DEComni API for OpenVMS

Installation Guide

Order Number: AA–Q75EC–TE

April 1997

This manual describes the prerequisites and procedures for installing the DEComni Application Programming Interface (API) software on OpenVMS platforms.

Revision/Update:	This is a new document for the current release.
Operating System:	OpenVMS Version 6.1, 6.2, 7.0, 7.1
Software Version:	DEComni API Version 3.1

Digital Equipment Corporation
Maynard, Massachusetts

First Printing, July 1996
Revised, April 1997

The postpaid Reader's Comments forms at the end of this document request your critical evaluation to assist in preparing future documentation.

This document was prepared using VAX DOCUMENT, Version 2.1.

Contents

A Dialogue Samples

B Files Added During Installation

Index

Tables

Preface

This document covers the installation of the DEComni API (Application Programming Interface) software on a OpenVMS system. Keep this document with your distribution kit. You may need it to install maintenance updates or to reinstall the DEComni API software.

For more information about DEComni API features and functions, refer to the DEComni API Software Product Description (SPD) in your software distribution kit.

Intended Audience

This document is for system and network managers and DIGITAL personnel who must plan or perform an installation of the DEComni API software on a OpenVMS system.

Document Structure

This manual is organized as follows:

- Chapter 1, Pre-installation Requirements, describes the hardware and software requirements that must be met before you can install the DEComni API software.
- Chapter 2, Installing DEComni API on OpenVMS Systems, describes the system environment required to install and run the DEComni API software. This chapter also provides the DEComni API installation procedures and guidelines.
- Chapter 3, Post-installation Considerations, describes how to start or stop the DEComni API process and access the release notes after the installation.The chapter also provides more information about the installed product.
- Appendix A, Dialogue Samples, provides a sample installation dialogue.
- Appendix B, Files Added During Installation, lists directories and files added to the system during the installation process.

Associated DEComni Documents

The following documents provide detailed information about using the DEComni API system:

- *DEComni API and DEComni MMS User Guide*
- *Guide to Using Omni Directory Services*
- *DEComni API Omni Definition Facility User Guide*
- *DEComni API Guide to Using OmniView*

Related ISO/IEC Documents

The following documents provide information about the ISO/IEC standard Manufacturing Message Specification (MMS):

> *Industrial Automation Systems — Manufacturing Message Specification Service Definition*, ISO/IEC 9506–1
> *Industrial Automation Systems — Manufacturing Message Specification Protocol Specification*, ISO/IEC 9506–2

Related System Documents

See your OpenVMS system documentation for information about software installation and licensing procedures.

Conventions

This manual uses the following conventions:

Convention	Meaning
DEComni	The term ''DEComni'' refers to the DEComni API product or to functions and services provided by the DEComni API software.
OpenVMS	The term ''OpenVMS'' refers to the OpenVMS product or to operations and functions performed by the OpenVMS operating system.
#	The pound sign is the default OpenVMS superuser prompt.
NODE>	The OpenVMS user prompt is the system name followed by a right angle bracket (>).
UPPERCASE, lowercase	The system differentiates between uppercase and lowercase characters. Literal strings that appear in descriptions, examples, or command syntax must be entered exactly as shown.
`Boldface Type`	This bold typeface emphasizes user input to system prompts.
`system output`	This typeface indicates system output in interactive examples. In OpenVMS descriptions, it also specifies the exact name of a command, option, partition, pathname, directory, or file.
`setld(8)`	Cross-references to online reference pages include the section number in parentheses. For example, `setld(8)` indicates that the information can be found on the `setld` command in section 8 of the reference pages.
/ /	Slash marks are part of the directory specification /`directory_name`/ on OpenVMS systems.
Ctrl/x	Indicates to hold down the Ctrl key while you press another key, indicated here by x.
n	A lowercase italic *n* indicates the generic use of a number.
x	A lowercase italic *x* indicates the generic use of a letter.
italic type	Italic type emphasizes important or unique information or the complete titles of documents.

Convention	Meaning
. . .	Vertical ellipses (dots) in examples represent data that has been omitted for clarity.

1 Pre-installation Requirements

This chapter lists the minimum system hardware and software environment required to install and run the DEComni API software on OpenVMS.

Verify that the installation kit contains all items listed on the Bill of Materials (BOM). Contact your DIGITAL representative if there are any problems or missing items.

The software distribution kit generally contains the following items:

- Installation media (CD–ROM) containing the DEComni API installation software
- Product Authorization Key (PAK) for registering the DEComni API software
- DEComni API installation and user documentation
- Online release notes are included on the installation media. Directions for accessing the release notes are provided in Section 2.1.5.

Note

DIGITAL recommends that you read the release notes before installing or running the DEComni API software.

1.1 System Hardware Requirements

You require a minimum hardware configuration to perform the installation as described in the DEComni Software Product Description (SPD). This minimum configuration is as follows:

- A supported VAX or Alpha processor (see the DEComni API *Software Product Description* for details)
- An online terminal with access to the system.

Refer to the DEComni API SPD to determine whether there are additional hardware requirements that apply to your application.

1.2 System Software Requirements

Your system requires the following minimum software versions:

- **Mandatory:**
  - OpenVMS Version 6.1, 6.2, 7.0 or 7.1

In order to successfully run DECcomni API Version 3.1 on OpenVMS Version 6.2, you must install a mandatory patch for the CMA (Common Multithread Architecture). The name of the CMA patch kit is as follows:

ALPCMAR04_062.A for OpenVMS Alpha
VAXCMAR04_062.A for OpenVMS VAX

- **Optional:**
  - DECwindows Motif Version 1.2—only necessary if you want to use OmniView
  - ORACLE Rdb Version 5.1 (VAX), or Version 6.0 or 6.1 (Alpha)—only necessary if you want to use the Omni Definition Facility.

Pre-requisite software and software versions may change. Refer to the current DECcomni API *Software Product Description* (SPD) for the latest pre-requisite information. The following command displays the OpenVMS system version you are running:

```
$ WRITE SYS$OUTPUT F$GETSYI("VERSION")
```

1.3 License Registration

You must use the License Management Facility (LMF) to register the DECcomni MMS software license in accordance with the license agreement signed for your site before you can run either the Installation Verification Procedure (IVP) or the software. License registration information is included in the Product Authorization Key (PAK) that is shipped with the DECcomni software. The PAK contains licensing information that should be registered *before* you start the installation.

To register the license, log into the SYSTEM account and set your default to SYS$UPDATE. You can register the license under LMF using either of the following two methods:

- Invoke the following procedure and enter the data supplied by the PAK:

  ```
  $ SYS$UPDATE:VMSLICENSE.COM
  ```

- Enter the following command, using the qualifiers that specify the data supplied by the PAK:

  ```
  $ LICENSE REGISTER DECOMNI-API/qualifier,...
  ```

If you plan to use the DECcomni software on more than one node of a VAXcluster system, you must load the license on each of the other nodes after you have completed this one.

See the *OpenVMS License Management Utility Manual* for more information about using LMF.

1.4 Installation Time

Installing DECcomni from CD–ROM on your target system takes approximately 2 minutes, depending on the system and media configuration. Installation may take longer on a system that is dependent on network activity during the installation.

1.5 Installation Privileges

You require either the SETPRV privilege or all of the following privileges:

CMKRNL
WORLD
SYSPRV

The following command verifies your installation privileges:

```
$ SHOW PROCESS /PRIVILEGES
```

1.6 Required Disk Space

The following free disk space is required on OpenVMS systems:

Blocks to Install	Blocks to Run	DEComni-specific Files	ODS Directory Structure
45,000	40,000	35,000	3,500

The following command verifies the available disk space:

```
$ SHOW DEVICE SYS$SYSDEVICE
```

A feature of the DEComni API installation process allows you to locate the Omni Directory Services (ODS) directory structure on a disk other than the system disk. If you install the ODS directory structure on a separate disk, make sure that the disk has adequate free space.

1.7 Omni Directory Services Directory Structure File Location in a VAXcluster Environment

During the DEComni API installation process, you are asked where you want to locate the Omni Directory Services (ODS) directory structure. If you choose the default SYS$SYSDEVICE, the ODS directory is common to all nodes in the VAXcluster. If you want a separate ODS directory structure for each node, locate the directory in SYS$SPECIFIC. The directory structure requires 3,000 blocks for each node in the cluster.

1.8 System Parameters and Settings

The DEComni API installation requires specific system parameter settings.

1.8.1 Displaying System Parameters

The following minimum system parameter values are required for the installation and to accommodate the space requirements for DEComni API data structures:

FREE_GBLPAGES = 1000
FREE_GBLSECTS = 15

You can determine the current number of free global pages and free global sections with the following commands:

```
$ WRITE SYS$OUTPUT F$GETSYI("FREE_GBLPAGES")
$ WRITE SYS$OUTPUT F$GETSYI("FREE_GBLSECTS")
```

The system displays the current values associated with the FREE_GBLPAGES and FREE_GBLSECTS parameter.

For additional information about these parameters, or for help when verifying the values, see the *OpenVMS System Manager's Manual* for information about the OpenVMS System Generation Utility (SYSGEN).

- If you need to change one or more system parameters, continue with Section 1.8.2.
- If you do not need to change system parameters, proceed to Section 1.9.

1.8.2 Changing GBLPAGES and GBLSECTIONS with AUTOGEN

If you need to change the number of global pages or global sections, invoke AUTOGEN after you have entered the new parameter values in the SYS$SYSTEM:MODPARAMS.DAT file. AUTOGEN adjusts values for the parameters associated with the values that you reset manually.

To change system parameters with AUTOGEN:

1. Use an editor to access the SYS$SYSTEM:MODPARAMS.DAT file.
   - To change a parameter value that is already listed in the file, delete the current value associated with that parameter and enter the new value.
   - To modify incremental parameters, such as GBLPAGES or GBLSECTIONS, use the ADD_ command. The following example increases the global page setting by 380:

     ADD_GBLPAGES = 380
2. After you have made all of your changes, exit from the editor.
3. Invoke the AUTOGEN procedure with the following command to recalculate your system parameters:

   ```
   $ @SYS$UPDATE:AUTOGEN GETDATA REBOOT [NOFEEDBACK]
   ```

   If you specify REBOOT, AUTOGEN performs a system shutdown and reboots the system when it has finished. Any users logged in on the system are immediately disconnected during the shutdown. The reboot then puts the new parameter values into effect.

   The AUTOGEN utility adjusts some SYSGEN parameters based on the consumption of resources since the last reboot. If you do not want to use this adjustment, include the NOFEEDBACK parameter in the command line.

See the *OpenVMS System Manager's Manual* for more information about using AUTOGEN.

1.9 Installation Account Process Quotas

You require the following minimum process quotas for the installation account (typically the SYSTEM account):

ASTLM	=	24
BIOLM	=	18
BYTLM	=	32768
DIOLM	=	18
ENQLM	=	2000
FILLM	=	50
PGFLQUO	=	30000
PRCLM	=	1+

WSDEF	=	256
WSEXTENT	=	3072
WSQUO	=	768

You can use the AUTHORIZE utility to display installation process quotas by entering the following commands:

```
$ SET DEFAULT SYS$SYSTEM
$ RUN AUTHORIZE
UAF> SHOW SYSTEM
```

If no changes are necessary, enter the EXIT command to leave the AUTHORIZE utility.

To modify system account quotas (BYTLM, for example):

```
UAF> MODIFY SYSTEM /BYTLM=32768
```

The utility responds with a message confirming the change and you can enter the EXIT command to leave the utility. You must then log out and log in again to reset the system account quotas.

For further information about using AUTHORIZE, refer to the *OpenVMS System Manager's Manual*.

1.10 Backing Up the System Disk

DIGITAL recommends that you do a full system disk backup before you start any installation procedures. Use the backup procedures established for your site. For information about performing a system disk backup, refer to your OpenVMS documentation.

2 Installing DEComni API on OpenVMS Systems

This chapter provides the requirements and procedures for installing the DEComni Application Programming Interface (API) software on a OpenVMS system.

2.1 Installation Considerations

This section provides information you may require during the installation process.

2.1.1 DASware Installation

The DEComni API Version 3.1 installation procedure allows you to install DASware (DASTKRUN, DASTK232).

DASTKRUN and DASTK232 provide extensions to DEComni API, as required by some device access systems. On OpenVMS, DASware is not required by the following protocols:

- MMS
- AP
- H1
- Allen-Bradley

You can therefore skip the DASware installation if you intend to use one of the suggested Device Access systems.

The pre-requisites are as follows:

- DASTKRUN—DEComni API
- DASTK232—DASTKRUN.

For further information, refer to the *BASEstar Open Device Connectivity DASware User Guide*.

2.1.2 VMSINSTAL Requirements

When you invoke VMSINSTAL, it verifies whether you have:

- Logged into a privileged account
- Adequate quotas for the installation
- Other users logged in on the system
- Minimum SYSTEM account quotas listed in Section 1.9

If VMSINSTAL detects any errors during the installation, it notifies you and exits from the installation procedure. You can restart the installation after you have corrected the problem.

2.1.3 VMSINSTAL Error Messages

If the installation fails for any reason, either during the installation or the IVP, VMSINSTAL displays a message similar to the following:

```
%VMSINSTAL_E_INSFAIL, The installation of OMNIAPI V3.1 has failed.
```

Refer to the *OpenVMS System Management Utilities Reference Manual* for information about VMSINSTAL error messages.

If it is possible to correct the problem, repeat the installation from the beginning. Otherwise, follow the procedures provided in Chapter 3.

2.1.4 Stopping the Installation

You can press Ctrl/Y to abort the installation at any time. You can also enter a question mark (?) at any time for additional information. To restart the installation after pressing Ctrl/Y, you must repeat the entire procedure.

2.1.5 Accessing the Release Notes

DEComni API release notes are in an online ASCII text file that you can read or print before or after the installation. The release notes contain information about known problems or restrictions that apply to the current version of the product.

Note

DIGITAL recommends that you read the release notes before you install or run the DEComni API software.

To display or print the release notes, use the OPTIONS N parameter when you invoke VMSINSTAL for the installation in Section 2.2. The OPTIONS N parameter also enables you to exit the installation procedure after reading the release notes.

VMSINSTAL copies the release notes to SYS$HELP by default. The release notes file installed by VMSINSTAL is labelled with the current product name and version number, such as OMNIAPI031.RELEASE_NOTES.

2.2 Installation Procedure

This section describes the procedures for installing the DEComni API software. To perform the installation, you must be logged in on the system manager's account, or an account with the appropriate privileges.

2.2.1 Invoking VMSINSTAL

Either of the following commands invokes VMSINSTAL for the installation:

```
$ @SYS$UPDATE:VMSINSTAL
```

```
$ @SYS$UPDATE:VMSINSTAL OMNIAPIA031 ddcu:
OPTIONS N
```

Here, *ddcu:* is the device name, where *dd* is the device code, *c* is the controller letter, and *u* is the drive unit number of one or more digits (for example, MUA100:).

Using OPTIONS N allows you to display or print the release notes before the installation. You can then continue the installation or exit from the procedure.

2.2.2 Installation Process

VMSINSTAL starts the installation process with the display of messages and prompts similar to the following:

1. The VMSINSTAL process starts by displaying a message similar to the following:

   ```
   OpenVMS ALPHA Software Product Installation Procedure V7.1

   It is 20-FEB-1997 at 11:43.

   Enter a question mark (?) at any time for help.
   ```

2. The process then asks if you are satisfied with your system disk backup.

   Note

   System disk backup is a prerequisite for the installation.

   If you are not satisfied with the disk backup, enter NO to terminate the installation and perform the backup. Otherwise, enter YES or press Return to begin the installation process:

   ```
   * Are you satisfied with the backup of your system disk [YES]? YES

   The following products will be processed:

     OMNIAPI V3.1

           Beginning installation of OMNIAPI V3.1 at 11:43

   %VMSINSTAL-I-RESTORE, Restoring product save set A ...
   %VMSINSTAL-I-RELMOVED, Product's release notes have been moved to SYS$HELP.

   © Digital Equipment Corporation. 1997. All Rights Reserved.

   Installation procedure for: DEComni for OpenVMS Alpha (TM) V3.1
   ```

3. If you used OPTIONS N in your VMSINSTAL command, the process prompts you for a release notes option number that executes one of the following:

   ```
   1 - Display release notes

       Displays the online release notes.

   2 - Print release notes

       The process prompts you for an ASCII text printer queue name.  You
       can  enter the queue name or press Return to send the file to your
       default print device.

   3 - Both 1 and 2

       Displays  the release notes online.   If you terminate the  display
       with Ctrl/C, the process prompts you for a printer queue name.  You
       can enter the queue name, or you can press Return to send the  file
       to the default print device.

   4 - None of the above
   ```

   When the selected operation has completed, VMSINSTAL copies the release notes to SYS$HELP.

   Note

   DIGITAL recommends that you read the release notes before you continue the installation.

   ---

4. The process asks you which products you want to install:

```
Select which products you want to install:

      1 DEComni API
      2) DASTKRUN
      3) DASTK 232
      4) ALL of the above

      **********************************************************

* Enter number(s): [1]:)YES
```

5. The process then displays the product identification and authorization key prompt:

```
       Product:       DECOMNI
        Producer:     DEC
        Version:      3.1
        Release Date: 15-MAY-1997

* Does this product have an authorization key registered and loaded?
YES
```

The DEComni API software can be installed but not run unless the Product Authorization Key (PAK) is registered. Enter YES to the authorization key prompt if you have registered and loaded the DEComni API PAK.

Enter NO if you have not registered your PAK. The installation continues but you cannot run either the Installation Verification Procedure (IVP) or the software. To run the IVP after the installation, you must register and load the PAK, then follow the IVP instructions in Section 3.1.

6. If VMSINSTAL finds a DEComni API Version 2.*n* database, you can continue to use that database or create a new one. Normally, this would only occur if you are reinstalling Version 2.2. The system displays a message and prompt similar to the following:

```
        ********************************************************
        DEComni Version 2.n is installed  on this system. It  is
        recommended that the database files be renamed or deleted
        if you do not wish to use the existing database.
        If you choose not to rename or  delete the database,  the
        existing database will continue to be used.
        ********************************************************

                 1. Rename existing database
                 2. Continue to use existing database

* Enter a number [2]: 1
```

7. The process enquires whether you wish to use the OMNI_ODF facility. If you answer YES, and if a proper version of RDB is active on the system, VMSINSTALL links the DEComni API shareable to the RDB shareable image and restores OMNI_ODF.EXE in SYS$COMMON:[SYSEXE].

```
* Do you want to use OMNI_ODF facility? [YES]?
```

8. VMSINSTAL then asks for the disk location where you want to install the Omni Directory Services (ODS) directory structure:

```
 A directory for Omni Directory Services will be created. You
 need to provide the disk name in which the directory will be
 located. If you do not specify a location, the directory will
 be placed in SYS$SYSDEVICE:[OMNI...]. Include the colon after the
 disk name.

* Enter the disk now [ddcu:] :
```

You can enter a new disk or press Return and use the default disk you specified in step 1.

Note

In a VAXcluster environment, specifying SYS$SPECIFIC locates the ODS directory structure in a location that can only be accessed by the node on which the installation is being performed.

---

9. VMSINSTAL displays the following questions about purging the files replaced by the installation, and whether you want to run the IVP as part of the installation:

```
* Do you want to purge files replaced by this installation [YES]? YES

* Do you want to run the IVP after the installation [YES]? YES
```

10. If a Multiversion product is installed, VMSINSTAL then displays the current Rdb/VMS environment:

```
*******************************************************
DEComni has determined that the Current PROCESS AXP
Rdb/VMS environment is version n.n (MULTIVERSION)
*******************************************************
```

11. The installation process then begins and displays a series of informational messages that include the following:

```
*****************************************************************

  This installation will now continue in an unattended mode
  of operation, which will not require any user interaction.

  Estimated time to completion from this point is 5 minutes.
  This estimate is based on a configuration consisting of a
  DEC 3000 Model 500 and installation from a CD Rom.

  The current time is: 11:45.47

*****************************************************************

%VMSINSTAL-I-RESTORE, Restoring product save set B ...

* Now building the Version 3.1 database...

%VMSINSTAL-I-SYSDIR, This product creates system directory [SYSHLP.EXAMPLES.
 OMNI].
    .
    .
    .

* If you would like to have DEComni for VMS startup during the system
  startup,  add the following line to SYS$STARTUP:SYSTARTUP_VMS.COM.

                    $ @SYS$STARTUP:OMNI_STARTUP.COM
```

```
%VMSINSTAL-I-MOVEFILES, Files will now be moved to their target
   directories...
    .
    .
    .
```

12. If you entered YES to run the IVP, the process displays informational messages about the progress of the installation, including the DEComni API and IVP startups.

    Appendix A provides a complete installation dialogue sample, including the DEComni API startup and IVP.

13. The process then displays a series of messages about the success of the installation, then exits from the VMSINSTAL utility:

```
Now checking that files have reached target directories...

%OMNI-S-IVP, THE DECOMNI IVP HAS SUCCESSFULLY COMPLETED

        Installation of OMNIAPI V3.1 completed at 11:51

    Adding history entry in VMI$ROOT:[SYSUPD]VMSINSTAL.HISTORY

    Creating installation data file: VMI$ROOT:[SYSUPD]OMNIAPI031.VMI_DATA

        VMSINSTAL procedure done at 11:56
```

3

Post-installation Considerations

This chapter explains how to start, verify, and stop the DEComni Application Programming Interface (API) software, and provides account process information.

3.1 Running the Installation Verification Procedure

If you did not run the Installation Verification Procedure (IVP) as part of the installation, you should run it separately after you have installed and started the product.

You must enable the following privileges to invoke the IVP:

NETMBX
PRMMBX
SYSLCK
WORLD
GROUP

Enter the following command to run the IVP:

```
$ @SYS$TEST:OMNI_IVP
```

The DEComni IVP verifies that the DEComni system is correctly installed and displays informational messages as it progresses. If the IVP fails, it displays a message indicating the reason for the failure. You should correct the problem, then reexecute the IVP. When the DEComni IVP has executed successfully, the DEComni API software is ready for use.

The DEComni IVP can fail for various reasons. Table 3–1 lists the possible causes for an IVP failure and ways to resolve the problem.

Table 3–1 Resolving DEComni IVP Failures

Possible Cause	Resolution
Error while checking process quotas	See the required process quotas in Section 1.9.
Error while checking process privileges	See the required process privileges in this section.
Missing files error	See Appendix B for the list of files.

3.2 Starting DEComni API

If you do not invoke the IVP during installation, you must use the following command to start DEComni API (before running any application program):

```
$ @SYS$STARTUP:OMNI_STARTUP FULL
```

Otherwise, you can execute the following command to start up DEComni API:

```
$ @SYS$STARTUP:OMNI_STARTUP [AST]
```

You must specify the AST parameter if you wish to use the AST-based version of DECOmni API, rather than the thread-based version.

3.3 Stopping DEComni API

You require the following privileges to invoke the DEComni API stop procedure:

DETACH
PRMMBX
SYSNAM
CMKRNL
SYSGBL
PRMGBL
WORLD
GROUP

Enter the following command to manually shut down the DEComni process:

```
$ @SYS$MANAGER:OMNI_STOP
```

If you want the DEComni process to shut down each time the system is brought down, include the command in the site-specific shutdown file SYS$MANAGER:SYSHUTDWN.COM.

3.4 Example Programs

The SYS$COMMON:[SYSHLP.EXAMPLES.OMNI] directory contains some examples of DEComni API usage.

3.5 Accessing the Release Notes

You can consult the DEComni API release notes at any time after the installation from the SYS$HELP: directory. VMSINSTAL labels the release notes file with the current product name and version number, such as OMNIAPI031.RELEASE_NOTES.

3.6 Getting Help and Reporting Problems

If an error occurs while using the DEComni API, and you believe that the error is the result of a problem associated with the product, take one of the following actions:

- If you have a basic or DECsupport Software Agreement, call your Customer Support Center (CSC). The CSC provides telephone support for high-level advisory and remedial assistance.
- If you have a Self-Maintenance Software Agreement, you can submit a Software Performance Report (SPR).
- If you purchased DEComni within the last 90 days, and you think the problem is caused by a software error, you can submit an SPR.

If you submit an SPR, please take the following steps:

- Describe as accurately as possible the circumstances and state of the system when the problem occurred. Include the description and version number of DEComni that you were using. Demonstrate the problem with specific examples.
- Reduce the problem to as small a size as possible.

- Remember to include listings of any command files, include files, relevant data files, and so forth.
- Provide a listing of the program.
- If the program is longer than 50 lines, submit a copy of it on machine-readable media (floppy diskette or magnetic tape). If necessary, also submit a copy of the program library used to build the application.
- Report only one problem per SPR. This will facilitate a faster response.
- Mail the SPR package to DIGITAL.

Experience shows that many SPRs do not contain enough information to duplicate or identify the problem. Concise, complete information helps DIGITAL give accurate and timely service to software problems.

If you find an error in the DEComni documentation, send an electronic mail message to *manufacturing@digital.com*.

A
Dialogue Samples

This appendix provides sample installation sessions from a OpenVMS system using disk media. The appendix includes the following:

- Installation Dialogue Sample
- Installation Verification Procedure Dialogue Sample

A.1 Installation Dialogue Sample

The following example shows an installation from disk media:

```
$ @sys$update:vmsinstal OMNIAPIA031 $1$DKB0:[OMNIAPI31]

        OpenVMS AXP Software Product Installation Procedure V7.1

It is 14-MAY-1997 at 10:59.

Enter a question mark (?) at any time for help.

%VMSINSTAL-W-ACTIVE, The following processes are still active:

       _RTA7:

* Do you want to continue anyway [NO]? y
* Are you satisfied with the backup of your system disk [YES]?

The following products will be processed:
  OMNIAPIA V3.1

            Beginning installation of OMNIAPIA V3.1 at 11:00

%VMSINSTAL-I-RESTORE, Restoring product save set A ...
%VMSINSTAL-I-RELMOVED, Product's release notes have been moved to SYS$HELP.

© Digital Equipment Corporation. 1997. All Rights Reserved.

Installation procedure for: DEComni API for OpenVMS Alpha (TM) V3.1

      **********************************************************

      Select which products you want to install:

      1) DEComni API
      2) DASTkRUN
      3) DASTk 232
      4) ALL of the above

      **********************************************************

* Enter number(s): [1]: 4

        Product:      DECOMNI
        Producer:     DEC
        Version:      3.1
        Release Date: 15-APR-1997

* Does this product have an authorization key registered and loaded? y
```

```
      **********************************************************
      DEComni API V3.1 is installed on this system.
      It is recommended that the database files be renamed or
      deleted if you do not wishto use the existing database.
      If you choose not to rename or delete the database,the
      existing database will conï tinue to be used.
      **********************************************************

               1. Rename existing database
               2. Continue to use existing database

* Enter a number [2]: <CR>

 A directory for Omni Directory Services will be created. You
 need to provide the disk name in which the directory will be
 located. If you do not specify a location, the directory will
 be placed in SYS$SYSDEVICE:[OMNI...]. Include the colon after the
 disk name.

* Enter the disk now [ddcu:] : <CR>

* Do you want to purge files replaced by this installation [YES]? <CR>

* Do you want to run the IVP after the installation [YES]? <CR>

        *******************************************************
        DEComni API has determined that the current Rdb
        version is 6.1 (MULTIVERSION)
        *******************************************************

*****************************************************************

  This installation will now continue in an unattended mode
  of operation, which will not require any user interaction.

  Estimated time to completion from this point is 5 minutes.
  This estimate is based on a configuration consisting of a
  DEC 3000 Model 500 and installation from a CD Rom.

  The current time is:11:00.43

*****************************************************************

%VMSINSTAL-I-RESTORE, Restoring product save set C ...
%VMSINSTAL-I-RESTORE, Restoring product save set B ...

%VMSINSTAL-I-SYSDIR, This product creates system directory
                                    [SYSHLP.EXAMPLES.OMNI].
%VMSINSTAL-I-SYSDIR, This product creates system disk directory
                                    $1$DKB400:[OMNI.ODS.CACHE].
%VMSINSTAL-I-SYSDIR, This product creates system disk directory
                                    $1$DKB400:[OMNI.ODS.EXAMPLES].
%VMSINSTAL-I-SYSDIR, This product creates system disk directory
                                    $1$DKB400:[OMNI.ODS.EXE].
%VMSINSTAL-I-SYSDIR, This product creates system disk directory
                                    $1$DKB400:[OMNI.ODS.HELP].
%VMSINSTAL-I-SYSDIR, This product creates system disk directory
                                    $1$DKB400:[OMNI.ODS.INCLUDE].
%VMSINSTAL-I-SYSDIR, This product creates system disk directory
                                    $1$DKB400:[OMNI.ODS.LIB].
%VMSINSTAL-I-SYSDIR, This product creates system disk directory
                                    $1$DKB400:[OMNI.ODS.LOCAL].
%VMSINSTAL-I-SYSDIR, This product creates system disk directory
                                    $1$DKB400:[OMNI.ODS.SCRIPTS].
%VMSINSTAL-I-SYSDIR, This product creates system disk directory
                                    $1$DKB400:[OMNI.INC30].
%VMSINSTAL-I-SYSDIR, This product creates system disk directory
                                    $1$DKB400:[OMNI.LIB30].
```

```
* If you would like to have DEComni for OpenVMS startup during the system
  startup,  add the following line to SYS$STARTUP:SYSTARTUP_VMS.COM.

      $ @SYS$STARTUP:OMNI_STARTUP.COM


Install DASTK Run Time

© DIGITAL EQUIPMENT CORPORATION. 1997. ALL RIGHTS RESERVED.

STARTUP PROCEDURE FOR: DEComni API for OpenVMS Alpha (TM) V3.1

Now checking process privileges...

Now checking for the DEComni license...

 %OMNI-I-LICENSE, The following valid license was found -  DECOMNI


Now checking that prerequisite software has been started...

Now installing the DEComni sharable images...

Now creating DEComni logicals...

Starting Omni Directory Services now...

Checking for required directories...

Checking for required files ...

ODS Identification information : 3.14.0

Checking cache area ODS:[CACHE]...

Checking Schema area ODS:[LOCAL]...

Required file ODS:[LOCAL]ods_known_attribute_types.dat not found
                                                 -- creating anew ...
Required file ODS:[LOCAL]ods_known_object_classes.dat not found
                                                 -- creating anew ...

Valid Schema definitions found in the attribute schema database as:
OC ALIASED_NAME CN C L S O OU P_ADDR APC OWNER APT AEQ MAPV

Valid Schema definitions found in the object class schema database as:
TOP ALIAS C L O OU AP AE DSA DEVICE

ODS_STARTUP: Normal completion.

Now checking ODS schema definitions...

Checking ODS misc schema definitions ...

 Added ODS Schema definition for Attribute Type SY (System Name)
 Added ODS Schema definition for Attribute Type OBJ_ID (Object Identifier)
 Added ODS Schema definition for Attribute Type BER (ASN.1 BER Encoding)
 Added ODS Schema definition for Attribute Type PSAP (Presentation Service Access Point)
 Added ODS Schema definition for Attribute Type SSAP (Session Service Access Point)
 Added ODS Schema definition for Attribute Type TSAP (Transport Service Access Point)
 Added ODS Schema definition for Attribute Type N_ADDR (Network Address)
 Added ODS Schema definition for Attribute Type AS_LIST (Supported Abstract Syntaxes)
 Added ODS Schema definition for Attribute Type DESCR (Description)
 Added ODS Schema definition for Attribute Type VMD (Associated VMD)
 Added ODS Schema definition for Attribute Type DOM (Associated Domain)
 Added ODS Schema definition for Attribute Type DEL (Deletable)
```

```
 Added ODS Schema definition for Object Class TAE (Titled Application Entity)
 Added ODS Schema definition for Object Class OAE (OMNI Application Entity)
 Added ODS Schema definition for Object Class AS (Abstract Syntax)

Checking ODS vmd schema definitions ...

 Added ODS Schema definition for Attribute Type VN (VMD Name)
 Added ODS Schema definition for Attribute Type APPSN (Application Simple Name)
 Added ODS Schema definition for Attribute Type APROF (Application Profile)
 Added ODS Schema definition for Attribute Type VERSION (MMS Version)
 Added ODS Schema definition for Attribute Type NLEVEL (Nesting Level)
 Added ODS Schema definition for Attribute Type MAX_CALLED (Max Services Called)
 Added ODS Schema definition for Attribute Type MAX_CALLING (Max Services Calling)
 Added ODS Schema definition for Attribute Type MAX_SEG (Max Segment Size)
 Added ODS Schema definition for Attribute Type CBB_LIST (CBB List)
 Added ODS Schema definition for Attribute Type SS_LIST (Services Supported List)
 Added ODS Schema definition for Attribute Type VENDOR (Vendor Name)
 Added ODS Schema definition for Attribute Type MODEL (Model)
 Added ODS Schema definition for Attribute Type REV (Revision)
 Added ODS Schema definition for Attribute Type INUM (Integrator Number)

 Added ODS Schema definition for Object Class VMD  (Virtual Manufacturing Device)

Checking ODS domain schema definitions ...

 Added ODS Schema definition for Attribute Type DN (Domain Name)
 Added ODS Schema definition for Attribute Type SHARE (Shareable)
 Added ODS Schema definition for Attribute Type CONT_FILE (Content File)
 Added ODS Schema definition for Attribute Type CAP_FILE (Capability File)

 Added ODS Schema definition for Object Class DOMAIN (Domain)

Checking ODS pi schema definitions ...

 Added ODS Schema definition for Attribute Type PN (PI Name)
 Added ODS Schema definition for Attribute Type REUSE (Reusable)
 Added ODS Schema definition for Attribute Type ARG (Execution Argument)
 Added ODS Schema definition for Attribute Type MON (Monitor Type)
 Added ODS Schema definition for Attribute Type DOM_LIST (Domain List)

 Added ODS Schema definition for Object Class PI (Program Invocation)

Checking ODS var schema definitions ...

 Added ODS Schema definition for Attribute Type VAR (Variable Name)
 Added ODS Schema definition for Attribute Type ATYPE (Application Type)
 Added ODS Schema definition for Attribute Type ADDR_T (Address Type)
 Added ODS Schema definition for Attribute Type ADDR_V (Address Value)

 Added ODS Schema definition for Object Class NVAR (Named Variable)
 Added ODS Schema definition for Object Class UNVAR (Unnamed Variable)

OMNI_SCHEMA.COM: Normal Completion.

 Added ODS Schema definition for Attribute Type PORTNAME (TT port)
 Added ODS Schema definition for Attribute Type STATIONID (Station address)
 Added ODS Schema definition for Attribute Type BAUDRATE (Line speed: baud rate)
 Added ODS Schema definition for Attribute Type CHARLEN (Character Lenght)
 Added ODS Schema definition for Attribute Type PRTYCHECK (Parity: NONE EVEN ODD)
 Added ODS Schema definition for Attribute Type STOPBIT (Stop bits)
 Added ODS Schema definition for Attribute Type FLOWCTRL (Flow Control: NONE HW SW)
 Added ODS Schema definition for Attribute Type LEV7CONN (Application Level 7 Conn)
 Added ODS Schema definition for Attribute Type PORTNUMBER (Port Number)
 Added ODS Schema definition for Attribute Type NETADDRESS (Network Address)
 Added ODS Schema definition for Attribute Type MODEM (Modem enabled)
 Added ODS Schema definition for Attribute Type RSPTIMEOUT (Response Timeout (sec))

 Added ODS Schema definition for Object Class DTK232 (DAS Toolkit for RS-232)
 Added ODS Schema definition for Object Class DTKTCP (DAS Toolkit for TCP/IP)
 Added ODS Schema definition for Object Class DTKOSI (DAS Toolkit for OSI)

%OMNI-S-STARTUP, DECommi startup successfully completed
```

```
    +------------------------------------------------+
    |                                                |
    |   The DEComni IVP will now be invoked....      |
    |                                                |
    +------------------------------------------------+

© DIGITAL EQUIPMENT CORPORATION. 1997. ALL RIGHTS RESERVED.
INSTALLATION VERIFICATION PROCEDURE FOR: DECOMNI for OpenVMS Alpha (TM) V3.1

Now checking process quotas...

Now checking process privileges...

Now checking that files have reached target directories...

%OMNI-S-IVP, THE DECOMNI IVP HAS SUCCESSFULLY COMPLETED
 Installation of OMNIAPIA V3.1 completed at 11:04
    Adding history entry in VMI$ROOT:[SYSUPD]VMSINSTAL.HISTORY
    Creating installation data file: VMI$ROOT:[SYSUPD]OMNIAPIA031.VMI_DATA
 VMSINSTAL procedure done at 11:04
```

B
Files Added During Installation

This appendix lists the directories and files that are added to a OpenVMS system by the DEComni API installation procedure.

```
ODS:[EXAMPLES]
--------------
ODS_EXAMPLE.C

ODS:[EXE]
---------
ODSCL.EXE
ODSVIEW.EXE
ODS_REHASH_DIRECTORY.EXE
ODS_CACHE_UTILITY.EXE
ODS_SCHEMA_UTILITY.EXE
ODS_TESTER.EXE

ODS:[HELP]
----------
ODSCL_ADD_EXPRESSION.TXT
ODSCL_ATTR_ABBREV.TXT
ODSCL_ATTR_EXPRESSION.TXT
ODSCL_ATTR_PROMPT.TXT
ODSCL_COMMANDS.TXT
ODSCL_COMMAND_ELEMENTS.TXT
ODSCL_COMMAND_SYNTAX.TXT
ODSCL_CONTINUATION.TXT

ODSCL_DEREGISTER.TXT
ODSCL_DEREGISTER_EXAMPLES.TXT
ODSCL_DEREGISTER_SYNTAX.TXT
ODSCL_DIRECTORY_PROMPT.TXT
ODSCL_DISPLAYS.TXT
ODSCL_EXIT.TXT
ODSCL_FILE_PROMPT.TXT
ODSCL_HELP.IDX
ODSCL_HELP.TXT

ODSCL_INVOKING.TXT
ODSCL_LIST.TXT
ODSCL_LIST_EXAMPLES.TXT
ODSCL_LIST_SYNTAX.TXT
ODSCL_LOCAL_DIRECTORY.TXT
ODSCL_MATCH_EXPRESSION.TXT
ODSCL_MATCH_PROMPT.TXT

ODSCL_MODIFY.TXT
ODSCL_MODIFY_EXAMPLES.TXT
ODSCL_MODIFY_PROMPT.TXT
ODSCL_MODIFY_SYNTAX.TXT
ODSCL_NAME_EXPRESSION.TXT
ODSCL_NAME_PROMPT.TXT
ODSCL_OC_ABBREV.TXT
ODSCL_OPTION_ABBREV.TXT
```

```
ODSCL_PATH_EXPRESSION.TXT
ODSCL_PATH_PROMPT.TXT
ODSCL_PROMPTS.TXT
ODSCL_READ.TXT
ODSCL_READ_EXAMPLES.TXT
ODSCL_READ_SYNTAX.TXT

ODSCL_REGISTER.TXT
ODSCL_REGISTER_EXAMPLES.TXT
ODSCL_REGISTER_SYNTAX.TXT
ODSCL_REMOVE_EXPRESSION.TXT
ODSCL_REPLACE_EXPRESSION.TXT
ODSCL_SET.TXT
ODSCL_SET_EXAMPLES.TXT
ODSCL_SET_SYNTAX.TXT

ODSCL_SHOW.TXT
ODSCL_SHOW_EXAMPLES.TXT
ODSCL_SHOW_SYNTAX.TXT
ODSCL_TO_FILE_EXPRESSION.TXT
ODSCL_TO_PROMPT.TXT
ODSCL_WITH_EXPRESSION.TXT
ODSCL_WITH_PROMPT.TXT
ODSVIEW_HELP.TXT

ODS:[INCLUDE]
-------------
ODS_DEFS.H
ODS_MSG.H

ODS:[LIB]
---------
LIBODS.OLB

ODS:[LOCAL]
-----------
ODSCL_PARSE_TBL.BIN
ODSVIEW.UID
ODS_COPY_DIALOG.UID
ODS_ENTRY_DIALOG.UID
ODS_MODIFY_DIALOG.UID
ODS_SELBOX_DIALOG.UID
ODS_VERSION.DAT

ODS:[SCRIPTS]
-------------
DASTK_SCHEMA.COM
DOMAIN_SCHEMA.COM
MISC_SCHEMA.COM
ODS_IVP.COM
ODS_STARTUP.COM
OMNI_SCHEMA.COM
PI_SCHEMA.COM
VAR_SCHEMA.COM
VMD_SCHEMA.COM

SYS$COMMON:[SYSEXE]
-------------------
OMNIVIEW.EXE
OMNIVIEW.UID
OMNI_ODF.EXE
OMNI_REGISTRY.EXE
```

```
SYS$COMMON:[SYSHLP]
-------------------
OMNI031.RELEASE_NOTES
OMNIVIEW_HELP.TXT
OMNI_ODF_HELP.HLB

SYS$COMMON:[SYSHLP.EXAMPLES.OMNI]
---------------------------------
OMNI_CONFIGURE.NCL
OMNI_CREATE_DEFINITION.C
OMNI_CREATE_DEFINITION.FOR
OMNI_FORTRAN.TXT
OMNI_ODF_EXAMPLE.COM
OMNI_RUN_TIME_PI_DOM.C
OMNI_RUN_TIME_VAR.C
OMNI_RUN_TIME_VMD.C
OMNI_VAR_READ_STRUCTURE.FOR
OMNI_VAR_READ_WRITE.FOR

SYS$COMMON:[SYSLIB]
-------------------
OMNI_API_PROTOS.H
OMNI_CODES.ADA
OMNI_CODES.BAS
OMNI_CODES.FOR
OMNI_CODES.H
OMNI_CODES.MAR

OMNI_CODES.PAS
OMNI_CODES.PLI
OMNI_CODES.R32
OMNI_CODES.REQ
OMNI_DATABASE.RDB
OMNI_DATABASE.SNP

OMNI_DEFS.ADA
OMNI_DEFS.BAS
OMNI_DEFS.FOR
OMNI_DEFS.H
OMNI_DEFS.MAR
OMNI_DEFS.PAS
OMNI_DEFS.PLI
OMNI_DEFS.R32
OMNI_DEFS.REQ

OMNI_INTEGRATORS_DEFS.H
OMNI_INTEGRATORS_DEFS.PAS
OMNI_INTEGRATORS_DEFS.BAS
OMNI_INTEGRATORS_DEFS.OFR
OMNI_INTEGRATORS_DEFS.MAR
OMNI_INTEGRATORS_DEFS.ADA

OMNI_AST_BASIC_SHR.EXE
OMNI_BASIC_SHR.EXE
ODS_SHR.EXE

SYS$COMMON:[SYSMGR]
-------------------
OMNI_CONFIG.COM
OMNI_STOP.COM

SYS$COMMON:[SYSMSG]
-------------------
OMNI_CODES_MSG.EXE
```

```
SYS$COMMON:[SYS$STARTUP]
------------------------
OMNI_LOGICALS.COM
OMNI_STARTUP.COM

SYS$COMMON:[SYSTEST]
--------------------
OMNI_IVP.COM

<ODS_DISK>:[OMNI.INC31]
-----------------------
OMNI_BASIC_TYPES.H
OMNI_CONFIG.H
OMNI_DEBUG.H
OMNI_EXCEPTIONS.H
OMNI_IDENT.H
OMNI_INTEGRATOR_PROTOS.H
OMNI_INT_PREPOST.H
OMNI_IOI_DEF.H
OMNI_QUEUE.H
OMNI_SHARED_TYPES.H
OMNI_SYNC.H
```

Index