MenuFinder

Version 3.2

For the OpenVMS/VAX operating system

Freeware version

Administrator's Guide

i3 - Italian Internet Information

NOTE: MenuFinder is packaged for distribution using Info-ZIP's compression utility. The installation program uses UnZip to read zip files from the distribution kit CD. Infor-ZIP's software, (Zip, UnZip and related utilities) is free and can be obtained as source code or executables from various bulletin board services and anonymous – fpt sites, including CompuServe's IBMPRO forum and ftp.uu.net:/pub/archiving/zip/*

First edition: September 1991

1. MENUFINDER INSTALLATION

1.1. Requirements for the Installation

The installation of MenuFinder for OpenVMS requires:

- OpenVMS/VAX version 6.2 or later
- About 5000 free disk blocks

1.2. How to test MenuFinder for OpenVMS/AXP

MenuFinder is free only for OpenVMS/VAX.

To test MenuFinder for OpenVMS/AXP visit the site http://www.itre.com/mf/download_axp.html and read the instructions to download the AXP package.

1.3. Copying the Installation Kit

In a cluster environment this phase is executed only once.

Login to the account SYSTEM and execute the following steps,

1.3.1. Creation of an Installation Directory

Create the MenuFinder directory .

```
$ CREATE/DIR name_directory /PROT=W:RE
```

For example,

```
$ CREATE/DIR SYS$SYSDEVICE:[MF32]/PROT=W:RE
```

N.B. If you are installing MenuFinder in a cluster environment, the chosen disk must be cluster wide mounted.

1.3.2. Copying the kit to the installation directory

Download the files MF32ENG.ZIP (Installation kit for VAX) and UNZIP.EXE (Unzip for VAX) saving them into the MenuFinder directory specified in paragraph 1.3.1 .

The download can be executed using an internet browser on a PC or a MAC ; the kit can be transfered into the MenuFinder directory afterwards via FTP (use BIN transfer mode) or DECnet.

If you have the files MF32ENG.ZIP and UNZIP.EXE already available in the Compaq Freeware CD or in a local OpenVMS directory (for example, after the decompression of the package AdvancedMenuFinder.ZIP downloaded from a freeware internet site) copy them into the MenuFinder directory specified in paragraph 1.3.1 .

1.3.3. Decompressing the Installation Kit

Decompress the Kit using the following commands :

$ DEFINE MENU$DIRECTORY name_directory (see paragraph 1.3.1).

$ SET DEF MENU$DIRECTORY

$ UNZIP == "$MENU$DIRECTORY:UNZIP.EXE"

$ UNZIP –ad MF32ENG.ZIP

1.4. Installation / Activation of MenuFinder on the OpenVMS node

This section must be repeated on each node on which you wish to install MenuFinder. The procedure recognises if previous versions of MenuFinder were installed and asks if you wish to preserve the existing configuration to guarantee that MenuFinder continues to function with the current Menu format. LOGIN on the node as SYSTEM user and define the logical name MENU$DIRECTORY with the command:

$ *DEFINE MENU$DIRECTORY name_directory (see paragraph 1.3.1)*

Install MenuFinder with the procedure:

```
$ @MENU$DIRECTORY:INSTALL
```

The install procedure does not activate immediately this new version of MenuFinder for all user's. The SYSTEM user can perform an operating check on the new version of the software while other users continue to use the preceding version (see 1.5).

1.4.1. Example of First Installation on a cluster node

```
$ DEFINE  MENU$DIRECTORY  SYS$SYSDEVICE:[MF32]

$ @MENU$DIRECTORY:INSTALL
```

MenuFinder v. 3.2 installation

Creating SYS$LOGIN:LOGIN.DMF ...
Creating SYS$SPECIFIC:[SYSMGR]SYLOGIN.DMF ...
Creating SYS$COMMON:[SYSMGR]CLULOGIN.DMF ...

Installation of sample menus.

In cluster installation the disk MUST be cluster wide mounted

FULL name of sample directory [SYS$SYSDEVICE:[MFSAMPLE]]:

Copying sample menus to SYS$SYSDEVICE:[MFSAMPLE] ...
See the Administrator Guide to complete the MenuFinder installation.

1.4.2. Example of Installation on Another Cluster Node

```
$ DEFINE MENU$DIRECTORY SYS$SYSDEVICE:[MF32]

$ @MENU$DIRECTORY:INSTALL
```

MenuFinder v.3.2 installation

Creating SYS$LOGIN:LOGIN.DMF ...
Creating SYS$SPECIFIC:[SYSMGR]SYLOGIN.DMF ...
See the Administrator Guide to complete the MenuFinder installation.

1.4.3. Example of a first upgrade of MenuFinder on a cluster node

```
$ DEFINE MENU$DIRECTORY SYS$SYSDEVICE:[MF32]

$ @MENU$DIRECTORY:INSTALL
```

MenuFinder v. 3.2 installation

SYS$LOGIN:LOGIN.DMF already exists.

Do you want to keep it ? [Y/n]:

SYS$SPECIFIC:[SYSMGR]SYLOGIN.DMF already exists.

Do you want to keep it ? [Y/n]:

SYS$COMMON:[SYSMGR]CLULOGIN.DMF already exists.

Do you want to keep it ? [Y/n]:

Installation of sample menus.

In cluster installation the disk MUST be cluster wide mounted

FULL name of samples directory [SYS$SYSDEVICE:[MFSAMPLE]]:

Copying sample menus to SYS$SYSDEVICE:[MFSAMPLE] ...

See the Administrator Guide to complete the MenuFinder installation.

1.4.4. Example of an upgrade of MenuFinder on Another Cluster Node

```
$ DEFINE MENU$DIRECTORY SYS$SYSDEVICE:[MF32]

$ @MENU$DIRECTORY:INSTALL
```

MenuFinder v. 3.2 installation

SYS$LOGIN:LOGIN.DMF already exists.

Do you want to keep it ? [Y/n]:
SYS$SPECIFIC:[SYSMGR]SYLOGIN.DMF already exists.
Do you want to keep it ? [Y/n]:
See the Administrator Guide to complete the MenuFinder installation.

1.5. What to do at the end of the installation

Before definitively activating MenuFinder for all users of the system, the SYSTEM user can try the product with the command:

$ *@SYS$COMMON:[SYSMGR]MFTEST.COM*

This test is particularly usefull if the system already has a previous version of MenuFinder already installed. This procedure defines the logical names and the symbols for the use of the TUTOR, MASTER and USER menus (see following paragraphs). The new version of MenuFinder is activated for all users when the system is re-booted.

NOTE: MenuFinder may not work correctly on the terminal connected to the console port (OPA0:).

1.6. The TUTOR Menu

The quickest method to learn MenuFinder is that to use the demo sessions menu with the command:

$ *MENU TUTOR*

The user's guide contains all the necessary information for the particular requirements of each type of user.

1.7. The MASTER and USER Menus

The MASTER and USER menus are supplied only as examples and can be modified in successive installations.

In order to call-up the example menus, enter respectively:

$ *MENU MASTER*

$ *MENU USER*

It is possible to use the MASTER and USER menus as a basis for a personalised menu environment by following the instructions in the paragraph "How to create a personalised menu environment."

1.8. How to Complete the Installation

Having tried the product, in order to complete the installation it is necessary to:

1) Enable/Activate MenuFinder for each user , (of a node) with the command

$ *@SYS$COMMON:* [SYSMGR]MFSTART*.COM*

2) Insert in the startup procedure of the node (for example in, SYS$MANAGER:SYSTARTUP_VMS.COM) the line:
$ *@SYS$COMMON:[SYSMGR]MFSTART.COM*

3) Insert in SYS$MANAGER:SYLOGIN.COM the line:
$ *@SYS$COMMON:[SYSMGR]SETMENU.COM*

1.9. How to Create a Personalised Menu Environment

The MASTER and USER menus are intended only as examples to be used as a starting point when creating a menu environment for the system manager and the users.

Modifying these menus directly is discouraged as a successive installation of MenuFinder will lose all such modifications.

To avoid this drawback, it is advisable to create a separate environment where the user specific personalised menus can be inserted.

To guide the system manager in the creation of a separate environment the procedure CONFIG.COM is available. To run it, type the command:

$ *@MENU$DIRECTORY:CONFIG.COM*

The procedure creates a directory for the users' menus and another for the system manager; it inserts the necessary files and defines the logical names in order to allow immediate personalisation of the menus.

To keep the USER and MASTER example menus distinct from the personalised ones, the procedure CONFIG.COM creates the 1USER and 1MASTER menus.

To call them up, use the commands:

$ *MENU 1USER*

$ *MENU 1MASTER*

To modify these menus refer to the User's Guide.

1.10. Send Us Your Comments

Do not esitate to send us your comments or suggestions about MenuFinder writing to mfinfo@itre.com.

To be informed about new MenuFinder versions and patches or to receive periodically free hints and tricks register yourself at http://www.itre.com/mf/download.html.

2. MANAGING MENUFINDER

The management of MenuFinder has to be done by the SYSTEM user and only from the ADM menu.

The management mainly consists of personalising the logical names of the program configuration.
For example, to modify the editor required to change a menu, the definition of the logical name MENU$EDIT must be changed.

A logical name can be defined by the administrator in several file types.
The file type determines the visibility for groups of users.

It may be useful to define logical names:

* for a specific user only;
* for all the users of a node;
* for all the users of all the nodes of a cluster.

In practice:

* if the logical name is defined in the SYS$LOGIN:LOGIN.DMF file of a user, it is then valid only for that user;
* if it is defined in the SYS$SPECIFIC:[SYSMGR]SYLOGIN.DMF file, it is valid for all the users of the node;
* if it is defined in the SYS$COMMON:[SYSMGR]CLULOGIN.DMF file, it is valid for all the users of the cluster.

These three files are analysed in detail below.

2.1. The SYS$LOGIN:LOGIN.DMF file

This file can be created by each user for their own personalisation of the MenuFinder environment (choice of default menu, choice of editor, print commands...)

The presence of this file is not essential as;

* if it is present, the logical names which are defined in it are executed in the LOGIN phase (Process Table);
* if it is not present, the logical names defined by the MenuFinder administrator in the file

  `SYS$SPECIFIC:[SYSMGR]SYLOGIN.DMF` (System Table)

  or

  `SYS$COMMON:[SYSMGR]CLULOGIN.DMF` (System Table)

  remain active.

During the installation of MenuFinder on each node, the file `LOGIN.DMF` is created only for the SYSTEM user. The management of this file in the login directory is usually the user's responsibility.

A user can create or modify this personal file with the command:

$ *ED SYS$LOGIN:LOGIN.DMF*

To activate it, it is necessary to give the command:

$ *@SYS$LOGIN:LOGIN.DMF*

The SYSTEM user may obtain the same result by calling-up the `ADM` menu and choosing option 13.

IMPORTANT: The logical names defined in this file MUST NOT have the /SYSTEM qualifier, as they might (if the user has the correct privileges) also be enabled for all users of the node!

SUGGESTION: In order to have available a single user customised configuration on all cluster nodes, it is useful for the SYSTEM user in a cluster environment to provide a single LOGIN.DMF file. To obtain this, it is sufficient to move the `LOGIN.DMF` from `SYS$SPECIFIC:[SYSMGR]` to `SYS$COMMON:[SYSMGR]`.

2.2. The SYS$SPECIFIC:[SYSMGR]SYLOGIN.DMF file

All logical names that apply to all node users are defined here.

This file is essential for the product to function.

The modification and the activation of this file are made ONLY by the SYSTEM user and ONLY through option 12 of the `ADM` menu.

IMPORTANT: The logical names defined in this file must have the /SYSTEM qualifier. Only in this manner can they be enabled for all node users.

2.3. The SYS$COMMON:[SYSMGR]CLULOGIN.DMF file

The logical names valid for all users of all cluster nodes are defined here.

This file is essential for the product to function.

The installation procedure provides for insertion of a first series of logical names, necessary for immediate use of some of the example menus (`MASTER` menu and USER menu) with MenuFinder.

NOTE: MenuFinder installed successively on another cluster node will recognise the existence of the file `SYS$COMMON:[SYSMGR]CLULOGIN.DMF` and will use it without modification during the start-up phase of the node.

The logical names present in this file are enabled in the start-up phase of the node (before execution of the `SYS$SPECIFIC:[SYSMGR]SYLOGIN.DMF` file).

The modification and the activation of this file is allowed ONLY by the SYSTEM user and ONLY through option 12 of the `ADM` menu.

IMPORTANT: To enable modifications on all cluster nodes it is necessary to activate option 12 of the `ADM` menu on each cluster node.

IMPORTANT: The logical names defined in this file must have the /SYSTEM qualifier, this is the only way they can be enabled for all users of the node!

2.4. The Priority of Logical Names

A logical name is enabled for the user according to the file in which the logical name is defined, but what happens if a logical name is simultaneously defined with different values in the three files?

The priority of a logical name is determined by the sequence of execution of the three files, which is the following:

- The procedure SYS$COMMON:[SYSMGR]MFSTART.COM is executed during the node start-up. It contains, respectively, the commands:

  @SYS$COMMON:[SYSMGR]CLULOGIN.DMF (defines the logical names in System Table)

  @SYS$SPECIFIC:[SYSMGR]SYLOGIN.DMF (redefines the logical names in System Table)

- The procedure SYS$COMMON:[SYSMGR]SETMENU.COM is executed at the LOGIN, which contains the command:

  @SYS$LOGIN:LOGIN.DMF (redefines the logical names in Process Table)

It can be deduced that a logical name defined in LOGIN.DMF has priority over the same logical name defined in SYS$COMMON:[SYSMGR]CLULOGIN.DMF or in SYS$SPECIFIC:[SYSMGR]SYLOGIN.DMF; therefore, a user may redefine the logical names defined by the MenuFinder administrator.

Moreover, a logical name defined in SYS$SPECIFIC:[SYSMGR]SYLOGIN.DMF has priority over the same logical name defined in SYS$COMMON:[SYSMGR]CLULOGIN.DMF because SYS$SPECIFIC:[SYSMGR]SYLOGIN.DMF is executed last during the node start-up.

IMPORTANT: To guarantee the correct sequence of definition of the logical names, it is essential to modify the configuration files;

SYS$COMMON:[SYSMGR]CLULOGIN.DMF

and

SYS$SPECIFIC:[SYSMGR]SYLOGIN.DMF

only from the ADM menu and can only be performed by the SYSTEM user.

3. PERSONALISATION OF THE LOGICAL NAMES

This section provides complementary information to the section "Personalisation of the logical names" of the User's Guide.

This User's guide should be consulted for the description of the logical names which maybe personalised by the users (print commands, etc.).

3.1. The Logical Name MENU$TEST

This logical name must be defined with the value "YES" in order to enable the MenuFinder commands E, D and P.

Example:

$ *DEFINE MENU$TEST YES*

If this logical name is not defined or if it is associated with the value "NO", the use of the MenuFinder commands is disabled at system level for all users.

IMPORTANT: `MENU$TEST` must be defined ONLY by the MenuFinder administrator in the configuration file `SYS$SPECIFIC:[SYSMGR]SYLOGIN.DMF` or `SYS$COMMON:[SYSMGR]CLULOGIN.DMF`.

3.2. The Logical Names MENU$TITLEn

To insert titles on a menu, the instruction `TITLE` can be used inside the relative MDF (Menu Description File, see User's Guide).

If it is necessary to have the same titles in all menus, the use of the instruction `TITLE` may not be practical, in that it would be necessary to repeat the same instructions in all menus and, in case of modification, all MDF's would have to be updated.

The logical names `MENU$TITLEn` can solve this problem because they allow a single definition of common titles which may then be recalled inside any MDF.

This result can be obtained from lines in `SYS$SPECIFIC:[SYSMGR]SYLOGIN.DMF` of the type:

```
$ DEFINE MENU$TITLE1 " "
$ DEFINE MENU$TITLE2 "+GENOA FOODS LTD"
$ DEFINE MENU$TITLE3 " "
```

and then by inserting in each MDF the instruction: `TITLE=%`

NOTE: The instruction `TITLE` must be inserted after the instruction `NAME`. If the latter is not present, the instruction `TITLE` must be the first in the MDF.

Four general titles can be defined in `SYS$SPECIFIC:[SYSMGR]SYLOGIN.DMF`, and n (from 1 to 4) determines the order in which these titles appear on the menu.

Inside the MDF the instruction `TITLE=%` can coexist with other `TITLE` instructions already present.

The sign "+" inside the definition of the title causes the text to be centre justified on the line.

By combining the two types of titles, up to 8 lines can be inserted in the menus.

3.3. Logical Names of the Menus

The logical names of the menus accessible by all node users should be defined in `SYS$SPECIFIC:[SYSMGR]SYLOGIN.DMF`, whereas the menu logical names accessible by all node users of the cluster should be defined in the file `SYS$COMMON:[SYSMGR]CLULOGIN.DMF`; this way, the definition of such names in the various users' `SYS$LOGIN:LOGIN.DMF` can be avoided.

IMPORTANT: IT IS NOT CONVENIENT TO ASSIGN THE NAMES OF THE USERS AS MENU LOGICAL NAMES SINCE CONFLICTS MAY ARISE, FOR EXAMPLE WITH THE MAIL UTILITY.

One solution might be that of adding a prefix (for example, a number) to the menu logical name that you want to associate with the user. For example, the menu name for the user Paolo could be `1PAOLO`.

4. OTHER CONFIGURATION FILES

The other files used by MenuFinder are listed below:

4.1. The SYS$COMMON:[SYSMGR]SETMENU.COM File

It is executed during the LOGIN phase of each user through `SYS$MANAGER:SYLOGIN.COM`.
It initiates the execution of the `SYS$LOGIN:LOGIN.DMF` file.

It is the file that defines the symbols to call-up MenuFinder (`MENU, LMENU, FMENU`,…)

It is a compulsory file and is created during the installation.

4.2. The SYS$COMMON:[SYSMGR]MFSTART.COM File

It initiates the execution of the `SYS$COMMON:[SYSMGR]CLULOGIN.DMF` and the `SYS$SPECIFIC:[SYSMGR]SYLOGIN.DMF` files.

It is executed during the node start-up phase.

It is a compulsory file and is created during the installation.