COMPAQ

Software Product Description

PRODUCT NAME: Compaq FMS for OpenVMS, Version 2.4 (Forms Management System) **SPD 45.06.01**

DESCRIPTION

Compaq FMS for OpenVMS (FMS) is a mature, character-cell based forms management system for interactive applications running on OpenVMS systems that use video forms as the user interface. FMS provides application programmers with a set of development tools to create and maintain forms-based user interfaces and a run-time system for displaying and managing the application's user interface at execution time.

FMS has been superseded by Compaq DECforms for OpenVMS.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

Forms defined using FMS provide the programmer with the ability to use the following features of VT terminals by Compaq:

- Individual character attributes of reverse video, bold, blinking, and underline
- Line attributes of double width, double height, and scrolled
- Screen width attributes such as 80 or 132 column lines, and reverse video
- Alternate character sets including the VT100 "special graphics character set" for line drawing

FMS form data structures are used by the Form Driver (FDV) during run time to display forms, and to access and validate data entered by the terminal operator.

Forms can be created and modified interactively with the interactive editing facility (FMS/EDIT), or they can be defined as a source form description with the Form Language and then converted to a form data structure using the Form Language Translator (FMS/TRANSLATE). In general, forms larger than 54 lines in 80-column mode and 63 lines in 132-column mode can only be created and modified with the Form Language Translator.

Forms usually reside in form library files on disk and are retrieved as needed by application programs at execution time. This arrangement results in a high degree of independence between form data structures and application programs. Forms can be modified without recompilation or relinking of the application program. Forms can also be converted into an object module and linked with application programs to create memory-resident forms, or forms can be dynamically loaded into memory by the application program.

FMS applications use the Form Driver to control the display of forms, their associated data fields, and the input and output of data into fields. All data input from the terminal and output to the terminal is passed as ASCII strings.

Data for each field in a FMS form is validated by the Form Driver. Field-validation characters are used to construct a "field validation picture." During program execution, characters input or output to the field are compared to the picture to determine if they are valid. In most instances, the field picture can contain several different field validation characters and also field-marker characters. Field-marker characters, used to improve the readability of the displayed form, are transparent to the application program.

Attributes from the following list can be assigned to each individual field:

- Name of the Field
- A default value
- A help message
- Autotab
- Clear Character
- Display Only
- Fixed Decimal
- Indexed
- Left Justify
- Right Justify
- Must Fill
- No Echo
- Response Required
- Supervisor Only
- Uppercase
- User action routines and associated data parameters
- Zero Fill
- Zero Suppress

The attributes assigned to each field are checked for validity when the form is created. Some combinations of attributes are not legal. Each form has attributes that apply to the form as a whole. These attributes include:

- Name of the form
- Name of an associated help form
- Screen background (black, white, or do not change)
- Screen width (80 columns, 132 columns, or do not change)
- Screen area to clear when displaying the form
- Terminal character set
- User action routines and associated data parameters
- Video attributes to be used if the field being worked on is to be highlighted at run time
- The order in which fields are to be accessed
- Named Data

Application Development Utilities

Interactive Form Editor

The FMS/EDIT command invokes the interactive editing facility that is used by the form developer to create and modify video forms.

The creation of forms with the interactive editing facility is accomplished in phases. The layout phase is used to specify the visual appearance of the form, the background text, and what types of characters are to be valid input to the fields. While laying out a form on the screen, one has available a set of 48 function keys and key combinations. These keys perform functions such as cursor movement, delete and undelete, cut and paste, box drawing, video attribute assignment, centering, and displaying Help. Additional phases are used to assign form and field attributes, to enter Named Data, and to assign field access order. Named Data and all form and field attributes, including user action routines, are specified by filling in questionnaires. Highlighting is used to indicate which field is being referred to during the field attribute and the field order assignment phases. Finally, the test phase allows the developer to interact with a form just as the end user would without exiting the editing session.

Form Language Translator

The Form Language Translator provides an alternate means for defining forms. The Form Language is a free-form, keyword-oriented, declarative language that provides the means to create and modify source form descriptions with a text editor or program.

The FMS/TRANSLATE command is used to convert the source form descriptions (contained in a text file) to form data structures. The data structures created are equivalent to those created with the FMS interactive editor. Existing form data structures can be converted back into Form Language source code for modification or documentation by using the FMS/DESCRIPTION/FULL command.

Other Application Development Aids

FMS provides additional facilities to aid in the application program development process. The facilities are invoked with the following commands:

- FMS/LIBRARY - To create and maintain form libraries.
- FMS/DESCRIPTION - To obtain any of the following basic types of descriptions of forms:
  - — /BRIEF - Summary information about a form, its fields, Named Data, and user action routines.
  - — /FULL - Text files containing Form Language statements suitable for input to the Form Language Translator.
  - — /DECLARATIONS - To obtain data division code that can be edited by the user to create a COBOL Data Division or DATATRIEVE Domain Definition reflecting the names, lengths, and order of fields on the screen.

  - — /DISPLAY_IMAGE - A printable image of a form in either terminal image format (with escape sequences) or line printer format (without escape sequences).
- FMS/DIRECTORY - To obtain a directory list of forms in a library or in form files.
- FMS/OBJECT - To generate an object module containing one or more form data structures that, when linked with the application program, will allow access to the forms through virtual memory.
- FMS/VECTOR - To generate an object module containing a vector table of addresses that point to the user action routines that are linked with the application program.
- FMS/TEST - To invoke the form testing facility, which allows the application developer to display a form as an application program would, to type data into fields, and to display field Help.

Form Driver

The Form Driver is the run-time component of FMS. It consists of AST reentrant, shareable subroutines that are called by application programs to control the interaction of the terminal, the user, and the application program. The Form Driver provides facilities for managing multiple work areas (each of which contains a single form data structure). Multiple forms can be displayed on a single terminal screen.

Form Driver subroutines provide for terminal input and output, display of forms, manipulation of the screen, basic input validation and formatting, calling user action routines, and responding to the terminal operator's requests for help. Input and output can be specified one field at a time or on a whole form basis.

The Form Driver has a number of features to facilitate fine tuning of the application human interface. A timeout facility allows the application program to continue execution after waiting a specified period of time for the operator to enter the next character. Field video attributes, cursor position, and Insert/Overstrike mode can be dynamically altered by Form Driver calls in the application program. Field highlighting, controlled by the Form Driver, causes the video attributes of each field to be altered as the cursor enters the field and to be restored to their original state when the operator exits the field.

The Form Driver may be purchased separately for systems that require only run-time support for applications using FMS.

Named Data

The Form Driver supports the creation of parameter-driven applications by providing Named Data.

Named Data are form constants that can be stored as part of the form data structure and retrieved at execution time by calls in the application program. Named Data allows information needed by the application program (such as form linkage, operator messages and other human language information, and data validation criteria for user subroutines) to be defined, stored, and modified independently of the application program.

User Action Routines

User action routines (UARs) are procedures written by the user in any supported OpenVMS programming language. UARs are associated with forms and fields during the creation of form data structures and are automatically called by the Form Driver under the following conditions:

- When processing for a field is finished
- Before and after processing the terminal operator's Help request
- When the terminal operator presses a function key
- When a screen refresh operation is requested

The Form Driver transfers program control to the user action routine under the circumstances defined. The user action routine has available to it (through Form Driver calls) a parameter string of 80 characters (defined as part of the form) which can be used to give processing information and the current Form Driver context.

Sample Application Program

The Sample Application Program (SAMP) is a supported FMS demonstration program, included in the distribution kit, and is used to certify proper installation of FMS. The documentation for FMS gives examples from the Sample Application Program. The sample programs (in each of the documented programming languages) and form descriptions are included in the documentation and in the distribution kit. The Sample Application Program can be installed on the target system at the user's discretion.

Printing Forms

Form images can be created and written to a file for subsequent printing by either an application program using a Form Driver call, FDV$PRINT_SCREEN, or by a terminal operator using an FMS DCL command, FMS/DESCRIPTION/DISPLAY_IMAGE. The following screen images can be created:

Line Printer Image - Line drawing graphic characters are converted to standard printable ASCII characters and all other characters from alternate character sets are ignored.

Terminal Image - All escape sequences and control characters are included to present an exact image of the screen as displayed. This image can be output properly only to a printer that understands or can interpret the control characters and escape sequences used by the current terminal.

Documentation

The FMS documentation set is bundled with the development kit and is also available separately. It is not part of the FMS run-time only kit.

The documentation set consists of:

- *Introduction to FMS*
- *Form Driver Reference Manual*
- *Utilities Reference Manual*
- *Language Interface Manual*
- *Mini-Reference*
- *Installation Guide*

HARDWARE REQUIREMENTS

Processors Supported

FMS is supported on all hardware configurations referenced in the OpenVMS Operating System for Alpha and VAX, Version 7.2, 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

Form Development

Using the Interactive Form Editor, a VT100, VT200, VT300, VT420, or VT500-series terminal is required for the creation of forms of up to 23 lines. A supported OpenVMS workstation is required for the creation of forms of up to 54 lines in 80-column mode or 63 lines in 132-column mode. Using the Form Language, forms of up to 254 lines can be created on any terminal.

Application Execution

FMS based application execution or usage of the form test facility using forms of up to 23 lines in length is fully supported on VT52, VT100, VT200, VT300, VT420, and VT500-series terminals. Forms up to 54 lines in 80-column mode and 63 lines in 132-column mode can only be displayed on supported OpenVMS workstations. With the exception of VT420 and VT500-series terminals that contain features to permit the display of forms up to 72 lines, forms greater than 63 lines cannot be displayed on any terminal, but may be printed.

Terminal	Notes
VT52	1
VT100,VT100 w/AVO	
VT101,VT102,VT103,VT125	
VT131	2
VT180	
VT220	
VT240	3
VT241	4
VT320	5
VT330	5,6
VT340	4,5,6
VT420	7
VT510	7
VT520	7
VT525	4,7

1. Supported for application execution only. The Interactive Form Editor cannot be used on this terminal.
2. The VT131 terminal is supported in VT102 mode only.
3. The usage of blinking, double-wide, and/or double-sized lines in forms can result in substantially reduced performance.
4. Usage of color is not supported.
5. Usage of the status line, line 25, is not supported.
6. Usage of the locator device (mouse) is not supported.
7. Forms of up to 72 lines may be defined for VT420 and VT500 series terminals. These terminals can display 24, 36, or 48 lines on a single screen. Forms larger than the screen size can be panned under local hardware control. FMS does not support access to the F1-F5 function keys or to the ALTERNATE key to activate another terminal session. The COMPOSE key is supported when configured for LOCAL COMPOSE.

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)

FMS can be installed with or without the Sample Application Program (SAMP).

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	VAX	Alpha
Development Installation with SAMP	3,000 blocks 1,536 Kb	4,000 blocks 2,000 Kb
Development Installation without SAMP	1,850 blocks 948 Kb	2,500 blocks 1,250 Kb
Development Permanent with SAMP	2,650 blocks 1,357 Kb	3,500 blocks 1,800 Kb
Development Permanent without SAMP	1,300 blocks 666 Kb	1,800 blocks 900 Kb
Run-Time Installation	350 blocks 180 Kb	500 blocks 250 Kb
Run-Time Permanent	300 blocks 154 Kb	400 blocks 200 Kb

SOFTWARE REQUIREMENTS

Software	OpenVMS VAX	OpenVMS Alpha
Operating System	V5.5-2, V6.2, V7.1, or higher	V6.2, V7.1, or higher

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
- Programming Support
- Utilities

OPTIONAL SOFTWARE

Compaq DATATRIEVE for OpenVMS, V6.1 - V7.2

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.

YEAR 2000 READY

This product is Year 2000 Ready and is capable of accurately processing, providing, and/or receiving date data from, into, and between the twentieth and the twenty-first centuries, and the years 1999 and 2000, including leap year calculations, when used in accordance with the associated product documentation and provided that all hardware, firmware, and software used in combination with the product properly exchange accurate date data with the product.

DISTRIBUTION MEDIA

This product is distributed on:

- CD–ROM (as an individual product for OpenVMS Alpha)
- CD–ROM (as part of the OpenVMS Software Product Libraries)

The software documentation is also available as part of the OpenVMS Online Documentation Libraries on CD–ROM.

ORDERING INFORMATION

OpenVMS VAX	Part Number
Development License, Unlimited Use	QL-VD7A*-AA
Development Update License	QL-VD7A*-RA
Development Software Product Services	QT-VD7A*-**
Run-Time License, Unlimited Use	QL-VD8A*-AA
Run-Time Update License	QL-VD8A*-RA
Run-Time Software Product Services	QT-VD8A*-**
Media and Documentation (OpenVMS VAX Software Product Library)	QA-5G88A-H8
Hardcopy Documentation	QA-VD7AA-GZ

OpenVMS Alpha	Part Number
Development License, Unlimited Use	QL-MVSA*-AA
Development Update License	QL-MVSA*-RA
Development Media and Documentation	QA-MVSAA-H8
Development Software Product Services	QT-MVSA*-**
Run-Time License, Unlimited Use	QL-MVTA*-AA
Run-Time Update License	QL-MVTA*-RA
Run-Time Media	QA-MVTAA-H8
Run-Time Software Product Services	QT-MVTA*-**
Media and Documentation (OpenVMS Alpha Software Product Library)	QA-03XAA-H8
Hardcopy Documentation	QA-MVSAA-GZ

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

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 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.

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

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.

© 2000 Compaq Computer Corporation

Compaq, the Compaq logo, and the DIGITAL logo are registered in the U.S. Patent and Trademark Office.

All other product names mentioned herein may be trademarks or registered trademarks of their respective companies.

Confidential computer software. Valid license from Compaq Computer Corporation or authorized sublicensor required for possession, use or copying. Consistent with FAR 12.211 and 12.212, Commercial Computer Software, Computer Software Documentation, and Technical Data for Commercial Items are licensed to the U.S. Government under vendor's standard commercial license.

Compaq Computer Corporation shall not be liable for technical or editorial errors or omissions contained herein. The information in this document is subject to change without notice.

CI, DATATRIEVE, DECforms, FMS, MicroVAX, OpenVMS, VAX, VAXstation, VMS, VMScluster, VT, VT52, VT100, VT220, VT300, VT320, VT330, VT340, and VT420 are trademarks of Compaq Computer Corporation.