COMPAQ

Software Product Description

PRODUCT NAME: Compaq TDMS for OpenVMS VAX, Version 1.9B (Terminal Data Management System) **SPD 25.71.18**

DESCRIPTION

Compaq TDMS for OpenVMS VAX (TDMS) is a product designed for the implementation of interactive, forms-intensive applications running on OpenVMS VAX systems. As a terminal subsystem, TDMS can reduce the application development and maintenance effort by replacing application program logic specific to terminal interactions with definitions that are external to the program.

TDMS has been superseded by Compaq DECforms for OpenVMS. DECforms is the preferred forms product for character-cell terminal applications and should be used for all new application development requiring a forms-based user interface.

Features

TDMS provides the following features:

- A Screen Editor to define forms that format data on the terminal.
- A nonprocedural language to define the exchange of data between an application program and its associated terminal. These predefined exchanges are called requests and are external to the program.
- Utilities that enable the creation, modification, and storage of form definitions and request definitions.
- A record-level programming interface that the application program uses to invoke the predefined requests. The application program calls the program interface, passing it the request name and the program record buffers used in the exchange of data with the terminal.
- A sliding window algorithm through which the behavior of TDMS can be modified to control how 2-digit year dates are interpreted upon input.

TDMS applications range from database inquiry, response, and update to real-time uses such as the periodic display of an industrial process. TDMS is typically used as a terminal subsystem in terminal data management applications such as order entry, inventory control, distribution, and other form-intensive applications. TDMS not only increases application development and maintenance productivity by providing a separation of terminal management code from application code, but it also promotes end user productivity by providing the following features:

- Vertical Field Traversal. Keys can be defined for rapidly traversing forms vertically rather than traversing the form on a field by field basis in the predefined field visitation order.
- Immediate access to the first field and last field. Keys can be defined for immediately moving the cursor to the first field and the last field on the form.
- The key function associated with a key can be changed and additional keys can be defined to have other functions. This allows keyboards to be redefined to match the environment of the end user.
- All TDMS definitions are stored in the Oracle Common Data Dictionary/Repository (CDD/Repository). Additionally, TDMS can use record definitions stored in the Oracle CDD/Repository by Compaq DATATRIEVE, Oracle CODASYL DBMS, or Oracle Rdb. TDMS provides full support for Oracle CDD/Repository Dictionary Management Utility (DMU) format definitions and read-only support for Common Dictionary Operator (CDO) definitions.

- Compaq DATATRIEVE for OpenVMS can use TDMS forms to DISPLAY/READ data.
- TDMS supports editing with the Language-Sensitive Editor from within the Request Definition Utility (RDU). This editor provides templates that help application developers remember RDU syntax as they develop request definitions.
- TDMS forms can be printed. Video attributes are not printed and lines are drawn using the characters "-", "+", "|" rather than a line-drawing character set.

Implementing TDMS Applications

1. Define the Request

A request defines what information is displayed at the terminal and what information is collected from the terminal. The TDMS request replaces code that would otherwise have to be designed, written, and debugged in the application program.

The request identifies the form and the record definitions to be used in the exchange of data with the terminal. INPUT and OUTPUT statements define the mapping between the form fields and the record fields during execution of the request.

2. Define the Form

A form definition describes the format of the data that is displayed on the terminal at run time. The form definition also specifies, for each field, what validation procedures are invoked. TDMS field validation includes range checks, list checks, and picture validations.

3. Define the Record

Data records that are used in a request and TDMS application program must be defined in the Oracle CDD/Repository. The record definition specifies the type, structure, and length of the record that is created by Oracle CDD/Repository, Compaq DATATRIEVE, Oracle CODASYL DBMS, or Oracle Rdb. During the execution of a request, TDMS performs the necessary data type conversion required to transfer data between the form and the record.

4. Implement the Application Program

The application program performs application-specific processing and controls the flow of information between the terminal and the database. The application communicates at a record level with the terminal by calling TDMS to execute requests and it communicates with the database using the subsystem of choice, including RMS, Compaq DATATRIEVE, Oracle CODASYL DBMS, Oracle Rdb, or any other database supported on OpenVMS VAX.

Applications can be written in any VAX native mode language that adheres to the VAX Procedure Calling and Condition Handling standard. Many languages are able to copy record definitions from the Oracle CDD/Repository at compile time. If the application program is written in one of these languages, then it and the TDMS request can share record definitions in the Oracle CDD/Repository. TDMS applications written in languages that do not support such copying must include the record definitions in the application program itself as well as in the Oracle CDD/Repository.

Components

The Form Definition Utility

The TDMS Form Definition Utility (FDU) provides all of the capabilities needed to create or modify form definitions and store them in the Oracle CDD/Repository. The TDMS FDU includes a Screen Editor that is used to create a screen image of the form and assign specific attributes. These include form field video attributes, form field validation procedures, and the order in which the input fields should be processed.

The Request Definition Utility

The TDMS Request Definition Utility (RDU) provides all of the capabilities needed to create and modify requests and store them in the Oracle CDD/Repository. RDU validates each request to make sure that form and record definitions exist and that all transfers of data between form and record fields are valid. The RDU also builds Request Library Files that the TDMS run-time system accesses during the execution of requests.

The request library capability is provided to avoid run-time access to the Oracle CDD/Repository and thus improve TDMS application performance.

Programming Call Interface

An application program uses the TDMS programming interface to execute a TDMS request. A request defines an exchange of data between the program and its associated terminal. The application program calls the program interface and passes to the program interface the request name and the program record buffers to be used in the exchange of data with the terminal. This record level interface thus eliminates the need for character level or field level communication with the terminal.

The Programming Call Interface allows application programs to perform additional operations including writing text to or reading text from the reserved message line on a terminal, enabling or disabling a facility that traces the action of a request, and canceling a request in progress.

The Programming Call Interface supports both synchronous and asynchronous calls from the application program.

Optional Run-Time System

An optional run-time version of TDMS is available. The run-time system allows for the execution of TDMS-based applications on a system other than the one used to develop the application.

Documentation

The TDMS documentation set is available in hardcopy only.

The documentation set consists of:

- *Forms Manual*
- *Request and Programming Manual*
- *Reference Manual*
- *Installation Guide*
- *Pocket Guide*

HARDWARE REQUIREMENTS

Processors Supported

TDMS is supported on all VAX hardware configurations referenced in the OpenVMS Operating System for Alpha and VAX Software Product Description (SPD 25.01.xx).

Processors Not Supported

The following processors are not supported:

- MicroVAX I
- VAXstation I
- VAXstation 8000
- VAX-11/725
- VAX-11/782

Terminals Supported

Terminal	Notes
VT52	1
VT100,VT100 W/AVO,VT101,VT102,VT125 VT131 VT220,VT240,VT241	2
VT320	3
VT330,VT340	3,4
VT420	5
VT510	5
VT520,VT525	5

1. Supported for application execution only. Run-time support for the VT52 is generally the same as for the VT100. When a VT100 feature is requested in a form definition, the feature will either be simulated or ignored when the form is displayed on a VT52. Wide forms (132 columns) and forms using scrolled areas cannot be displayed on a VT52.
2. Supported in VT102 mode only.
3. Usage of the status line, line 25, is not supported.
4. Usage of the locator device (mouse) is not supported.
5. TDMS form definitions are either 80 or 132 columns by 24 lines or less. These terminals contain features allowing the display of more than 24 lines including a status line. TDMS does not support these features.

Terminal Emulators

Terminal emulators on workstations and personal computers are supported only to the extent that the emulator conforms to the supported terminal environment being emulated.

Disk Space Requirements (Block Cluster Size = 1)

These counts refer to the disk space required on the system disk. The sizes are approximate; actual sizes may vary depending on the user's system environment, configuration, and software options.

Kit	Disk Space
Development Installation	12,000 blocks 6.1 Mb
Development Permanent with Samples	5,000 blocks 2.6 Mb
Development Permanent without Samples	2,500 blocks 1.3 Mb
Run-Time Installation	3,000 blocks 1.5 Mb
Run-Time Permanent	500 blocks 0.3 Mb

SOFTWARE REQUIREMENTS

Software	Versions
OpenVMS VAX Operating System	V5.5-2, V6.2, V7.2
Oracle CDD/Repository	V7.0.1

OpenVMS Tailoring Classes

For VMS V5.x systems, the following VMS classes are required for full functionality of this layered product:

- VMS Required Save Set
- Network Support
- Programming Support
- Utilities

OPTIONAL SOFTWARE

Certain versions of these products depend on a specific version of the Operating System. Please refer to the SPD of the product in question to determine which version you need.

- Compaq ACMS for OpenVMS, V4.3
- Compaq DATATRIEVE for OpenVMS, V7.2A
- Language-Sensitive Editor/Source Code Analyzer (LSE/SCA), V4.7 (part of Compaq DECset for OpenVMS VAX Systems)

SOFTWARE LICENSING

This software is furnished only under a license. For more information about Compaq's licensing terms and policies, contact your local Compaq office.

LICENSE MANAGEMENT FACILITY (LMF)

This layered product supports the OpenVMS License Management Facility.

License units for both the Development System and the Run-Time System are allocated on a capacity basis by hardware tier.

For more information on the License Management Facility, refer to the OpenVMS Operating System Software Product Description (SPD 25.01.xx) or the License Management Facility manual of the OpenVMS Operating System documentation set.

CLUSTER ENVIRONMENT

This layered product is fully supported when installed on any valid and licensed VMScluster configuration without restrictions. The *Hardware Requirements* section of this Software Product Description details any special hardware required by this product.

VMScluster configurations are fully described in the VMScluster Software Product Description (SPD 29.78.xx) and include CI, Ethernet, and Mixed Interconnect configurations.

DISTRIBUTION MEDIA

This product is distributed on CD-ROM as part of the OpenVMS VAX Software Product Library.

ORDERING INFORMATION

Software Licenses

Development and Run-time: QL-706A*-**

Run-time only: QL-711A*-**

Documentation

Hardcopy: QA-706AA-GZ

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

Media for this product is available as part of the OpenVMS VAX Software Product Library. The documentation is available only in hardcopy and must be ordered separately.

SOFTWARE PRODUCT SERVICES

The Mature Product Support with Sustaining Engineering Service is offered for this product and includes telephone support, tested software patches for any new problems, and problem escalation to engineering-level resources if appropriate. This support offering is equivalent to current version support. Compaq will not be adding new features or functionality to this product and no new releases are planned. Accordingly, License Subscription and Software Update Distribution Services are no longer available. Please contact your local Compaq Services Sales Specialist for more information about the Mature Product Support with Sustaining Engineering Service.

GROWTH CONSIDERATIONS

The minimum hardware and software requirements for any future version of this product may be different from the requirements for the current version.

SOFTWARE WARRANTY

This software is provided by Compaq with a 90-day conformance warranty in accordance with the Compaq warranty terms applicable to the license purchase.

The above information is valid at time of release. Please contact your local Compaq office for the most up-to-date information.

Compaq shall not be liable for technical or editorial errors or omissions contained herein. The information is provided "as is" without warranty of any kind and is subject to change without notice. The warranties for Compaq products are set forth in the express limited warranty statements accompanying such products. Nothing herein should be construed as constituting an additional warranty.