

PANVALET (MVS only)

The Com-plete PANVALET interface allows a Com-plete terminal user to edit PANVALET library members up to Release 14 using the Com-plete online editor, UEDIT.

This chapter covers the following topics:

- Installation Overview
 - Installation Procedure (MVS)
-

Installation Overview

Before a PANVALET library can be accessed, two load modules must be created using the linkage editor. These modules are standard PANVALET routines provided by Pansophic Systems with additional modules added by the linkage editor. These modules provide compatibility with the Com-plete online environment and allow the Com-plete printout spooling facility to be used for spooling the PANVALET reports.

The PANVALET libraries to be accessed by UEDIT must be defined in the UEDTB1 table described in **Com-plete Files and Associated User Files**, and in **UEDTB1 Entry DSECT**.

Installation Procedure (MVS)

Step 1: Link edit and zap the PANVALET modules.

The member PANLNK contains an inline procedure to link edit the PANVALET modules. This procedure must be modified to use the load module library that contains the PANVALET modules at your installation. Do this by modifying the keyword parameter PANLIB default value in the PANLNK PROC.

Proceed as follows:

1. **Link edit FG PAN23.**

The PAN23 step is used to link edit FG PAN23, creating a load module on the Com-plete user load module library defined in the PANLNK PROC as LLIB. FG PAN23 is a module accessed by PANVALET during the processing of abnormal conditions. The execution of this link edit step prevents an abend of UEDIT (S806) from occurring because PANVALET could not find FG PAN23.

2. **Link edit COMPAM.**

The COMPAM step is used to link edit COMPAM, the load module used by UEDIT for read-access to the PANVALET libraries. COMPAM is identical to the PANVALET access method module PAM provided by Pansophic Systems.

3. **Link edit COMPAN**

The COMPAN step is used to link edit COMPAN, the load module used by UEDIT for write-access to the PANVALET libraries. COMPAN is identical to the PANVALET module PAN#1.

4. Link edit PVPVLAMS

The PVPVL step is used to link edit PVPVLAMS, PANVALET's general access module. PVPVLAMS is comprised of PVPVLAMS and Com-plete's PANVALET interface exit \$USTIMER. You must place this module in an APF-authorized library.

\$USTIMER functions as an interval timer routine that will do a ROLLOUT (if Com-plete), or a STIMER SVC (if TSO). The timer interval is currently set at two seconds.

The \$USTIMER module is supplied in source format. If this requires changes to the time interval, the module must be modified and reassembled.

5. Alter the print options

IMBLIST can be used to determine the offsets of the individual entry points in the module and the options modified as required. The zap has the format:

```
NAME COMPAN COMPAN
VER xxx yy    <-- old value
RER xxx zz    <-- new value
```

The entry point names and the functions of the five bytes are listed in the following table:

Entry Name	Function														
OPTION	<p>A switch used to control the printing of reports. Bit settings are as follows:</p> <table border="0"> <tr> <td data-bbox="646 344 716 373">X'80'</td> <td data-bbox="776 344 1252 407">Output all reports to the PANPRNT data set.</td> </tr> <tr> <td data-bbox="646 436 716 466">X'40'</td> <td data-bbox="776 436 1252 499">Output no reports to the PANPRNT data set.</td> </tr> <tr> <td data-bbox="646 529 716 558">X'20'</td> <td data-bbox="776 529 1240 592">Output only reports selected by the user code to the PANPRNT data set.</td> </tr> <tr> <td data-bbox="646 621 716 651">X'08'</td> <td data-bbox="776 621 1154 684">Output all reports to the selected terminal.</td> </tr> <tr> <td data-bbox="646 714 716 743">X'04'</td> <td data-bbox="776 714 1154 777">Output no reports to the selected terminal.</td> </tr> <tr> <td data-bbox="646 806 716 835">X'02'</td> <td data-bbox="776 806 1240 869">Output only reports selected by the user code to the selected terminal.</td> </tr> <tr> <td data-bbox="646 898 716 928">X'01'</td> <td data-bbox="776 898 1252 1003">Select a user terminal for reports (otherwise, the terminal assigned as the screen-to-hardcopy terminal is selected).</td> </tr> </table>	X'80'	Output all reports to the PANPRNT data set.	X'40'	Output no reports to the PANPRNT data set.	X'20'	Output only reports selected by the user code to the PANPRNT data set.	X'08'	Output all reports to the selected terminal.	X'04'	Output no reports to the selected terminal.	X'02'	Output only reports selected by the user code to the selected terminal.	X'01'	Select a user terminal for reports (otherwise, the terminal assigned as the screen-to-hardcopy terminal is selected).
X'80'	Output all reports to the PANPRNT data set.														
X'40'	Output no reports to the PANPRNT data set.														
X'20'	Output only reports selected by the user code to the PANPRNT data set.														
X'08'	Output all reports to the selected terminal.														
X'04'	Output no reports to the selected terminal.														
X'02'	Output only reports selected by the user code to the selected terminal.														
X'01'	Select a user terminal for reports (otherwise, the terminal assigned as the screen-to-hardcopy terminal is selected).														
SPPOS	Offset in the PANVALET user code to be tested for the X'02' option selected above.														
SPCHAR	Value of the report selection code for the X'02' option selected above.														
PRPOS	Offset in the PANVALET user code to be tested for the X'20' option selected above.														
PRCHAR	Value of the report selection code for the X'20' option selected above.														

Step 2: Modify the Com-plete job stream JCL

The Com-plete execution procedure must be modified to include the data definition statements for the files required by PANVALET, as well as for each PANVALET library to be accessed by UEDIT.

PANVALET requires that four data sets be identified by the JCL in addition to the libraries to be accessed by UEDIT. The first required data set is used to pass the PANVALET control statements to the routine COMPAN (PAN#1). This data set must be defined as an 80-byte fixed block sequential file, and must be large enough to hold the maximum number of control statements that will be required by PAN#1. This control statement file, identified by the name PANSEQ, serves the same function as the usual SYSIN DD statement used by PAN#1.

The other required statements are used for output of reports generated by PAN#1. These data sets must be defined as 133-byte fixed block data sets and identified by the names PANPRNT, PANDD2, and SYSPUNCH. The first character of each record is used for ANSI carriage control. Note that if these data sets are not defined as SYSOUT data sets, they should be defined with DISP=MOD and be large enough to contain all the reports generated by PANVALET.

All the required files can be temporary files. Examples of the PANSEQ and PANPRNT DD statements are as follows:

```
//PANSEQ DD DSN=&&X,UNIT=SYSDA,SPACE=(CYL,5),DISP=(,DELETE),
//          DCB=(RECFM=FB,LRECL=80,BLKSIZE=4000)
//PANPRNT DD DSN=COM.PAN.REPORTS,UNIT=SYSDA,
//          DISP=(MOD,KEEP),
//          DCB=(RECFM=FB,LRECL=133,BLKSIZE=2660)
           or
//PANPRNT DD SYSOUT=A
```

PANDD2 and SYSPUNCH are specified similarly to PANPRNT.

Each PANVALET library identified in the JCL must also be identified to UEDIT by adding a statement to the UEDTB1 table. An example of such a set of definitions follows.

In the Com-plete JCL for a PANVALET library identified by:

```
//MYPANLIB DD DSN=USER.PAN.MYLIB,DISP=SHR
```

the corresponding UEDTB1 table entry would be:

```
CMEDTB1 ID=MP,DSN=MYPANLIB,ACM=PANVALET
```

Note that the ID is the two-character value assigned by the user, and the DSN is the DD name, not the data set name.

The two PANVALET load modules made resident within Com-plete will require additional storage from within the Com-plete region. It will therefore be necessary to increase the REGION parameter for the Com-plete job step. In the case of VS1, the size of the partition in which Com-plete runs must be increased. The increase in size should be the combined sizes of the two load modules.

Step 3: Modify the Com-plete sysparms

The Com-plete sysparms must be modified in order to cause Com-plete to include COMPAM and COMPAN as resident programs. Do this by adding the following statements to the SYSPARM member used to set Com-plete options:

```
RESIDENTPAGE=COMPAM
RESIDENTPAGE=COMPAN
```

**Warning:**

Do not modify the COMPAN or COMPAN load modules used by Com-plete while Com-plete is running. During normal use of the PANVALET interface by UEDIT, these modules are refreshed in the Com-plete resident program area. Changing the attributes of these modules while Com-plete is active can have catastrophic results.

Step 4: Add the PANVALET modules to MLPA (MVS and XA only)

Add PVPVLAMS, FGPAN23, and PVEXTUSR to the MLPA list, and IPL if necessary.

Step 5: Name the user exit (optional)

An optional user exit is available that allows users to enforce installation standards for the PANVALET ++ADD and ++UPDATE statements. This exit must be named UXEEX4. Control is passed to UXEEX4 for each of these statements generated by UEDIT or explicitly inserted by the UEDIT user. This routine may be included in the link edit of the UEDIT subprogram UEBP, as the operating system or Com-plete resident program, or in the Com-plete program library. Note that if it is present as a resident program, UXEEX4 must be reentrant.

An example of this exit is provided in the distributed source library COM.SOURCE. See **Security and User Exit Facilities** for more information on the coding and installation of UXEEX4.