DECnet/OSI for OpenVMS

Release Notes

November 1995

Documentation Comments

If you have comments or suggestions for this book or any of the DECnet/OSI documents, mail your comments to the following address:

DECnet/OSI Documentation Group
550 King St. (LKG2-2/Q5)
Littleton, MA 01460

If you have access to the Internet, mail your comments electronically to the following address:

doc_quality@lkg.mts.dec.com

Revision/Update Information: This is a new manual.

Operating System: OpenVMS Alpha and VAX Version 7.0

Software Version: DECnet/OSI Version 7.0 for OpenVMS

November 1995

Contents

4 Programming

5 CDI

6 DECdns

7 DECdts

8 FTAM

9 FTAM Application Programming Interface (API)

10 Virtual Terminal

1

General Information

1.1 New Features for DECnet/OSI for OpenVMS

The following new features were added in Version 7.0 for DECnet/OSI for OpenVMS:

- Support for 64-bit virtual addresses using the $QIO system service on Alpha. Applications can use the DECnet/OSI $QIO interface to send messages from and receive messages into buffers residing in extended address space. For further information, see Chapter 5 of these release notes.
- DECNET_REGISTER allows importing as well as exporting into a Phase IV database. You can copy exported information back to a Phase IV database, giving you a convenient way of updating all existing namespaces.
- Two new cluster configuration options have been added to the ADVANCED configuration: configuring satellite nodes and configuring cluster script locations. They are both sub-menus of the Configure Cluster Alias option. In order to run either of these, you must have successfully run NET$CONFIGURE ADVANCED and have NET$CONFIGURE executing on a cluster system. For more information, refer to the *DECnet/OSI for OpenVMS Part 2: Applications Installation and Advanced Configuration* guide, Modifying a Current Configuration chapter.
- NET$MGMT now supports the full X.25 entity set.
- Node abbreviations or nicknames can be added by defining logical names in the CDI$SYSTEM_TABLE logical name table.
- Reverse path caching has been added to routing to improve cache information about the paths used to reach remote nodes. In some cases, this also improves network performance.

1.2 Notes About Upgrading Your Systems

The following sections discuss information relevant to upgrading your systems.

1.2.1 Upgrading from DECnet/OSI Version 5.7 (Alpha Only)

If you are upgrading from DECnet/OSI Version 5.7 to Version 6.0 and later on an OpenVMS Alpha system, the installation procedure removes DECnet/OSI identifiers from RIGHTSLIST.DAT. Although the installation procedure adds them again, all user accounts that had DECnet/OSI identifiers granted to them no longer have them. This could potentially be disruptive to the proper operation of applications. To rectify this problem, you can do one of the following:

- Use AUTHORIZE to GRANT the identifiers after the upgrade.
- Install ECO2 of DECnet/OSI Version 5.7 before upgrading to Version 6.0 or later.

- Before you upgrade to Version 6.0 or later, edit the SYS$UPDATE:NET$PCSI_INSTALL.COM file and remove the following lines from that file:

```
$       set default sys$system
$       run sys$system:authorize
revoke/id NET$MANAGE SYSTEM
remove/id NET$EXAMINE
remove/id NET$MANAGE
remove/id NET$SECURITY
remove/id NET$DIAGNOSE
remove/id NET$DECLAREOBJECT
remove/id NET$REGISTERDNSOBJECT
remove/id NET$DECNETACCESS
remove/id NET$POSTEVENT
remove/id NET$TRACEALL
remove/id NET$TRACEHEADERS
remove/id NET$TRACEALLREMOTE
remove/id NET$TRACEHEADERSREMOTE
```

  These lines can be found after:

```
$!
$! REMOVE function
$!
$   if p1 .eqs. "REMOVE"
$   then
```

1.2.2 DEC X.25 Client (Alpha Only)

If you are upgrading to DECnet/OSI from DECnet Phase IV and have the DEC X.25 Client for OpenVMS Alpha Systems product installed, there are two X.25 products for OpenVMS Alpha systems. DEC X.25 Client is for DECnet Phase IV systems only. DEC X.25 is for DECnet/OSI systems only. If you upgrade to DECnet/OSI and want to retain access to X.25, you should first remove the X25CLIENT product and then install the DEC X.25 product, which provides both client and native X.25 functionality.

Refer to the *DECnet/OSI Applications Installation and Advanced Configuration* and the *DEC X.25 Installation and Configuration* manuals for details on installing and configuring the DEC X.25 product.

1.2.3 VAX P.S.I. DTE X.121 Mapping (VAX Only)

If your system is running the VAX Message Router X.400 Gateway V2.3 (MRX) and you upgrade to DECnet/OSI Version 5.7A or later, you need to either reconfigure the MTA dictionary used by MRX or use a PSI$L3CS.EXE image that does not perform X.121 mapping.

With the introduction of X.121 mapping in DECnet/OSI, the incoming DTE address seen by MRX is an NSAP preceded by a digit count. For inbound call verification to work correctly, you need to add a new /NETWORK_ADDRESS to each MTA record that contains a DTE address in its existing /NETWORK_ADDRESS. The new /NETWORK_ADDRESS must specify the NSAP as the DTE address.

The existing /NETWORK_ADDRESS is still used for outgoing calls, so do not modify or delete the existing /NETWORK_ADDRESS.

The NSAP is generated as follows. The DTE must be exactly 14 digits long. If it is less than 14 digits, you must pad it as follows:

- If it begins with a 0, pad it with leading 1s.

- If it begins with any number other than 0, pad it with leading 0s.

The padded DTE is preceded by 1052 if it has been padded with 1s or began with a 0. It is preceded by 1036 if it has been padded with 0s or began with any digit other than a 0. Therefore, a DTE of 0123456789 appears to be a DTE of 105211110123456789 and a DTE of 1234567890 appears to be a DTE of 103600001234567890.

For example, for an MTA with the MTA_NAME MTA_1 that currently has two /NETWORK_ADDRESSes of .ENG.X25_QNET%0123456789 and .ENG.X25_QNET%1234567890, you need to enter the following commands:

```
$ RUN SYS$SYSTEM:MRXMAN
MRXMAN> SELECT MTA/MTA_NAME=MTA_1
MRXMAN> MODIFY MTA/NETNO=NEW/NETWORK=.ENG.X25_QNET%105211110123456789
MRXMAN> MODIFY MTA/NETNO=NEW/NETWORK=.ENG.X25_QNET%103600001234567890
MRXMAN> ADD MODIFIED
MRXMAN> EXIT
```

To use an image that does not perform X.121 mapping, find the saveset DNVOSI058.S on your distribution media. Define the logical name KIT_DIRECTORY to be the directory that contains the saveset and perform the following backup command:

```
$ backup KIT_DIRECTORY:DNVOSI058.S;2/save-
     /select=PSI$L3CS.EXE-X121-NSAP-MAPPING-DISABLED -
     sys$common:[sys$ldr]PSI$L3CS.EXE/log/new_version
%BACKUP-S-CREATED, created SYS$COMMON:[SYS$LDR]PSI$L3CS.EXE;36
```

In the event of any difficulty, contact your Customer Support Center.

1.3 OpenVMS Operating System Notes

DECnet/OSI Version 7.0 for OpenVMS requires OpenVMS Version 7.0.

1.3.1 Upgrading OpenVMS from Version 1.5 (Alpha Only)

If you are upgrading an OpenVMS Alpha Version 1.5 system and have DECnet/OSI installed on OpenVMS Version 1.5, the upgrade disables the DECnet/OSI software. Once you install the current version, your previous DECnet/OSI configuration is retained. It is not necessary to reconfigure the DECnet/OSI software. However, if you want to configure DECnet/OSI to use X.25 or HDLC, you must reconfigure.

After you upgrade OpenVMS Alpha, you see boot-time error messages involving the files NET$MESSAGE.EXE and LES$LES_V30.EXE. This is normal and the messages cease once you install the current version.

Consult the OpenVMS Alpha Version 6.1 operating system release notes for further details.

1.3.2 OpenVMS Operating System Monitor Utility

The OpenVMS monitor utility function, MONITOR DECnet, shows the arriving and departing local packet rate, as well as the count of available LRPs. The fields Arriving Transit Packet Rate, Transit Congestion Loss Rate, and Receive Buffer Failure Rate are not used.

1.3.3 Case-Sensitivity of Nodes Running Old Versions of DECnet–VAX Software

Prior to the VMS Version 5.3-2 release, the DECnet–VAX software did not correctly process lowercase task names. Since the DECnet–VAX software stored object names in the database as uppercase, the comparison of a lowercase incoming task specification to the elements in the object database would not find any matches. The DECnet–VAX software was modified to convert incoming task specifications to uppercase before searching for a match in the object database.

When communicating with nodes running old versions of the DECnet–VAX software, you should specify uppercase task specifications on systems that preserve the case of outgoing task specifications. For example, use the following:

```
$ TYPE NODE::"TASK=TEST"  rather than
$ TYPE NODE::"TASK=test".
```

1.4 Backtranslation of IP Addresses

For IP addresses, session control now returns the full name if the synonym is not available regardless of the "Session Control Application Node Synonym" setting. If neither the full name nor synonym are available, the IP$aa.bb.cc.dd format is returned.

1.5 Known Restrictions

The following section refers to issues regarding MOP.

1.5.1 Enabling MOP

If you require the use of DECnet/OSI MOP on an Alpha Version 7.0 system, the SYSGEN value MULTITHREAD must be set to 0. (Note that this will also disable kernel threads functionality systemwide.)

MOP (Maintenance Operation Protocol) is a component of DECnet used to load assist satellites in a cluster, communicate with terminal servers, and provide a general downline loading mechanism.

2
Installation and Configuration

2.1 Migration Notes

The following sections refer to issues regarding migration from DECnet Phase IV to DECnet/OSI for OpenVMS.

2.1.1 Migrating DECnet Phase IV Database Into Local Database

When migrating the DECnet Phase IV database into the local database, DECNET_REGISTER finds only NETNODE_REMOTE.DAT in SYS$SYSTEM. It currently cannot use the logical NETNODE_REMOTE to find this database. For the automatic conversion of your Phase IV database to take place on your system disk, you need to copy NETNODE_REMOTE.DAT to SYS$COMMON:[SYSEXE] if this file does not already exist. This problem will be corrected in a future release.

2.1.2 Migrating DECnet Phase IV Objects to Phase V Session Applications

When migrating the DECnet Phase IV Objects database into the Phase V session control applications script, DECNET_REGISTER finds only NETOBJECT.DAT in SYS$SYSTEM. It currently cannot use the logical NETOBJECT to find this database. For the automatic conversion of your Phase IV objects to take place on your system , you need to copy NETOBJECT.DAT to SYS$COMMON:[SYSEXE] if this file does not already exist. This problem will be corrected in a future release.

2.2 Reinvoke NET$CONFIGURE.COM After Installing DECnet/OSI

After up grading from a DECnet/OSI Version 5.*x* system to DECnet/OSI Version 6.*x* (where *x* stands for the release number), use NET$CONFIGURE.COM to reconfigure your node so the directory services part is configured appropriately. You may either use Option 1 (perform full configuration) or Option 2 (change node name/namespace name) to do this.

As a result, the DECnet/OSI checksum file is upgraded for you and a new NCL script called SYS$MANAGER:NET$SEARCHPATH_STARTUP.NCL is created.

Configuring New Namespaces

After configuring a new name space on a server, NET$CONFIGURE will not be able to rename the server node into the new name space. To register the server node in the new name space the decnet_register utility needs to be used. This only applies to the server node itself.

2.3 Message Displays After Reboot (VAX Only)

After the installation of OpenVMS Version 6.1, but before the installation of DECnet/OSI Version 7.0, the following messages appear soon after a reboot:

```
%SYSINIT-W- Failed to load LES comms executive
%SYSINIT-E- Error loading <SYS$LDR>LES$LES_V30.EXE    status = 00002398
```

After the installation of DECnet/OSI Version 7.0, these messages no longer occur.

2.4 Use of /REMOTE with POLYCENTER Installation (Alpha Only)

Although the POLYCENTER Software Installation (PCSI) utility allows you to install software on a target system that is not the currently running system, this should not be done with DECnet/OSI. The DECnet/OSI installation procedure performs functions that affect the running system, so it does not support use of the /REMOTE qualifier.

2.5 DCL Symbol Problem

Using long strings as descriptive names for data links and routing circuits in the NET$CONFIGURE.COM procedure reduces the number of definable data links and routing circuits. For maximum use of all possible data links and routing circuits, use the default values.

3
Network Management

3.1 Known Problems, Restrictions, and Workarounds

The following sections discuss known network management problems and workarounds.

3.1.1 NET$SHUTDOWN Known Problems

The following list discusses the known problems with NET$SHUTDOWN.

- NET$SHUTDOWN.COM hangs with FDDI controllers (DEMFA, DEFQA, DEFTA).

  A known problem exists in which NET$SHUTDOWN hangs if there is an FDDI routing circuit in use. This is a synchronization problem and NET$SHUTDOWN can hang at either the "Disable Routing Circuit *" or "Delete Routing Circuit *" command. This problem will be fixed in a future release. In the meantime, the workaround is to explicitly disable and delete those routing circuits before you execute NET$SHUTDOWN.

- NET$SHUTDOWN.COM does not work properly on a remotely displayed DECterm window.

  If you use SYS$SYSTEM:SHUTDOWN.COM to shut down your system and you ask that it delay shutting down your system for a certain number of minutes, SHUTDOWN.COM calls NET$SHUTDOWN.COM to shut down the network 5 minutes prior to the time you specified. This means that anyone connected to the system using DECnet/OSI is disconnected from the system 5 minutes earlier than expected. This problem will be corrected in the future.

3.1.2 RECEIVE BUFFERS Attribute on HDLC LINK Entity (Alpha Only)

The RECEIVE BUFFERS attribute of the HDLC LINK entity is not implemented in DECnet/OSI. If you attempt to set the attribute, you will see the following message from NCL:

```
NCL>SET NODE 0 HDLC LINK HDLC-0 RECEIVE BUFFERS 8

Node 0 HDLC Link HDLC-0
at 1994-04-28-09:51:10.780-04:00I0.113

 command failed due to:
set list error

Characteristics
no such attribute ID: Receive Buffers = 8
```

This message is harmless. HDLC allocates the number of receive buffers that it requires; it is not settable through network management.

3.1.3 Outgoing Alias Enabled Defaults to TRUE

For Active Servers (Declared Objects), the Session Control Application characteristic "Outgoing Alias Enabled" defaults to TRUE when the equivalent on Phase IV was FALSE. You can work around this problem by predefining the characteristic in the application database.

For example, an active server defines itself to be called "BRYAN1" and it is necessary for it to have Outgoing Alias and Outgoing Proxy to be FALSE. Pre-define these characteristics in the application database by running NET$CONFIGURE.

```
        Configuration Options:

        [0]     Exit this procedure

        [1]     Perform an entire configuration
        [2]     Change naming information
        [3]     Configure Devices on this machine
        [4]     Configure Transports
        [5]     Configure Timezone Differential Factor
        [6]     Configure Event Dispatcher
        [7]     Configure Application database
        [8]     Configure MOP Client database
        [9]     Configure Cluster Alias

* Which configuration option to perform?                [1] : 7
* Do you want to ADD or DELETE an Application?        [ADD] :
* What is the name of the Application?                      : bryan1
* What is the destination type for 'bryan1'?         [NAME] :
* What is the destination name for 'bryan1'?                : bryan1
* Do you want to specify another application address?   [NO] :
* What is the name of the Client for 'bryan1'?              :
* What is the Image name for 'bryan1'?                      :
* Incoming Alias for 'bryan1' enabled?               [TRUE] :
* Incoming Proxy for 'bryan1' enabled?               [TRUE] :
* Outgoing Alias for 'bryan1' enabled?               [TRUE] : false
* Outgoing Proxy for 'bryan1' enabled?               [TRUE] : false
* Require node synonym for 'bryan1' enabled?         [TRUE] :
* What is the Incoming OSI TSEL for 'bryan1'?               :
* What is the User Name for 'bryan1'?        [BRYAN$SERVER] :
* What UIC should 'bryan1' use?                 [[200,200]] :
* Rights identifiers for 'BRYAN$SERVER'?                    :
%NET$CONFIGURE-I-MAKEACCOUNT, this procedure creates user account BRYAN$SERVER
* Do you want to generate NCL configuration scripts?   [YES] : yes
```

3.1.4 Restrictions on Using DECdtm Services (VAX Only)

If you are using DECnet/OSI, you cannot use DECdtm services on OpenVMS VAX Version 6.0.

While DECdtm supports DECnet/OSI full names, Rdb and ACMS do not. Rdb and ACMS only work with node synonyms.

3.2 Routing Notes

The following sections refer to routing.

3.2.1 Use Default Integrated Routing Mode

Use the default Integrated routing mode in a integrated routing environment where the routers can handle Phase IV to DECnet/OSI or DECnet/OSI to Phase IV packet format conversions. Cluster alias does not work properly if you do not use the default. Use the Segregated routing mode only when the adjacent router(s) cannot perform Phase IV to DECnet/OSI or DECnet/OSI to Phase IV packet conversions.

3.2.2 Change to DNA ADDRESS FORMAT Attribute

The DNA ADDRESS FORMAT attribute controls only the interpretation of Phase IV addresses. It no longer controls autoconfiguration. An empty set of MANUAL NETWORK ENTITY TITLES attributes indicates the use of autoconfiguration. When non-empty, autoconfiguration does not take place. This routing attribute may only be changed from the empty set to the non-empty set or from the non-empty set to the empty set when the entity is in the off state. These changes allow the end systems to disable autoconfiguration, but not form adjacencies to-out-of area Phase IV addresses.

3.2.3 End System No Longer Requires Autoconfiguration for Cluster Alias

The end system no longer requires autoconfiguration to use the cluster alias. It can now construct the NSAP address for the alias from the manual NETs.

3.3 Transition Notes

The following sections discuss transition information.

3.3.1 Decimal Syntax DSPs Not Supported

DECnet/OSI for OpenVMS supports only those IDPs (initial domain parts) that specify the use of binary syntax DSPs (domain-specific parts). IDPs that specify decimal syntax DSPs are not supported.

3.3.2 COLLECT Command and Addressing in DECNET_MIGRATE Commands

Some connection errors might be reported for nodes with DECnet/OSI addresses that are not also Phase IV compatible.

This occurs because the network management interface on OpenVMS cannot deal with connections to nodes by the explicit use of a DECnet/OSI address that is not also Phase IV compatible.

If a DECnet/OSI node also has a Phase IV compatible address, information is collected for it using that address.

3.4 NSP Transport Notes

The following sections discuss information relevant to NSP Transport.

3.4.1 Requirements for Deleting and Creating the NSP Entity

If the NSP entity is deleted and subsequently recreated, you must issue the following directives to inform DNA SESSION CONTROL that the NSP transport service is available:

```
NCL> DELETE SESSION CONTROL TRANSPORT SERVICE NSP
NCL> CREATE SESSION CONTROL TRANSPORT SERVICE NSP PROTOCOL %X04
```

You cannot issue DELETE SESSION CONTROL TRANSPORT SERVICE NSP while NSP PORT entities exist.

3.4.2 DISABLE Command Problem

The DISABLE NSP command can hang at the NCL prompt. To recover, follow these steps:

1. Enter another NCL session and issue:

   ```
   NCL> SHOW SESSION CONTROL PORT * ALL
   ```

2. Delete each session in turn with:

   ```
   NCL> DELETE SESSION CONTROL PORT session_port_name
   ```

Do not try to re-enable NSP until the NCL prompt returns from the DISABLE NSP command.

3.5 OSI Transport Notes

The following sections discuss OSI Transport information.

3.5.1 Connection to Non-Conformant Systems–Connection Failure

By default, OSI Transport sends Preferred maximum TPDU size, Request Acknowledgment, and Implementation ID parameters in its CR TPDU. According to ISO 8073, OSI Transport providers should ignore unknown parameters while processing CR TPDU.

However, some vendor implementations do not conform and do not ignore unknown parameters in CR TPDU. This results in Connection failure. The following example provides a way to prevent issuance of these parameters in CR TPDU.

```
NCL> set osi transport template template-id -
 send Preferred maximum TPDU size FALSE

NCL> set osi transport template template-id -
 send Request Acknowledgment FALSE

NCL> set osi transport template template-id -
 send implementation ID FALSE
```

3.5.2 OSI Transport Congestion Avoidance/Change of Default Value

One feature of OSI Transport is the ability to use the "Congestion Experienced" field in the Connectionless Network Service (CLNS) routing header, and to implement a "Congestion Avoidance" scheme in heavily congested networks. The CLNS Congestion Experienced field is used by routers that support this feature (such as DECNIS) to give an early indication of congestion. When OSI Transport receives data that passed through a network path where the "Congestion Experienced" bit is set, OSI Transport reduces the transmit rate of the sending end system to help alleviate network congestion.

While this feature works well in networks where all protocols support "Congestion Avoidance" mechanisms, it has been noted that in some heavily congested multi-protocol networks this feature can negatively impact the performance of DECnet compared to other protocols.

Digital recognizes that most of its customers have multiprotocol networks. In this environment, not all network protocols have "Congestion Avoidance" mechanisms. Therefore, we have changed the default of this characteristic to be disabled.

If you operate in an environment where you can take advantage of "Congestion Avoidance" mechanisms we recommend that you enable the feature again.

To change Transport Congestion Avoidance values, you must invoke NET$CONFIGURE in ADVANCED mode and use Option 4 (Configure Transports). You should answer NO to the question, Is this System operating in a Multi-Protocol Network? [YES] :

3.5.3 Known Problems and Workarounds

The following lists the known OSI Transport problems and the workarounds.

- If OSI Transport times out during data transfer mode because the remote end is unreachable, the error SS$_CONNECFAIL may also be returned in the IOSB for the $QIO(IO$_READVBLK) or $QIO(IO$_WRITEVBLK) call. As a workaround, user code should be made to handle SS$_TIMEOUT as well as SS$_CONNECFAIL.
- When defining an RFC1006 address using the OSIT$NAMES logical name table, you must put a quotation mark around the address part.
- The DISABLE OSI Transport command can hang at the NCL prompt. To recover, follow these steps:
  1. Enter another NCL session and issue:

     ```
     NCL> SHOW SESSION CONTROL PORT * ALL
     ```

  2. Delete each session in turn with:

     ```
     NCL> DELETE SESSION CONTROL PORT session_port_name
     ```

  Do not try to re-enable OSI Transport until the NCL prompt returns from the DISABLE OSI Transport command.
- The prohibition against 0 and 1 as valid NSELs has been removed, but the following restrictions apply:
  - The NSELs for OSI Transport must be the same in both transport partners if the packet is traversing a backbone containing Phase IV routers (as opposed to routing vector domains).
  - The cluster alias uses two ranges of NSELs to reduce the risk of a collision, Digital recommends that you do not choose values from those ranges, or at least that you do not choose the first few values in each range.
- Expedited data could be delivered before normal data on on SMP VAXstation 3250 that has more than one processor.
- The OSI Transport entity does not support a value of zero for the maximum remote NSAPs attribute. The OSI Transport entity does not support a value of ANY for the template network service attribute. If this attribute is set to ANY, it is treated by OSI Transport as CLNS.

3.5.4 Configuring X.25 Access Filters for Use by OSI Transport (VAX Only)

Create X.25 Access filters with the VAX P.S.I. configuration procedure, using the Declaring a Network Process section, as follows:

1. On the introduction screen to Declaring a Network Process section, answer YES to the question:

   Do you want X.25 or X.29 programs to specify filter names in $QIO(IO$_ACPCONTROL) calls?

2. On the next screen, answer NO to the question:

Do you want IO$_ACPCONTROL calls issued by your programs to name any dynamic filters?

3. On the next screen, answer YES to the question:

   Do you want IO$_ACPCONTROL calls issued by your programs to name any static filters?

4. On the following two screens, you can set up the attributes for the X.25 Access filter. You are prompted to enter Network Process Filter information. You must complete the following fields:

   - Filter Name: OSI Transport
   - Call Data Value: for example, %X03010100
   - Call Data Mask: for example, %XFFFFFFFF

   You can set the Filter Name to any name. However, the name you use must match the name you entered as the X.25 access Template Name and as the OSI Transport CONS template name. The OSI TRANSPORT TEMPLATE attribute CONS TEMPLATE is case sensitive and must match the OSI TRANSPORT attribute CONS FILTERS exactly.

   The Call Data Value and Call Data Mask entries are used by VAX P.S.I. to determine whether an inbound network connect should be passed to OSI transport.

   For other fields, use the default value provided.

   You can set up a security filter corresponding to this X.25 Access filter in the Incoming Security for Network Processes section of the VAX P.S.I. configuration procedure.

3.5.5 Configuring CONS

To configure CONS support, each element in the set of CONS Filters attribute of the OSI Transport entity must have a corresponding X25 access filter of the same name. By default the CONS Filters attribute of the OSI Transport entity is set to OSI Transport. Similarly, the CONS Template attribute of the OSI Transport Template subentity must contain a name that is a PSI filter and is contained in the set of CONS Filters of the OSI Transport entity. The default value of the CONS Template attribute of an OSI Transport Template subentity is OSI Transport.

3.5.6 Requirements for Deleting and Creating the OSI Entity

If the OSI entity is deleted and subsequently recreated, you must issue the following directives to inform DNA SESSION CONTROL that the OSI Transport service is available:

```
NCL> DELETE SESSION CONTROL TRANSPORT SERVICE OSI
NCL> CREATE SESSION CONTROL TRANSPORT SERVICE OSI PROTOCOL %X05
```

You cannot issue DELETE SESSION CONTROL TRANSPORT SERVICE OSI while OSI PORT entities exist.

3.5.7 Communications Between OSI Transport Systems and VOTS V2.0 Systems Using CLNS

If you need communication between a VOTS V2.0 system and an OSI Transport system using the full Internet CLNS protocol, you must use an intermediate system (router). OSI Transport implements only the Internet protocol. An OSI Transport system has no way of finding another end system that does not support ES-IS without using an intermediate system.

If a DEC WANrouter is used as an intermediate system, you must configure it as a link state router.

If the VOTS V2.0 system and the DEC WANrouter reside on the same LAN subnetwork AND the VOTS V2.0 system is configured with a DECnet/OSI compatible NSAP address, the DEC WANrouter you need only to configure it as a level 1 router.

If the VOTS V2.0 system does not have a Phase V DNA compatible NSAP address, or if the VOTS V2.0 system and the DEC WANrouter do not reside on the same LAN subnetwork, the DEC WANrouter yo must configure it as a level 2 router.

When using a level 1 router, you must create a manual adjacency on the router for the VOTS V2.0 system. When using a level 2 router, you must create a reachable address on the router for the VOTS V2.0 system. See the DEC WANrouter configuration and management guides for details about how to configure manual adjacencies and reachable addresses.

OSI Transport systems and VOTS V2.0 systems on the same LAN can communicate without an intermediate system, using the Null Internet CLNS protocol.

4
Programming

4.1 64-Bit Virtual Address Support (Alpha only)

DECnet/OSI V7.0 for OpenVMS provides support for 64-bit virtual addresses via the $QIO system service. Applications can use the DECnet/OSI $QIO interface to send messages from and receive messages into buffers residing in extended address space. Existing 32-bit applications will continue to run with the 64-bit capable DECnet/OSI software. It is not necessary to recompile or relink existing applications that use 32-bit addresses with the DECnet/OSI $QIO interface.

The following table specifies the DECnet/OSI $QIO parameters that accept 64-bit virtual addresses. Refer to the *DECnet/OSI for OpenVMS Programming* book for additional information about using the DECnet/OSI $QIO interface. Also, refer to general 64-bit information in the OpenVMS operating system documentation.

Operation	$QIO function code and modifier	64-bit address parameter(s)
Connect Initiate	IO$_ACCESS	P2 - 32-bit or 64-bit address of a 32-bit or 64-bit descriptor for the Network Control Block (NCB).
Connect Accept	IO$_ACCESS	P2 - 32-bit or 64-bit address of a 32-bit or 64-bit descriptor for the Network Control Block (NCB).
Connect Reject	IO$_ACCESS!IO$M_ABORT	P2 - 32-bit or 64-bit address of a 32-bit or 64-bit descriptor for the Network Control Block (NCB).
Read	IO$_READVBLK	P1 - 32-bit or 64-bit buffer address.
Write	IO$_WRITEVBLK	P1 - 32-bit or 64-bit buffer address.
Disconnect (Synchronous)	IO$_DEACCESS!IO$M_SYNCH	P2 - 32-bit or 64-bit address of a 32-bit or 64-bit descriptor for optional user data.
Disconnect (Abort)	IO$_DEACCESS!IO$M_ABORT	P2 - 32-bit or 64-bit address of a 32-bit or 64-bit descriptor for optional user data.
Declare Named Object/Application	IO$_ACPCONTROL	P1 - 32-bit or 64-bit address of a 32-bit or 64-bit descriptor for a NFB$C_DECLNAME NFB block.
		P2 - 32-bit or 64-bit address of a 32-bit or 64-bit string descriptor for the object/application name.
Declare Numbered Object/Application	IO$_ACPCONTROL	P1 - 32-bit or 64-bit address of a 32-bit or 64-bit descriptor for an NFB$C_DECLOBJ NFB block.

4.1.1 Restrictions

64-bit virtual addresses can only be used with $QIO operations performed on the "NET:" devices. 64-bit virtual addresses cannot be used with $QIO operations to the "OS:" devices.

For Transparent task-to-task communication, the $ASSIGN system service will only accept 32-bit addresses and 32-bit descriptors for the DEVNAM parameter when it describes a Network Control Block (NCB). Non-transparent task-to-task operations can be used when it is desirable to use 64-bit descriptor addresses and 64-bit descriptors to specify an NCB.

4.2 Backtranslation of IP Addresses

For IP addresses, session control now returns the full name if the synonym is not available regardless of the "Session Control Application Node Synonym" setting. If neither the full name nor synonym are available, the IP$aa.bb.cc.dd format is returned.

If the system is configured to use IP addresses, the application should be prepared to receive a node name/address string longer than six characters in length even if "Session Control Application Node Synonym = TRUE" is specified for the application.

4.3 CMISE API

For applications that call the CMISE API to perform network management functions, you should be aware of the following changes new in this release.

4.3.1 Stricter Functional Unit Validation

Previously, the CMISE API, while supporting the use of functional units on an association, had not strictly enforced the relationship between functional units and permitted operations on an association. Beginning with this release, Functional Unit restrictions as specified in ISO 9595 Section 7.1 and all of 7.2.1–7.2.5 are enforced.

While it is the application program's responsibility to only supply legal parameters that are supported within the functional units negotiated on an association, the CMISE API performs validation on every verb called by an association initiator or responder. The API now returns an error if an invalid request is made by the application program outside the scope of the agreed-upon functional units.

For example, the linked id parameter is passed on an M_Get_Resp and the multiple reply functional unit is not found to be valid on that association. CMISE will return an error back to the application (CMISE_S_ASNERR, MCC_S_ILVMISTYPED).

4.3.2 More PDU Encodings Supported

The CMISE API now supports additional PDU encodings as specified in Appendix A of ISO/IEC 9596-2. These correspond to table A.57, A.63, A.65, and A.69. Prior to this release, the CMISE API did not support transmission or receipt of PDUs containing only an Invoke Id and no operation value. The application program implications are that a new flag, M_FG_RORS_BASIC, has been added as one of the bit flags that can be passed in the Flags parameter. If this flag is set, only the Invoke ID will be encoded in a PDU sent as a response. If a responder application wishes to send this type of PDU, all it needs to do is set this flag. Verbs that now accept this flag are:

- M_Action_Resp
- M_Delete_Resp
- M_Event_Resp
- M_Set_Resp

The verbs M_Create_Resp and M_Get_Resp will return an error if the flag M_FG_RORS_BASIC is set (CMISE_S_ASNERR, MCC_S_BAD_PDU).

CMISE API does not allow encoding a response containing only the Invoke ID for these verbs. See ISO 9596-2, Tables A.61 and A.67.

4.4 Applications Must Wait for Connect Accept to Complete Before Using a Connection

In Phase IV, a program can accept an incoming request to establish a link with a no-wait $QIO (FUNC=IO$_ACCESS) call, followed immediately with a read or write operation on the channel. This behavior is not preserved with DECnet/OSI. In DECnet/OSI, your application must wait for the the completion of the $QIO used to accept the connect before attempting to use the connection.

5
CDI

5.1 CDI_SYSTEM_TABLE Logical Table

It is possible in Version 6.2 or later to use a logical table (CDI$SYSTEM_TABLE) to define node synonyms. The following commands are used to create and examine logical names in a CDI_SYSTEM_TABLE. A system named "bks.pub.dec.com" maybe defined as "bks".

To define a table:

```
create/name_table/exec/parent=LNM$SYSTEM_DIRECTORY CDI$SYSTEM_TABLE
```

To define a logical name:

```
define/table=CDI$SYSTEM_TABLE bks bks.pub.dec.com
```

To examine a logical name:

```
show logical/table=CDI$SYSTEM_TABLE bks
   "bks.pub.dec.com" = "bks" (CDI$SYSTEM_TABLE)
```

System privileges SYSNAM is required.

5.2 Node Access Problem

When a node is configured as a DNS server, do not use NET$CONFIGURE.COM to define Local as the primary name service (to be searched first) and DECdns as the secondary name service (to be searched after Local).

When called by NET$CONFIGURE.COM during configuration, the DECNET_REGISTER tool can return a node lookup failure due to the local node not having the proper access to read information stored in DECdns, the secondary name service.

5.3 Backtranslation Failures Over DOMAIN (DNS/BIND)

Incoming connections over an IP network may not work with applications that require a Phase IV-style (six character or less) node name unless Domain synonyms are set up as described in the DECnet/OSI for OpenVMS Installation guides.

6
DECdns

6.1 DECdns Control Program Limitation

You cannot use the DECdns Control Program, DNSCP, to manage information stored in the Local namespace. Instead, use the new DECNET_REGISTER tool.

6.2 System Hangs at Startup in DNS Clerk

The system may hang upon startup in the DNS Clerk initialization procedures. This may occur when a system with three or more LAN adapters fails on startup of the DECdns Clerk. The startup fails because the DECdns Clerk Buffer Limit has been reached. The workaround is to edit the SYS$STARTUP:DNS$CLERK_STARTUP.COM "/BUFFER_LIMIT=150000", and modify this parameter to "/BUFFER_LIMIT=300000".

6.3 DECdns Known Problem (VAX Only)

This version of DECdns has a memory leak that is most apparent on systems which experience high rates of process creation and deletion (for example, systems running any sort of network monitoring/management software that spawns off subprocesses frequently). On these systems, non-paged pool will eventually be consumed, requiring a reboot. For systems that do not spawn/create new processes doing DECnet networking, this leak should not have as severe an impact.

6.4 Limitation for Number of Members in a Single Group

DECdns has an internal limitation for the number of members (principals) that may be stored in a single group. The result of this problem is that the server process can crash during the skulk procedure. This skulk does not have to happen immediately, and may occur up to 24 hours after the group size limit was exceeded. This can make diagnosis of the problem more difficult, due to the indirectness of the symptoms. This problem affects DECdns servers (OpenVMS, ULTRIX and Digital UNIX) who use groups to control access to objects stored in their DECdns namespaces.

There is a 100-entry table used to hold the members of a group while the group is being validated to ensure that it does not contain members who are groups that contain the original group itself again. This places a 100-entry limit on the number of members in a group.

The workaround for this problem is to limit the number of entries to less than 100. We recommend that no group contain more than 75 members so that this limitation is never reached.

A simple way to limit the numbers of members of any one group is to create subgroups that are members of the original group. For example, an administrator wants to create a group with 200 members who all have authorization to manage a certain directory in the namespace. The group will be called ".dir_admin".

Two possible solutions are:

- Create three groups called ".dir_admin_a_to_f", ".dir_admin_g_to_l", and ".dir_admin_m_to_z". These three groups are entered as the only three members of the group ".dir_admin". Each contains a part of the alphabetically sorted list of members.
- Create subgroups using some other partitioning algorithm, such as by site code, or group function instead of the alphabet as used in solution 1.

6.5 DECdns Server Software Is Not Available for Alpha Systems

DECdns server software is not available for Alpha systems. All references to DECdns servers (and to their resident clearinghouses) throughout the remainder of these release notes apply to the DECdns servers in your namespace that are running on ULTRIX or OpenVMS VAX systems.

6.6 DNS$SHARE.EXE Not Supported

DECnet/OSI for OpenVMS no longer supports SYS$LIBRARY:DNS$SHARE.EXE. The SYS$DNS[W] system service is resolved using SYS$PUBLIC_VECTORS. No special link options are necessary.

6.7 DECdns Clerk Startup

The DECdns clerk startup may output the following error message:

```
Create Node 0 DNS Clerk Known Namespace CZ

command failed due to:
process failure

A Known Namespace with this name or NSCTS already exists
```

This is harmless. The DECdns clerk configuration procedure puts a CREATE DNS CLERK KNOWN NAMESPACE command for the default namespace into the DECdns clerk NCL startup file. The command is there for the case when the DECdns cache file has been deleted or corrupted.

6.8 DECdns Clerks and Servers May Require Additional PAGEDYN Resources

DECdns clerk and server systems can require additional paged dynamic memory resources (PAGEDYN).

For DECdns clerk systems, consider increasing the PAGEDYN resource on the node if you see the following errors: RESOURCEERROR or NONSRESOURCES (during skulk operations).

Before you configure a system as a DECdns server, check that the system has at least 50,000 free bytes of paged dynamic memory. Insufficient paged dynamic memory on servers can cause configuration errors, skulk failures, and in some cases normal clerk operations can fail with either of the following errors: RESOURCEERROR or NONSRESOURCES.

The DNS$SERVER.LOG file can also contain messages regarding failures due to this resource.

Servers holding master replicas of directories that have many read-only replicas can also require additional paged dynamic memory.

6.9 New DNS$CLERK_STARTUP.COM

The DNS$CLERK_STARTUP.COM procedure now calls SYS$STARTUP:DNS$NAMES.COM (if present). Use this file to define any system-local name abbreviations you want DECdns to use. Names should be defined with the following command:

```
$ DEFINE/NOLOG/TABLE=DNS$SYSTEM name "equivalence"
```

6.10 DECdns Clerks Can Now Use the Outgoing Alias When Connecting to DECdns Servers

With a change in DECnet/OSI Session Control, all DECdns requests from nodes in an OpenVMS cluster can now send the cluster alias address as the source address. The current behavior is to send the individual node address.

To use this new feature, edit the file SYS$MANAGER:DNS$CLERK_CLUSTER.NCL and set OUTGOING ALIAS = TRUE.

To affect the running system, use the following NCL command on all nodes in your cluster running this release of DECnet/OSI:

```
NCL> SET SESSION CONTROL APPLICATION DNSCLERK OUTGOING ALIAS = TRUE
```

Note

Implementing this new feature may require nontrivial changes to the current access control in your namespace since you are essentially changing the source address of DECdns clerk requests.

6.11 Unknown LAN Device Types

To configure DECdns over an unknown LAN device type, use the system logical name DNS$ETHERNET_DEVICE, where XY defines the new device type:

```
$ DEFINE/SYSTEM DNS$ETHERNET_DEVICE "XYA0"
```

At the present time, DECdns automatically configures over the following LAN devices:

```
EC*, EF*, EL*, ER*, ES*, ET*, EW*, EX*, EY*, EZ*,
FA*, FC*, FQ*, FR*, FW*, FX*, FZ*, IC*, IR*, XE*, XQ*
```

6.12 High Convergence Directories Not Recommended

If any of your DECdns directories are set to high convergence (DNS$Convergence = high), Digital strongly recommends that you reset them to medium convergence. The high convergence setting is intended only for temporary use in limited, troubleshooting situations. Directories that are permanently set to high convergence can, in certain cases, cause extensive network and memory usage—possibly leading to a database corruption. Some elements of this behavior are defects that may be corrected in a future release of DECdns. Permanent use of the high convergence setting, however, will continue to be discouraged.

To set a directory's convergence to medium, use the following DNS$CONTROL command:

```
dns> set directory <directory-name> DNS$Convergence = medium
```

6.13 Known DECdns Problems

You may encounter the following problems:

- DNS$Control may return "Syntax Error" if the clerk is disabled.
- You should not use DNS$Control to modify the DNA_NodeSynonym attribute on DECnet node objects. DNS$Control displays the attribute properly, but does not modify it properly. Use the appropriate DECnet node registration tools to modify DECnet data stored in DECdns.
- Recreated entries retain the case of the original entry name.
- On OpenVMS VAX systems, if you have trouble creating a clearinghouse, make sure that either the DNS$SERVER account (UAF entry) does not exist or its default directory does exist.

6.14 UNKNOWN and WORLD Pseudo-Namespaces

The DECdns server now recognizes two predefined namespaces: UNKNOWN and WORLD. The UNKNOWN namespace provides principal names for nodes whose names cannot be determined. If a Phase IV node name cannot be determined with backtranslation, the server generates a name for the node in the form:

```
UNKNOWN:.%Xnnnn
```

where nnnn is the NSAP of the connecting node. The following is an example principal generated by this situation:

```
UNKNOWN:.%X490004AA00000400FB1020.MCINTYRE
```

The WORLD namespace is used solely in access control sets. The WORLD namespace matches the specified principal in any namespace presented to the server (including the UNKNOWN namespace). For example:

FREDCO:.*... matches any principal in the FREDCO namespace.

WORLD:.*... matches any principal in any namespace.

WORLD and UNKNOWN are ignored by DNS Version 1 servers. They can display as hexadecimal NSCTSs on older DECdns implementations.

6.15 DNS Version 1 and DECdns Version 2 Server Incompatibility

With DECdns Version 2 servers on ULTRIX and OpenVMS VAX systems, the DNS$CHDIRECTORIES attribute was added to a clearinghouse object to list the set of directories replicated at that particular clearinghouse. A problem arises when:

- You have more than 200 directories stored in the DECdns Version 2 clearinghouse (for example, .eng.host_ch).

- You have a DNS Version 1 server replicating the directory containing the clearinghouse object (for example, a DNS Version 1 server replicating the .eng directory).

This causes the DNS$CHDIRECTORIES attribute to grow too large to be handled by DNS Version 1 servers. The directory fails to skulk with the following status:

```
"Insufficient local resources at the server node"
```

As a pre-emptive action, this release includes a switch to delete and disable updating the DNS$CHDIRECTORIES attribute. This attribute is read-only and it in no way affects the proper running of a server.

To set the switch, create a file called DNS.CONF in the SYS$SYSDEVICE:[DNS$SERVER] directory and add the following line:

```
dnsd.chdirectories_setting: #
```

where # can be:

0 : Off - The current behavior of updating the DNS$CHDIRECTORIES attribute.
1 : Delete - Delete the DNS$CHDIRECTORIES attribute and never update it.
2 : Auto - Update DNS$CHDIRECTORIES until a size threshold is reached, then delete it.

Since the DNS.CONF file gets read once during server startup, you need to stop and then restart the server to change the setting.

6.16 Message Text Change–BADCLOCK

The text of the BADCLOCK error has been changed from "Server clocks not synchronized" to "Distributed update contained an invalid timestamp". Although unsynchronized clocks can certainly generate BADCLOCK errors, the problem may be the result of another problem (or the clocks may have been fixed after DECdns detected the problem).

The SYS$MANAGER:DNS$SERVER.LOG file often contains additional information about the BADCLOCK error.

6.17 New DECdns Error Message–DNS-E-LOCALNAMEABBR

There is a new error message, "DNS-E-LOCALNAMEABBR, Bad local name abbreviation translation."

Explanation: A local name abbreviation translation is invalid because it contains invalid characters or because it causes a translation loop.

User Action: For OpenVMS, review the logical names in the DNS$SYSTEM logical name table, where local name abbreviation translation is done. For DECnet–ULTRIX, review the /usr/var/dss/dns/dns-names file, which contains the local name abbreviation translations.

6.18 Size and Creation of Clerk Cache File

The DECdns clerk has been modified so that it no longer resizes the clerk cache file every time DECdns is started. The DECdns clerk now calculates the correct size for the cache file only when it finds no cache file and must create a new one.

If the amount of physical memory available to a system has changed or if the GBLPAGFIL system parameter has been modified, check the SYS$MANAGER:DNS$ADVER_ERROR.LOG file. The DECdns clerk indicates in this file if it has calculated a new recommended cache size.

When you see the message "Insufficient Global Page File Limit - no cache", there were fewer than 10 GBLPAGFILs available and the cache file was not created. When this happens, you need to increase GBLPAGFIL, run AUTOGEN, and reboot your system to get a functioning DECdns clerk.

The formula to determine the size of the cache file is as follows:

SIZE (in blocks) = MIN (1000, .5% Total memory)

If SIZE exceeds 75 percent of the available GBLPAGFIL, then it is set to that figure, so as to not use up all of the available GBLPAGFIL. For OpenVMS systems, the maximum SIZE value is 512 MB.

If the recommended change in the size of the cache file is substantial and you want to have DECdns use the new cache size, then follow these steps:

1. Shut down DECdns.
2. Delete the existing cache files (SYS$SYSTEM:DNS$CACHE.*).
3. Reboot the system. Since the cache sizing algorithm must run on a fresh boot of OpenVMS, it is important to reboot the system without first restarting DECdns.
4. DECdns can be started during the reboot or at any time thereafter.

The first time DECdns Version 2 runs on a system (or if DECdns runs and finds the cache file missing), the advertiser creates the file SYS$SYSTEM:DNS$CACHE.0000000001 (if it does not already exist). This file is the backing store file for the DECdns clerk cache. The backing store update interval is 30 minutes. The extension part of the file name (0000000001) is incremented by 1 at each interval and is updated in the associated file SYS$SYSTEM:DNS$CACHE.VERSION. If you want to start with an empty cache, delete both files with the following command before starting the clerk:

```
$ DELETE SYS$SYSTEM:DNS$CACHE.*
```

6.19 Removing Obsolete DNS$CACHE Files

Multiple obsolete copies of the DECdns clerk cache backing store file (SYS$SYSTEM:DNS$CACHE.000000000n) can, under unusual circumstances, accumulate and cause disk space problems on the system. The backing store file is updated every 30 minutes, at which time the extension part of the backing store file name (000000000n) is incremented by 1 and the backing store file name is updated in the associated file (SYS$SYSTEM:DNS$CACHE.VERSION). DECdns uses only the DNS$CACHE.VERSION file and the one or two DNS$CACHE.000000000n files referenced in DNS$CACHE.VERSION. DECdns normally deletes prior unreferenced versions of the file.

If you perform a directory of the SYS$SYSTEM: directory and see more than one backing store file, type the DNS$CACHE.VERSION file to see which backing store files DECdns is currently using and delete all prior DNS$CACHE.000000000n files from the directory.

6.20 Clarification of WORLD Access in DNS and DECdns

The DNS Version 1 principal expression *::* and the DECdns Version 2 principal expression .*... permit access to any user on any node, but *only* within the particular namespace in which the ACEs containing these expressions were created.

Because users are not required to enter a namespace nickname as part of a principal expression and, because DNS Version 1 does not display the namespace nickname associated with the principal expression of ACEs, many DNS Version 1 users have logically assumed that both *::* and .*... can be interpreted as any user, on any node, in any namespace. This is not the case. (See Section 6.14 of these release notes.)

6.21 Default Parameters for Process Limits on DECdns Servers

This section describes several default parameters for process limits on NET$ACP and DNS$SERVER.

- When FILLM is set to 100 on a DECdns server, it limits the number of DECdns clerks that can connect. This causes the DECdns clerk to log a USERREJECT error into the DNS$CHFAIL.LOG file when the limit of 100 connections is exceeded. You can raise this limit by modifying the line in SYS$STARTUP:DNS$SERVER_STARTUP.COM which specifies `/FILLM=100`.
- When the ASTLM on NET$ACP is set to 100, it limits the number of connection setup logical links NET$ACP can process at one time.
- When NET$ACP runs out of ASTs, FILLM on the NET$ACP is closely following ASTLM towards 0. Both parameters must be raised together.
- The VIRTUALPAGCNT limit for nodes with DECdns servers must be approximately 10,000 blocks greater than the DECdns checkpoint file size in blocks. The page files on the node must also be sized accordingly. If the system has more than one page file, the individual page files must be at least as large as the checkpoint file. Since each OpenVMS process is assigned to a single page file, the total combined size of the page files is not useful to DECdns since it is only able to use the capacity of one of them when it reads the entire clearinghouse checkpoint file into memory. Note also that DECdns is not guaranteed the larger page file if one page file is sufficiently large and others are not.

6.21.1 DECdns Servers Support DNS$SkulkStatus Attribute

DECdns servers now support a directory attribute called DNS$SkulkStatus, which provides details on why a directory skulk failed. Use the SHOW REPLICA command, directed at the master replica's clearinghouse, to obtain this information. Information presented in this attribute is also recorded in the server's DNS$SERVER.LOG file.

7
DECdts

7.1 Automatic Time Zone Changes on Rebooting Clusters

If all members of a cluster are down when Daylight Savings Time takes effect, and automatic time zone changes are enabled, then members may reboot with the incorrect local time. If you anticipate that the cluster will be down during the change to or from Daylight Savings Time, you should disable automatic time zone changes and make the changes manually.

7.2 Unknown LAN Device Types

DECdts automatically configures over the following LAN devices:

```
EC*, EF*, ES*, ET*, EX*, EZ*, FC*, FX*, FZ*, XE*, XQ*
```

To configure DECdts over any other device type, use the system logical name DTSS$ETHERNET_DEVICE, where XY defines the new device type:

```
$ DEFINE/SYSTEM DTSS$ETHERNET_DEVICE "XYA0"
```

7.3 Configuring LANs with Global Servers

Unreliable synchronizations can occur on LANs that contain a single global server and one or two additional local servers. The local servers on the same LAN as the global server can contact it twice: once using DECnet and again using LAN protocols. If the SERVERS REQUIRED value for the local servers is set to 3, the global server can contribute as many as two of three values used to compute a new time for a local server, thereby reducing fault tolerance and steering a local server's time.

To prevent this problem, add more local servers to the LAN, so that there is a minimum of four servers, including the global server; then set the SERVERS REQUIRED value to 4 on each server.

7.4 Corrections for CHANGE DTSS Command

The syntax for the NCL CHANGE DTSS command is:

```
ncl> change [node node-id] dtss epoch integer [, time absolute-time]
```

Although the EPOCH argument is required, NCL does not check for its value. If the EPOCH argument is missing, NCL returns an "invalid itemlist" error. If you get this error message, reenter the CHANGE DTSS command and specify the EPOCH argument.

Examples of the CHANGE DTSS command appearing in *DECnet/OSI DECdts Management* that involve changing both a server's epoch and time values are incorrect. The documentation fails to specify that you must include a comma between the EPOCH value and the TIME argument on the command line. The following example shows the correct syntax for this iteration of the CHANGE DTSS EPOCH command:

```
ncl> change dtss epoch 1, time 1991-03-21-16:07:45.00-07:00I0.00
```

7.5 Advertising Global Servers Document Change

There is incomplete information in Section 3.5.2 of the *DECnet/OSI DECdts Management* guide. The passage reads:

```
dns> add directory WAK:.DTSS_GlobalTimeServers -
     access .mynode.DNA$SessCtrl for r,w
```

See Section 7.6 of these release notes for the correct wording.

7.6 Advertising Global Servers

When advertising a DTSS global server into the namespace, you need write access to the directory where the DTSS object is written. The following DECdns commands add access rights to the DTSS GLOBAL SERVER directory.

On DECdns servers at Version 2.0 and later, enter the following commands:

```
dns> add directory WAK:.DTSS_GlobalTimeServers -
     access wak:nodename.DNA$SessCtrl for r,w,t,c

dns> add directory WAK:.DTSS_GlobalTimeServers -
     access wak:nodename.system for r,w,t,c
```

In these examples, "nodename" is the full name of your local system.

On DECdns servers before Version 2.0, enter the following commands:

```
dns> add directory WAK:.DTSS_GlobalTimeServers -
     access wak:dns$iv.nodename.DNA$SessCtrl for r,w,t,c

dns> add directory WAK:.DTSS_GlobalTimeServers -
     access wak:dns$iv.nodename.system for r,w,t,c
```

In these examples, "nodename" is the node synonym of your local system.

7.7 Documentation Addition for SHOW DTSS Local and Global Server Commands

When you enter the SHOW DTSS LOCAL SERVER * command or the SHOW DTSS GLOBAL SERVER * command at a DTSS local or global server, the list does not include the server itself. To verify what DECdts role the current node is running, enter the following command:

```
ncl> show dtss type
```

7.8 Documentation Correction to SHOW DTSS Command Example

The documentation (*DECnet/OSI DECdts Management*) shows an incorrect example of using the SHOW DTSS command to check the value of the SYNCHRONIZATION HOLD DOWN attribute. If you abbreviate the attribute name as shown in the example, (SHOW DTSS SYNC HOLD DOWN), you receive an "unrecognized command" error message. To execute this SHOW command successfully, you must spell out the attribute name as follows:

```
ncl> show dtss synchronization hold down
```

7.9 The DTSS Synchronization Completed Event Is No Longer Blocked by Default

The DTSS event, Synchronization Completed, is no longer blocked by default. (In previous releases this event was blocked during DTSS startup.) The block command has been added to the NET$EVENT_LOCAL.TEMPLATE file. This is consistent with the method used to block other events that may occur frequently.

7.10 Document Change - Interoperation with NTP

The DECdts kit now contains two NTP time-provider programs:

- dtss_ntp_provider.c for use on ULTRIX systems
- DTSS$NTP_PROVIDER.C for use on OpenVMS systems

Appendix C (Interoperation with NTP) in *DECnet/OSI DECdts Management* explains how to use both DECdts and NTP in the same DECnet/OSI for ULTRIX network. The appendix applies as written for ULTRIX systems. With the time-provider program name DTSS$NTP_PROVIDER.C substituted for dtss_ntp_provider.c, the appendix applies for OpenVMS systems.

7.11 Using a Time-Provider on DECdts Server Nodes

On a node with a DECdts server and a time-provider, you must protect the DECdts server against time faults introduced by the time-provider. If the time-provider returns an invalid time, the DECdts server modifies its time incorrectly.

To improve the fault tolerance of the DECdts server, set the system logical name DTSS$_TP_MAXERROR to a value less than 300 seconds. For example:

```
$ DEFINE/SYSTEM DTSS$_TP_MAXERROR 299
```

If the difference between the local time and the value returned by the external clock is greater than the interval represented by DTSS$_TP_MAXERROR, then the DECdts server ignores the value returned by the external clock. The DTSS$_TP_MAXERROR value depends on the type of external clock used. It should not exceed 300 seconds because DECdns requires synchronization to within 300 seconds.

The DECdts interface code examples in SYS$SYSROOT:[SYSHLP.EXAMPLES.DTSS] show how to use this and other time-provider logical names. The ACTS Provider example uses a command line interface rather than OpenVMS logical names.

7.12 Time-Provider Interface (TPI) Advisory

Future versions of DECdts that support additional protocols will use a new TPI. To ease future porting of time-provider programs to new protocol versions, use the sample time-provider programs that are supplied with the kit, with as few modifications as possible.

7.13 Update to List of Supported Radio Receivers

The following note updates information contained in Appendix B, Time-Providers and Time Services, in the *DECdts Management* guide.

Table B-3, Radio Receiver Manufacturers, lists supported radio receivers by manufacturer. Update this list by replacing "Spectracom 8170" with "Spectracom Netclock/2".

8
FTAM

8.1 Installation

Please read these notes and the *DECnet/OSI for OpenVMS Installation and Configuration* manual completely before performing the installation.

8.2 DAP/Gateway

Support now exists for the DAP/FTAM gateway `append` function.

8.3 RMS Record I/O Performance Enhancements

This version of FTAM includes changes to enhance performance with RMS when doing record-oriented I/O. In addition to internal changes, two logical names are now provided which allow some control over initial file creation and extend sizes. These logicals are most useful when you know in advance that a typical file will exceed a certain size, and that FTAM is acting as the receiver of data.

The logicals are:

```
OSIF_FILE_ALQ - controls initial file allocation size (in blocks)
OSIF_FILE_DEQ - controls extension size (in blocks)
```

NOTE: the logicals will be ignored if a `future_filesize` parameter is supplied in the `f-create-request`.

These logicals may be defined in any appropriate location, such as in SYS$SYSTEM:OSIF$RESPONDER.COM. For example:

```
$ define OSIF_FILE_ALQ 1000   ! allocate files initially at 1000 blocks
$ define OSIF_FILE_DEQ  500   ! extend by 500 blocks when needed
```

Limited performance testing has been done with FTAM-1 fixed files, and a performance increase in the range of 2 to 14 times was observed when FTAM was doing record-oriented I/O. This increase is dependent on the platform and other variables. Because so many factors influence performance, your actual observed increases will vary.

8.4 Documentation Corrections

8.4.1 OSAK Start-Up Command File

Section 10.4 in *DECnet/OSI FTAM and Virtual Terminal Use and Management* shows the following incorrect command line:

```
$ @sys$startup:osif$start.com ! osak start-up command file
```

The correct command line is:

```
$ @sys$startup:osak$start.com ! osak start-up command file
```

8.4.2 COPY/DELETE Command

The `COPY/DELETE` command does not currently appear in *DECnet/OSI FTAM and Virtual Terminal Use and Management*. Here is the description of the command.

Qualifier type: Global qualifier

Use of the `/DELETE` qualifier causes FTAM to delete the source file after the source file has been copied to the destination. Note that this qualifier is NOT positional, and that it always applies to the source file, regardless of its location on the command line.

Examples:

```
$ copy/application_protocol=ftam/delete mydata.dat -
        remote_system::remote_file.dat
```

`mydata.dat` is copied to the alias `remote_system` as a file name `remote_file.dat`. After the copy operation is complete, FTAM deletes the local file named `mydata.dat`.

```
$ copy/application_protocol=ftam/delete remote_system::remote_file.dat -
        mydata.dat
```

`remote_file.dat` is copied from the alias `remote_system` to a local file named `mydata.dat`. After the copy operation is complete, FTAM deletes the remote file `remote_file.dat`.

8.5 Size Restriction for FTAM-3 Fixed Files

Both NIST phase 2 and phase 3 agreements and the ISO/IEC ISP 10607-3 have size restrictions. Therefore, if an FTAM-3 fixed file has records larger than 6.5K, it cannot be sent because fixed records cannot be segmented. As a result, save sets that have block sizes greater than 6.5K must be unwound and rewound with block sizes less than 6.5K.

8.6 DTE Address Logical

For X.25 access situations where DTE addresses (not NSAPs) are passed to the X.25 Access module, you must define a logical in the `osit$names` logical name table for the DTE of the peer system. You must then specify this logical in the NSAP field of the alias definition in `sys$system:isoapplications.dat`.

For example, you can define a logical called `REMOTE_DTE` in the `osit$names` table, as follows:

```
$ define/table=osit$names REMOTE_DTE 031346174301212
```

Then you can make the following alias definition in `sys$system:isoapplications.dat` (this one loops back to the same OpenVMS node):

```
LOCAL_FTAM_X25 :FTAM:::RMS.FTAM.OSIF.REMOTE_DTE,provider=osi,
                template=osit$loop_x25:
```

Note

For situations where X.25 NSAPs are passed to the X.25 Access module, insert the NSAP into the alias definition in `sys$system:isoapplications.dat` in the normal NSAP position.

8.7 OpenVMS File Protection Assignment Upon File Creation

The behavior in assigning of FTAM file protection on remote OpenVMS systems has been modified. In the past, when an FTAM user created a remote file on an OpenVMS system, the file protection was assigned by RMS in one of the following ways:

- Using the protection assigned to an existing file of the same name
- Using the default file protection of the directory
- Using the process-default protection

FTAM no longer assigns file protection for the WORLD, GROUP, and SYSTEM access codes.

FTAM maps permitted-action against only the OWNER access code, in the following way:

```
-------------------------------+---------------------------------------------
    FTAM Permitted Action:     |    OpenVMS Filestore Protection (OWNER only)
-------------------------------+---------------------------------------------
    read                       |    read
    read-attribute             |
    traversal                  |
    reverse-traversal          |
    random-order               |
-------------------------------+---------------------------------------------
    insert                     |    write
    replace                    |
    extend                     |
    erase                      |
    change-attribute           |
-------------------------------+---------------------------------------------
    delete-file                |    delete
-------------------------------+---------------------------------------------
```

8.8 File Error Recovery

The docket information maintained to perform an error recovery is now stored in `sys$scratch:osif$recovery.dat`. This file should be truncated or deleted periodically by the FTAM user. This file maintenance should be done when no active FTAM responders are being used as part of FTAM associations.

9

FTAM Application Programming Interface (API)

This chapter discusses information applicable to the FTAM Application Programming Interface (API). As of this version, FTAM includes support for the FTAM API component which was previously shipped as part of the OSI Application Developer's Toolkit.

9.1 Changes for V3.2

The following sections describe changes to FTAM API for V3.2.

9.1.1 OSIF.H osif_ae_entry Structure Changes

With the FTAM V3.2 API, the `osif_ae_entry` structure has been enhanced to provide additional addressing capabilities. Two new fields and their associated data types have been added to the structure as follows:

Data Type	Field Name
osif_nsap_queue	nsap_queue_ptr
osif_template_queue	template_queue_ptr

These changes require that programs written to earlier versions of the API be recompiled and relinked. However, no code changes are necessary unless you want to use these new fields.

9.1.2 FTAM API Version Identification

The FTAM API checks the existing `nsap_queue[0].nsap.length` field of the `osif_ae_entry` structure to determine which API format to use. If the value of `length` is non-zero, then the API assumes that the FTAM V3.0 format is being used, and the new fields in the `osif_ae_entry` structure are ignored. If the value of `length` is zero, then the API assumes that the V3.2 format is being used, the `nsap_queue` array is ignored, and the API looks for information in the new fields.

9.1.3 NSAPs and Templates

The FTAM V3.2 format provides the ability to disassociate specific NSAPs from specific templates (that is, there is no longer a one-to-one correlation between NSAP and template as required by the V3.0 interface). The user is expected to simply provide a list of potential NSAPs, along with the type of network service that each NSAP is expected to use.

The OSAK constants OSAK_C_CLNS, OSAK_C_CONS or OSAK_C_RFC1006 (note that OSAK_C_RFC1006 is only valid for use on DECnet/OSI for Digital UNIX or DECnet/OSI for OpenVMS Version 6.0 or later) may be used to indicate whether the NSAP is appropriate for a CLNS, CONS or RFC1006 network service. In addition to the list of NSAPs, the user also provides a list of potential transport templates. When the FTAM API passes the NSAP and

Template lists to OSAK, OSAK attempts to establish an association with each appropriate NSAP/Template pair.

For example, suppose two NSAPs and two templates are passed:

NSAP List	Template List
%x21 (CLNS)	OSIT$LOOP_CONS
%x22 (CONS)	OSIT$LOOP_CLNS

OSAK matches the first Template in the list with an appropriate NSAP (in this case, the second NSAP in the list), and constructs a final address to attempt an association. Using our example, the address looks something like:

```
OSIT$LOOP_CONS%x22
```

If the association attempt fails with this particular template/NSAP pair, OSAK continues searching the NSAP list looking for another NSAP appropriate for a CONS connection. Once OSAK attempts all possible combinations within the NSAP list for the first template, OSAK then attempts an association with the next template in the template list, repeating the template/NSAP pairing operation until an association is established, or until all valid combinations of template/NSAPs have been attempted.

9.2 Documentation Notes

The *FTAM Programming* manual describes FTAM parameters that are part of attribute groups not supported in the FTAM API code. Parameters for unsupported attribute groups should not be used when programming with the FTAM API. For example, the use of abstract-syntax names and constraint set names causes unknown results and should not be used.

9.3 Restrictions

The following list describes known restrictions.

- The `osif_protocol_error` vector and the `osif_prot_error_count` variable will not be filled in if OSIF_PROTOCOL_ERROR is returned by any function call. OSIF_PROTOCOL_ERROR is used to signal that an error has occurred at a lower layer. The `osif_protocol_error` vector is used to list all the errors that have occurred in the lower layers.
- The FTAM API only supports a buffer list with one buffer. In other words, one P_DATA must be contained in one user buffer. For this release, the size of user buffers passed to the FTAM API must be at least 8K bytes. If the buffer is less than 8K, then the user receives the OSIF_NOBUFFS error for the `osif_get_event` function call.
- If a contents type list is not specified in the F-INITIALIZE-request primitive, the FTAM API sends all the supported document types.
- The checkpoint window parameter will be defaulted to one even though the Recovery functional unit is not supported.
- An error should be returned by the service provider when a universal class number is specified with FTAM-3 files on F-OPEN and F-CREATE requests. Currently, the universal class number information is ignored and no error is returned.

- The osif_fadu_locking parameter of the F-OPEN-request primitive is specified in the documentation and the osif.h file, but it is not used by the FTAM API.

10
Virtual Terminal

10.1 Local Command Mode

The following sections discusses issues specific to local command mode.

10.1.1 CTRL-@

The `SET` command will not accept `CTRL-@` as a valid value for the break, command, or disconnect characters.

10.1.2 Amode Repertoire

The `SEND` command may incorrectly indicate that a character is not in the `Amode-default` repertoire.

10.2 Initiator

The `/trace` option is not supported.

10.3 Responder

The Virtual Terminal responder may stop accepting connections after several connections have already been established and aborted. If you encounter this symptom, you can remedy the problem by stopping and restarting VT. To stop VT, enter:

```
$ @sys$manager:vt_stop.com
```

To start VT, enter:

```
$ @sys$startup:vt_start.com
```

10.4 Gateways

The following sections discuss issues specific to the gateways.

10.4.1 VT/LAT Gateway Will Hang

The VT/LAT gateway will hang if an unknown LAT service is provided at the following prompt:

```
Enter LAT Service name:
```

If the VT/LAT gateway hangs for this reason, it is best to stop the VT/LAT gateways and restart them.

To stop the VT/LAT gateways, enter:

```
$ @sys$startup:vt_stop lat
```

To restart the VT/LAT gateways, enter:

```
$ @sys$startup:vt_start lat
```

10.4.2 Starting VT/LAT and Telnet Gateways (Alpha Only)

To start the gateways while the VT responder is running, execute the following commands.

To start the VT/LAT and LAT/VT gateways, enter:

```
$ @sys$startup:vt_start lat
```

To start the VT/TELNET and TELNET/VT gateways, enter:

```
$ @sys$startup:vt_start telnet
```

10.5 Interoperability with DECnet/OSI Virtual Terminal Version 1.0 for ULTRIX

The following sections discuss interoperability with ULTRIX systems.

10.5.1 SEND VT-BREAK

A Virtual Terminal association with a VAX ULTRIX system will hang if you attempt to log out after issuing a `SEND VT-BREAK` command. The association can be terminated by using the `QUIT` command from Local Command Mode.

A Virtual Terminal responder on MIPS ULTRIX will abort the Virtual Terminal connection if it receives an incoming vt-break request.

10.5.2 SEND SYNCH

The `SEND SYNCH` command will not work over a Virtual Terminal association with a VAX ULTRIX system. The following message displays instead:

```
%OSAK-E-INVFUNC, the call is invalid
```

10.6 Interoperability with DECnet/OSI Virtual Terminal Version 1.0 for OpenVMS VAX

The following sections discuss interoperability with OpenVMS VAX systems.

10.6.1 SEND SYNCH

DECnet/OSI Virtual Terminal Version 1.0 for OpenVMS VAX does not properly handle updates to the SY control object. The DECnet/OSI Virtual Terminal Version 1.0 for OpenVMS VAX responder will abort the Virtual Terminal Connection if it receives an SY control object update.

10.6.2 VT-BREAK

DECnet/OSI Virtual Terminal Version 1.0 for OpenVMS VAX does not support the vt-break service as a responder. The DECnet/OSI Virtual Terminal Version 1.0 for OpenVMS VAX responder will abort the Virtual Terminal association if it receives a vt-break indication.

11
OSAK

This chapter describes issues specific to this version of the DECnet/OSI OSAK software.

11.1 New features

This section contains information about new features in the OSAK software and programming interfaces.

11.1.1 OSAK Application Programming Interface (API)

This version of the DECnet/OSI OSAK software includes support for the OSAK API, which previously shipped in the OSI Application Developer's Toolkit.

11.1.2 New OSI Session Programming Interface

This version of the software includes a new programming interface to the OSI session layer (the SPI). This interface is a replacement for the existing backwards-compatible interface to OSAK Version 1.1.

More information about this interface is given in the books *OSAK Programming* and *OSAK SPI Programming Reference*.

11.2 Known Problems and Restrictions

This section contains details of known problems and restrictions in the OSAK software and the OSAK programming interfaces.

11.2.1 OSAK Software

11.2.1.1 Session Disconnect Timer Not Supported

This version of the OSAK software does not support the session disconnect timer.

11.2.1.2 NCL Does Not Display Form 1 AE-Titles Correctly

NCL does not display a Form 1 AE-Title correctly. Instead, it displays an object identifier.

11.2.1.3 Enumerated Data Types Not Decoded

The OSAKtrace utility does not decode enumerated data types.

11.2.1.4 OSAKtrace Output Alignment Problem

The trace analyzer does not correctly align its output when tracing an AE-title in Form 1 format. However, no information is missing from the output.

11.2.2 OSAK API and SPI Programming Interfaces

11.2.2.1 Incorrect Decoding of SET

The OSAK API interface does not always correctly decode the mode selector SET in a CP-PPDU (A-ASSOCIATE indication) or a CPA-PPDU (A-ASSOCIATE-ACCEPT confirm).

If [0] IMPLICIT Mode-selector comes after [2] IMPLICIT SEQUENCE the OSAK interface does not decode the mode selector, but passes it to your application as user data.

11.2.2.2 Session Segmentation Not Supported

This version of the OSAK interface does not support session segmentation. This does not hinder segmentation by the user. For an explanation of the two sorts of segmentation, see the manual *OSAK Programming*.

11.2.2.3 Multiple Upper Layer Addresses Not Allowed on a Transport Selector

You should not use the same transport selector (TSEL) on more than one process. Any TSEL that your application uses should be unique. Re-use of a TSEL results in one of two secondary status codes in the status_block parameter. The primary status in each case is OSAK_S_INVAEI. The two possible secondary codes are:

- OSAK_S_TSELINUSE

  Indicates that a TSEL in the local_aei or calling_aei parameter is already being used on another port or by another application.

- OSAK_S_MULTADDR, multiple upper layer addresses for t-selector

  Indicates that you have opened more than one OSAK responder port within the same process using the same TSEL, but a different SSEL or PSEL. This is not allowed. If you want to specify a different SSEL or PSEL, you should also specify a different TSEL.

11.2.2.4 Support for Programming Languages

The OSAK interface includes data type definitions for all supported programming languages, but only provides language bindings for the C programming language.

11.2.2.5 Use of ASCII Strings in NSAPs

If you need to use an ASCII string in an NSAP you should define the string as a logical name in the table OSIT$NAMES and then pass the logical name as an input parameter in the call to the routines `osak_associate_req` or `spi_connect_req`. NSAPs passed to the OSAK interface directly in the call to the routines `osak_associate_req` or `spi_connect_req` must be in hexadecimal format.

11.2.2.6 Receiving Odd Length NSAPs

If the OSAK interface passes to the application an NSAP preceded by a zero, the zero can be ignored. This additional digit is added during the translation process if a NSAP has an odd number of characters.