.
- supports version 1 of the protocol.

Presentation Layer Standards

ISO 8823: Information Processing Systems - Open Systems Interconnection - Connection Oriented Presentation Protocol Specification.

With reference to section 3 (Conformance), VAX DEC/MAP:

- supports the session kernel and duplex functional units.

- supports the presentation kernel functional unit.

- can initiate or respond to a CP-PPDU.

- supports all transfer syntaxes derived from the ASN.1 basic encoding rules defined in IS 8825 for: The following abstract syntax defined in IS 8650

- Joint-iso-ccitt association-control (2) abstract-syntax (1) apdus (0) version1 (1)

and the following abstract syntaxes defined in IS 8571

- ISO standard 8571 abstract-syntax (2) ftam-pci (1)
- ISO standard 8571 abstract-syntax (2) ftam-fadu (2)
- ISO standard 8571 abstract-syntax (2) unstructured-text (3)
- ISO standard 8571 abstract-syntax (2) unstructured-binary (4)

- supports normal mode.

Session Layer Standards

ISO 8327: Information Processing Systems - Open Systems Interconnection - Basic Connection Oriented Session Protocol Specification including Addendum 2 incorporating unlimited user data.

With reference to section 3 (Conformance), VAX DEC/MAP:

- implements session protocol, Versions 1 and 2.

For Session Protocol, Version 1, VAX DEC/MAP:

- implements session kernel and duplex functional units.
- does not implement extended concatenation.
- implements segmentation. The maximum TSDU size is 1490 octets.
- use of the transport expedited service is not implemented.

For Session Protocol, Version 2, VAX FTAM:

- implements session kernel and duplex functional units.
- implements major synchronize, resynchronize, and negotiated release functional units.
- does not implement extended concatenation.
- does not implement segmentation.
- use of the transport expedited service is not implemented.

Transport Layer Standards

ISO 8073: Information Processing Systems - Open Systems Interconnection - Connection Oriented Transport Protocol specification.

The Transport implementation supports many of the options provided for in the Transport Protocol Specification, International Standard ISO 8073. It conforms to ISO 8073 (as specified in Section 14 Conformance) providing:

- Protocol classes 0*, 2*, 4.
- Initiating Connect Request Transport Protocol Data Units (CR TPDUs).
- Responding to any other TPDU and operating the network service in accordance with the procedures for the class.
- Maximum TPDU size. The Transport implementation supports a maximum TPDU size of 8192 octets for class 4. When operating over a LAN, a TPDU size of 2048 octets is used to ensure that the ISO 8802-2 (IEEE 802.2) frame can be completely filled with data.
- All mandatory procedures and many optional procedures for each class supported as specified in International Standard ISO 8073.

* Not offered as a supported feature of VAX DEC/MAP V3.0.

Procedure	**Class 4**
TPDU with Checksum	Yes
TPDU without Checksum	Yes
Expedited Data Transfer*	Yes
No Expedited Data Transfer	Yes
Class 2 Flow Control	N/A
No Class 2 Flow Control	N/A
Normal Formats	Yes
Extended Formats	Yes

* Only supported via the VOTS programming interface

Key:
Yes = procedure is supported by the implementation.
No = procedure is not supported by the implementation.
N/A = procedure is not applicable to this class.

Network Layer Standards

ISO 8473: Information Processing Systems - Data Communications - Protocol for Providing the Connectionless-mode Network Service.

The Internet implementation supports the following functions:

Function	**Send**	**Receive**
PDU Composition	Yes	N/A
PDU Decomposition	Yes	Yes
Header Format Analysis	N/A	Yes
PDU Lifetime Control	N/A	Yes
Route PDU	Yes	N/A
Forward PDU	N/A	N/A
Segment PDU	Yes[1]	N/A
Reassemble PDU	N/A	Yes
Discard PDU	N/A	Yes
Error Reporting	N/A	Yes
PDU Header Error Detection	Yes	Yes
Padding	Yes[2]	Yes
Security	N/A	No
Complete Source Routing	Yes[1]	N/A
Partial Source Routing	Yes[1]	N/A
Record Route	N/A	N/A
Quality of Service Maintenance	N/A	N/A

Notes:

[1] This feature is implemented insofar as it applies to an End System implementation.

[2] The Internet implementation does not generate any padding on transmitted messages. Should a message be received which includes padding, it deals with the extra pad octets according to the protocol specification.
