VisualAge Pacbase 2.5

VISUALAGE PACBASE - ENDEVOR INTERFACE IBM MVS, IMS ENVIRONMENT & INSTALLATION

DDENDIMS251A

Note

Before using this document, read the general information under "Notices" on the next page.

According to your license agreement, you may consult or download the complete up-to-date collection of the VisualAge Pacbase documentation from the VisualAge Pacbase Support Center at:
http://www.ibm.com/software/ad/vapacbase/support.htm

Consult the Catalog section in the Documentation home page to make sure you have the most recent edition of this document.

First Edition (October 1999)

This edition applies to the following licensed program:

- VisualAge Pacbase Version 2.5

Comments on publications (including document reference number) should be sent electronically through the Support Center Web site at:
http://www.ibm.com/software/ad/vapacbase/support.htm
or to the following postal address:
IBM Paris Laboratory
VisualAge Pacbase Support
30, rue du Château des Rentiers
75640 PARIS Cedex 13
FRANCE

When you send information to IBM, you grant IBM a nonexclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

NOTICES

References in this publication to IBM products, programs, or services do not imply that IBM intends to make these available in all countries in which IBM operates. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Subject to IBM's valid intellectual property or other legally protectable rights, any functionally equivalent product, program, or service may be used instead of the IBM product, program, or service. The evaluation and verification of operation in conjunction with other products, except those expressly designated by IBM, are the responsibility of the user.

IBM may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

Intellectual Property and Licensing
International Business Machines Corporation
North Castle Drive, Armonk, New-York 10504-1785
USA

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of information which has been exchanged, should contact:

IBM Paris Laboratory
SMC Department
30, rue du Château des Rentiers
75640 PARIS Cedex 13
FRANCE

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

IBM may change this publication, the product described herein, or both.

TRADEMARKS

IBM is a trademark of International Business Machines Corporation, Inc.
AIX, AS/400, CICS, CICS/MVS, CICS/VSE, COBOL/2, DB2, IMS, MQSeries, OS/2, PACBASE, RACF, RS/6000, SQL/DS, TeamConnection, and VisualAge are trademarks of International Business Machines Corporation, Inc. in the United States and/or other countries.

Java and all Java-based trademarks and logos are trademarks of Sun Microsystems, Inc. in the United States and/or other countries.
Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States and/or other countries.
UNIX is a registered trademark in the United States and/or other countries licensed exclusively through X/Open Company Limited.

All other company, product, and service names may be trademarks of their respective owners.

TABLE OF CONTENTS

1. INTRODUCTION

1.1. INTRODUCTION

INTRODUCTION

ENDEVOR is an integrated configuration management product which manipulates source programs, load modules, JCL, etc. It provides control functions and development guidelines for applications and for their operational implementation.

The VA Pac/ENDEVOR Interface enables to integrate VA Pac generated objects into the ENDEVOR management environment.

The Interface permits the user to know:

- in VA Pac, the ENDEVOR "target" contexts which manage the generated objects: Environment, Stage, System, Sub-system and Type;

- in ENDEVOR, the "source" context of these VA Pac objects: Library Code, Session Number, User Code, Generation Date and Time.

DEFINITIONS

From VA Pac entities, the Interface produces two objects which have to be imported into ENDEVOR:

- The first one is the generated source itself, coming from VA Pac entities;

- The second one contains the generation context of the first one (i.e. the Library Code, the Session Number, the User Code, and the Generation Date and Time). This object has a purely functional role, and cannot be manipulated by the user outside the Interface.

Throughout this manual, we name:

- VA Pac ELEMENT any Source or Load object that comes from a VA Pac generation and that is managed by ENDEVOR;

- INFOPAC ELEMENT the associated object that contains the context of the VA Pac ELEMENT generation.

To each VA Pac ELEMENT corresponds one INFOPAC ELEMENT, which has the same name, and is located in the same ENDEVOR context.

2. DESCRIPTION OF THE INTERFACE ELEMENTS

2.1. VA PAC ELEMENT

VA Pac ELEMENT

The VA Pac/ENDEVOR interface manages objects generated from the following VA Pac entities:

. Program entity (P)

. Screen entity (O)

. Database Block entity (B)

. the Copy clauses of the Data Structure entity (D)

Each generated VA Pac object becomes a VA Pac ELEMENT in ENDEVOR.

The VA Pac ELEMENT can be managed in ENDEVOR as a generated source compilable by an ENDEVOR Process, or as a Load form.

A VA Pac entity can correspond to several generated objects found in several different ENDEVOR contexts. The entity can therefore correspond to several VA Pac ELEMENTS in ENDEVOR.

2.2. INFOPAC ELEMENT

INFOPAC ELEMENT

A VA Pac ELEMENT is always accompanied by an INFOPAC ELEMENT in the same ENDEVOR environment. The INFOPAC ELEMENT contains the generation context of the VA Pac ELEMENT. Its presence in ENDEVOR allows the user to find the original corresponding VA Pac ELEMENT in VA Pac. The two ELEMENTs have the same code which is the external name of the VA Pac entity. The only difference between the ELEMENTs is the TYPE.

By using appropriate Before and After commands, the generation of a VA Pac entity produces the VA Pac ELEMENT followed automatically by the INFOPAC ELEMENT.

The INFOPAC ELEMENT contains the following information:

. Nature of the VA Pac ELEMENT (=Entity code: P, O, B, D)

. Generated VA Pac entity occurrence code

. VA Pac ELEMENT code (VA Pac external name)

. VA Pac database code

. Library code

. Generation session number

. Generation date

. Generation time

. User code

2.3. VA PAC OBJECT TYPE TABLE

VA Pac OBJECT TYPE TABLE

Objects coming from the generation of VA Pac entity occurrences must have specific TYPEs in ENDEVOR. These TYPEs permit to recognize the objects and to differentiate them from the other objects managed in ENDEVOR.

In an ENDEVOR context, a VA Pac ELEMENT is always accompanied by an INFOPAC ELEMENT which has the same code but a different TYPE. The interface relate them using the correspondence of pre-defined types in the TYPEs TABLE.

The TYPEs TABLE contains all the TYPEs reserved for VA Pac ELEMENTS managed by ENDEVOR and all the corresponding TYPEs of INFOPAC ELEMENTS.

A VA Pac ELEMENT TYPE corresponds to one and only one INFOPAC ELEMENT TYPE. A VA Pac ELEMENT TYPE already defined cannot be used as an INFOPAC ELEMENT TYPE, and vice versa.

WARNING: The Endevor administrator must update the VA Pac ELEMENTS TYPEs and the INFOPAC ELEMENTS TYPES in the TYPE DEFINITION screen BEFORE using the interface.

The Endevor administrator must also enter a 'I' in the regression severity field (REGR SEV field in the TYPE DEFINITION screen).

It is advisable not to reuse TYPEs that already exist in Endevor and assign them to VA Pac ELEMENTs.

2.4. TARGET LIBRARY/SESS TABLE

TARGET LIBRARY/SESSION TABLE

The ENDEVOR environments which manage the generated objects are stored in VA Pac in the form of occurrences of the predefined .NDVLM User Entity.

As for all entities in the VA Pac Specifications Dictionary, these occurrences (UEO) can be found in different libraries and sessions. By default, ENDEVOR data is uploaded to the same VA Pac environement (library/session) as the generation environment.

The TARGET LIBRARY/SESSION TABLE allows the administrator to personalize the library/session environments used to manage ENDEVOR data in VA Pac. The table makes it possible to choose a target library for one or more generation libraries and a target session for one or more ranges of generation sessions. The target session can be the current session or a frozen session.

Example: Target library

A sample VA Pac database can be represented as follows:

```
        CEN
         :
 -----------------
 :               :
A00             B00
 :               :
 :          -----------
 :          :         :
A01        B01       B02
```

The administrator can decide to store the ENDEVOR data of the generated objects from all the libraries in CEN, and to keep the data from the A00 library in A00.

For this example, the Target Library/Session Table must therefore contain the following parameters:

Target Library	Line Number	Generation Library	Comment
CEN	100	B00	
CEN	110	B01	
CEN	120	B02	
CEN	130	A01	
CEN	140	CEN	
A00	100	A00	

Example: Target session

In this example, the administrator decides that:

. the ENDEVOR data of the VA Pac objects generated in a session older than or equal to 1200T must uploaded into session 1200T;

. the data of a session between 1300T and 1400T must remain in the generation session;

. the data of a session between 1501T and 2100T must be be uploaded into session 2100T;

. the data of a session higher than 2100T must be uploaded into the current session.

For this example, the TARGET SESSION TABLE must contain the following values:

Target Session	Line Number	Generation Session (Start)	Generation Session (End)
1200T	100	0001T	1200T
1300T	110	1300T	1300T
1400T	120	1400T	1400T
2100T	130	1501T	2100T
9999Z	140	2101T	9999Z

2.5. PREDEFINED USER ENTITY

THE INTERFACE PREDEFINED USER ENTITIES

The Endevor context of VA Pac generated objects is memorized in the VA Pac repository. The VA Pac user can then locate each generated object in the Endevor environment.

The default Endevor target environments where VA Pac ELEMENTs are imported can be defined in the predefined .NDENV User Entity. A default Endevor target environment corresponds to an occurrence of this User Entity. Each default Endevor target environment can present a list of VA Pac entities to be generated. This User Entity has a description for each entity type.

Data concerning the Endevor context of ONE VA Pac entity occurrence is stored in ONE occurrence of the predefined .NDVLM User Entity.

A VA Pac entity occurrence is related to an occurrence of the .NDVLM or .NDENV User Entity via the .NDVRL predefined relationship.

Before the first generation, the user must define the Endevor target environment where the VA Pac ELEMENT must be imported. This target environment may be defined at three different levels:

. in the definition of the .NDENV User Entity occurrence. The generated object must be present in one of the descriptions of the entity;

. in the definition of the .NDVLM User Entity occurrence;

. in the description of the .NDVLM User Entity occurrence.

Some parameters related to the processing of the generated object during its import into Endevor (the TYPE, the PROCESSOR, the GROUP, the CCID,...) may be defined at three different levels:

. in the definition of the .NDENV User Entity occurrence;

. in one of the descriptions of the .NDENV User Entity occurrence;

. in the definition of the .NDVLM User Entity occurrence.

The default Endevor target environment or the parameters related to the processing of the generated element are taken into account in the following priority order:

. the .NDVLM User Entity occurrence description;

. the .NDVLM User Entity occurrence definition;

. the .NDENV User Entity occurrence description;

. the .NDENV User Entity occurrence definition.

2.6. .NDENV PREDEFINED USER ENTITY

.NDENV USER ENTITY

The .NDENV User Entity supplied during installation allows the user to define the default Endevor Environments and the VA Pac ELEMENTs which are imported into them on generation. The call code of .NDVENV is 7M. Its structure comprises the following parts:

- the defintion of the default Endevor target environment for the generation of the VA Pac ELEMENT (definition);

- the list of the programs to be generated in this environment (-D1 description);

- the list of the screens to be generated in this environment (-D2 description);

- the list of the DBD blocks to be generated in this environment (-D3 description);

- the list of the Copy Data to be generated in this environment (-D4 definition).

DEFINITION OF THE DEFAULT ENDEVOR TARGET ENVIRONMENT

An occurrence of the .NDENV User Entity corresponds to an Endevor target environment into which the VA Pac ELEMENTs are imported. It must be defined in the VA Pac Library and Session which store the data concerning the Endevor environments, in the VA Pac ELEMENT which corresponds to an occurrence of the .NDVLM User Entity (see TARGET LIBRARIES AND SESSIONS TABLE).

The Endevor target environment comprises the following information:

. the ENVIRONMENT code

. the SYSTEM code

. the SUBSYSTEM code

. the VA Pac ELEMENT TYPE

. the PROCESSOR GROUP of the Processes to be activated

. the CCID number

. the OVERRIDE SIGNOUT option

. the DELETE INPUT SOURCE option

. the VA Pac ELEMENT version number

. the Comments

- The stage is implicit. The generated object is always imported into Endevor in stage 1.

- For a screen-type VA Pac ELEMENT, to differentiate the TYPE of the Screen Program from the TYPE of the Map, it is possible to code the TYPE of the Screen Program in the first 8 positions, and the TYPE of the Map in the last 8 positions of the "Element type" field.

- The PROCESSOR GROUP is optional. The user can indicate a Processor Group other than that defined by default in Endevor for the element TYPE. For a Screen-type VA Pac ELEMENT, it is possible to indicate a PROCESSOR GROUP for the Screen Program in the first 8 positions and a PROCESSOR GROUP for the MAP in the last 8 positions of the "Processor Group" field.

- The CCID number is optional. It is coded on the first 12 positions of the "CCID/OvSig/Del/Ver" field.

- The OVERRIDE SIGNOUT option imports into Endevor a new version of the VA Pac ELEMENT assigned to a user code different from that of the current version. To select this option specify the value "Y" in the 14th position of the "CCID/OvSig/Del/Ver" field. All other values are ignored.

- The DELETE INPUT SOURCE option deletes the object source once imported into Endevor. To select this option, specify the value "Y" in the 16th position of the "CCID/OvSig/Del/Ver" field. All other values are ignored.

- To assign a version number to the VA Pac ELEMENT to be imported into Endevor, specify a value included between 01 and 99 in positions 18 and 19 in the "CCID/OvSig/Del/Ver" field. You must reinitialize this value after the current generation to avoid conflicts between the version number in Endevor in the following generations.

- The comments in the COMMENTS field are optional.

--> VA Pac Choice: $7Meeeeee (with eeeeee: UEO occurrence code).

```
 --------------------------------------------------------------------------------
 !                                                                              !
 !                       DOCUMENTATION                       *PTXX.PDEV.BMS.651 !
 !                                                                              !
 ! TYPE               : 7M           USER ENTITY........: .NDENV                !
 ! -----------------------------------------------------------------------------!
 ! Dflt Environ.        ENVIR1                                                  !
 ! U.E. ITEM NAME     : DEVELOPMENT                                             !
 !                                                                              !
 ! Environment        : DEMO                                                    !
 ! System             : FINANCE                                                 !
 ! Sub-system         : ACCTREC                                                 !
 ! Element Type       : PCOBOL                                                  !
 ! Processor Group    : PACPROC                                                 !
 ! CCID/OvSig/Del/Ver : 123456789012 Y Y                                        !
 ! Comments           : COMMENTS                                                !
 !                                                                              !
 !                                                                              !
 !                                                                              !
 !                                                                              !
 !                                                                              !
 ! EXPLICIT KEYWORDS..:                                                         !
 !                                                                              !
 ! SESSION NUMBER.....: 0721        LIBRARY......: BMS    LOCK....:             !
 !                                                                              !
 ! O: C1 CH: $7Menvir1                  ACTION:                                 !
 --------------------------------------------------------------------------------
```

LIST OF VA Pac ENTITIES TO BE GENERATED

Several VA Pac entities can be generated in the same Endevor Environment. At this level the user indicates the list of VA Pac entities to be generated in this same Endevor Environment.

The .NDENV User Entity has 4 descriptions. Each corresponds to the list of an entity type to be generated. They are:

. the list of programs (-D1 description)

. the list of screens (-D2 description)

. the list of DBDs (-D3 description)

. the list of Copy Data (-D4 desription)

In each description, the user can indicate parameters different from those indicated on the target environment definition. The parameters defined at this level have priority over those defined at the occurrence definition level.

The additional parameters defined in each entity to be generated are:

. the VA Pac ELEMENT TYPE

. the PROCESSOR GROUP of the Processes to be activated

. the CCID number

. the OVERRIDE SIGNOUT option

. the DELETE INPUT SOURCE option

. the VA Pac ELEMENT version number

. the Comments

- The stage is implicit. The generated object is always imported into Endevor by the stage 1.

- For the Screen-type VA Pac ELEMENT, to differentiate the Screen Program TYPE from the Map TYPE, it is possible to code the Screen Program TYPE in the first 8 positions, and the Map TYPE in the last 8 positions of the "Element Type" field.

- The PROCESSOR GROUP is optional. The user can indicate a Processor Group other than that defined by default in Endevor for the element TYPE. For a Screen-type VA Pac ELEMENT, it is possible to indicate a PROCESSOR GROUP for the Screen Program in the first 8 positions and a PROCESSOR GROUP for the MAP in the last 8 positions of the "Processor Group" field.

- The CCID number is optional. It is coded on the first 12 positions of the "CCID/OvSig/Del/Ver" field.

- The OVERRIDE SIGNOUT option imports into Endevor a new version of the VA Pac ELEMENT assigned to a user code different to that of the current version. To select this option specify the value "Y" in the 14th position of the "CCID/OvSig/Del/Ver" field. All other value is ignored.

- The DELETE INPUT SOURCE option deletes the object source once imported into Endevor. To select this option, specify the value "Y" in the 16th position of the "CCID/OvSig/Del/Ver" field. All other value is ignored.

- To assign a version number to a VA Pac ELEMENT to be imported into Endevor, specify a value included between 01 and 99 in positions 18 and 19 of the "CCID/OvSig/Del/Ver" field. You must reinitialize this value after the current generation to avoid conflict of version number in Endevor in the following generations.

- The comments in the COMMENTS field are optional.

--> VA Pac Choice: $7MeeeeeeDn (with eeeeee: EUO occurrence code and n: the description number).

```
--------------------------------------------------------------------------------
!                                                                              !
!                       DOCUMENTATION                       *PTXX.PDEV.BMS.651 !
! Program                        7M ENVIR1 1 DEVELOPMENT                       !
!                                                                              !
! A LIN : Progr. Type     Process  CCID/OvSig/Del/Ver  Comments                !
!   001 : PROGR1                                                               !
!       : ______ ________ ________ ___________________ ___________________     !
!       : ______ ________ ________ ___________________ ___________________     !
!       : ______ ________ ________ ___________________ ___________________     !
!       : ______ ________ ________ ___________________ ___________________     !
!       : ______ ________ ________ ___________________ ___________________     !
!       : ______ ________ ________ ___________________ ___________________     !
!       : ______ ________ ________ ___________________ ___________________     !
!       : ______ ________ ________ ___________________ ___________________     !
!       : ______ ________ ________ ___________________ ___________________     !
!       : ______ ________ ________ ___________________ ___________________     !
!       : ______ ________ ________ ___________________ ___________________     !
!       : ______ ________ ________ ___________________ ___________________     !
!       : ______ ________ ________ ___________________ ___________________     !
!       : ______ ________ ________ ___________________ ___________________     !
!       : ______ ________ ________ ___________________ ___________________     !
!       : ______ ________ ________ ___________________ _____________________   !
!       : ______ ________ ________ ___________________ _____________________   !
! *** END ***                                                                  !
! O: C1 CH: $7Menvir1 D1                                                       !
--------------------------------------------------------------------------------
```

```
--------------------------------------------------------------------------------
!                                                                              !
!                       DOCUMENTATION                       *PTXX.PDEV.BMS.651 !
! Screen                         7M ENVIR1 2 DEVELOPMENT                       !
!                                                                              !
! A LIN : Screen Screen/Map Type  Screen/Map Proc. CCID/OvSig/Del/Ver  Comments!
!   001 : SCR1                                                                 !
!       : ______ ________________ ________________ ___________________ ________!
!       : ______ ________________ ________________ ___________________ ________!
!       : ______ ________________ ________________ ___________________ ________!
!       : ______ ________________ ________________ ___________________ ________!
!       : ______ ________________ ________________ ___________________ ________!
!       : ______ ________________ ________________ ___________________ ________!
!       : ______ ________________ ________________ ___________________ ________!
!       : ______ ________________ ________________ ___________________ ________!
!       : ______ ________________ ________________ ___________________ ________!
!       : ______ ________________ ________________ ___________________ ________!
!       : ______ ________________ ________________ ___________________ ________!
!       : ______ ________________ ________________ ___________________ ________!
!       : ______ ________________ ________________ ___________________ ________!
!       : ______ ________________ ________________ ___________________ ________!
!       : ______ ________________ ________________ ___________________ ________!
!       : ______ ________________ ________________ ___________________ ________!
!       : ______ ________________ ________________ ___________________ ________!
! *** END ***                                                                  !
! O: C1 CH: $7Menvir1 D2                                                       !
--------------------------------------------------------------------------------
```

```
--------------------------------------------------------------------------------
!                                                                              !
!                       DOCUMENTATION                       *PTXX.PDEV.BMS.651 !
! DBD                            7M ENVIR1 3 DEVELOPMENT                       !
!                                                                              !
! A LIN : DBD    Type     Process  CCID/OvSig/Del/Ver  Comments                !
!   001 : DBD1                                                                 !
!       : ______ ________ ________ ___________________ ___________________     !
!       : ______ ________ ________ ___________________ ___________________     !
!       : ______ ________ ________ ___________________ ___________________     !
!       : ______ ________ ________ ___________________ ___________________     !
!       : ______ ________ ________ ___________________ ___________________     !
!       : ______ ________ ________ ___________________ ___________________     !
!       : ______ ________ ________ ___________________ ___________________     !
!       : ______ ________ ________ ___________________ ___________________     !
!       : ______ ________ ________ ___________________ ___________________     !
!       : ______ ________ ________ ___________________ ___________________     !
!       : ______ ________ ________ ___________________ ___________________     !
!       : ______ ________ ________ ___________________ ___________________     !
!       : ______ ________ ________ ___________________ ___________________     !
!       : ______ ________ ________ ___________________ ___________________     !
!       : ______ ________ ________ ___________________ ___________________     !
!       : ______ ________ ________ ___________________ _____________________   !
!       : ______ ________ ________ ___________________ _____________________   !
! *** END ***                                                                  !
! O: C1 CH: $7Menvir1 D3                                                       !
--------------------------------------------------------------------------------
```

```
--------------------------------------------------------------------------------
!                                                                              !
!                       DOCUMENTATION                       *PTXX.PDEV.BMS.651 !
! Copy Data                      7M ENVIR1 4 DEVELOPMENT                       !
!                                                                              !
! A LIN : Data   Type     Process  CCID/OvSig/Del/Ver  Comments                !
!   001 : DBD1                                                                 !
!       : ______ ________ ________ ___________________ ___________________     !
!       : ______ ________ ________ ___________________ ___________________     !
!       : ______ ________ ________ ___________________ ___________________     !
!       : ______ ________ ________ ___________________ ___________________     !
!       : ______ ________ ________ ___________________ ___________________     !
!       : ______ ________ ________ ___________________ ___________________     !
!       : ______ ________ ________ ___________________ ___________________     !
!       : ______ ________ ________ ___________________ ___________________     !
!       : ______ ________ ________ ___________________ ___________________     !
!       : ______ ________ ________ ___________________ ___________________     !
!       : ______ ________ ________ ___________________ ___________________     !
!       : ______ ________ ________ ___________________ ___________________     !
!       : ______ ________ ________ ___________________ ___________________     !
!       : ______ ________ ________ ___________________ ___________________     !
!       : ______ ________ ________ ___________________ ___________________     !
!       : ______ ________ ________ ___________________ _____________________   !
!       : ______ ________ ________ ___________________ _____________________   !
! *** END ***                                                                  !
! O: C1 CH: $7Menvir1 D4                                                       !
--------------------------------------------------------------------------------
```

2.7. .NDVLM PREDEFINED USER ENTITY

THE .NDVLM USER ENTITY

The .NDVLM User Entity supplied at the time of installation supports the Endevor information of a VisualAge Pacbase ELEMENT. The call type of .NDVLM is 7N. It is constituted of the following parts:

- the definition of the default Endevor target environment for the generation of the VisualAge Pacbase ELEMENT (definition),

- the description of the Endevor environments where the generated objects of the VisualAge Pacbase ELEMENT are managed (-D1 description),

- the characteristics of the last action carried out on the objects of the VisualAge Pacbase ELEMENT (-D2 description),

- the characteristics of the different versions of the VisualAge Pacbase ELEMENT in Endevor (-D9 description).

If there is no occurrence of the .NDVLM User Entity corresponding to the occurrence of the VisualAge Pacbase entity to manage, the Interface creates it when the Endevor contexts of the entity are updated in VisualAge Pacbase. Then the Interface updates the contents of this UEO after the successive generations.

DEFINITION OF THE DEFAULT ENDEVOR TARGET ENVIRONMENT

Before the first generation, if the definition of the VA PAC ELEMENT is not in the list of entities to be genreated in an occurrence of the .NDENV User Entity, the user must create an occurrence of the .NDVLM User Entity to indicate the ENDEVOR target environment where the VA Pac ELEMENT will be generated. The occurrence must be defined in the VA Pac library and session where the ENDEVOR target environment information is stored (see 'Target Libraries/Sessions Table').

Rule: The User Entity Occurrence code that is created must be identical to the VA Pac entity code to be generated.

The ENDEVOR target environment must include the following information:

. ENVIRONMENT code

. SYSTEM code

. SUB-SYSTEM code

. VA Pac ELEMENT TYPE

. PROCESSOR GROUP of the Process to activate

. VA Pac ELEMENT (Program) code or

. VA Pac ELEMENT (Screen) code or

. VA Pac ELEMENT (Database Block) code or

. VA Pac ELEMENT (Copy Data) code

. CCID number

. OVERRIDE SIGNOUT option

. DELETE INPUT SOURCE option

. VA Pac ELEMENT version number

. Comments

The VA Pac ELEMENT code allows to associate the User Entity Occurrence with the VA Pac Entity. The VA Pac entity must then be defined in a library whose level is equal to or higher than that of the occurrence.

The Stage is implicit. The generated object is always imported into ENDEVOR in Stage 1.

The PROCESSOR GROUP is optional. It allows the user to indicate a PROCESSOR GROUP other than that defined by defulat in Endevor for the ELEMENT TYPE. . For a VA Pac ELEMENT of Screen type, it is possible to indicate a PROCESSOR GROUP for the Screen Program on the first 8 positions and a PROCESSOR GROUP for the MAP on the last 8 positions of the PROCESSOR GROUP area.

For a VA Pac ELEMENT of Screen type, to differentiate the TYPE of the Screen Program from the TYPE of the MAP, it is possible to enter the TYPE of the Screen Program on the first 8 positions, and the TYPE of the MAP on the last 8 positions of the ELEMENT TYPE field.

The CCID number is optional. It is coded on the first 12 positions of the "CCID/OvSig/Del/Ver" field.

The OVERRIDE SIGNOUT option enables you to import, into Endevor, a new version of the VA Pac ELEMENT assigned to a user code different from that of the current version. To select this option, enter a "Y" in the 14th position of the "CCID/OvSig/Del/Ver" field. Any other value will be ignored for this option.

The DELETE INPUT SOURCE option deletes the source of the object imported into Endevor. To select this option, enter a "Y" in the 16th position of the "CCID/OvSig/Del/Ver" field. Any other value will be ignored for this option.

The VA Pac ELEMENT to be imported in Endevor can be assigned a version number by coding a value between 01 and 99 in positions 18 and 19 in the "CCID/OvSig/Del/Ver" field. You must reinitialize this number after the current generation in order to avoid conflicting version numbers in Endevor in the next generations.

The comments in the COMMENTS field are optional.

Upon generation before import, this description enables the Interface to complete the Endevor context with the chosen options.

The call type of the .NDVLM User Entity is 7N.
VA Pac choice: $7Neeeeee (with eeeeee: UEO occurrence code)

```
 --------------------------------------------------------------------------------
 !                                                                              !
 !                      DOCUMENTATION                        *PTXX.PDEV.BMS.651 !
 !                                                                              !
 ! TYPE               : 7N           USER ENTITY........: .NDVLM                !
 ! ------------------------------------------------------------------------------
 ! ENDEVOR ELEMENT      PROGR1                                                  !
 ! U.E. ITEM NAME     : COMMAND CHECK                                           !
 !                                                                              !
 ! ENVIRONMENT        : DEMO                                                    !
 ! SYSTEM             : FINANCE                                                 !
 ! SUB-SYSTEM         : ACCTREC                                                 !
 ! ELEMENT TYPE       : PCOBOL                                                  !
 ! PROGRAM CODE       : PPROGR1                                                 !
 ! SCREEN CODE        :                                                         !
 ! DBD CODE           :                                                         !
 ! DATA CODE          :                                                         !
 ! PROCESS GROUP      : PACPROC                                                 !
 ! CCID/OVSIG/DEL/VER : 123456789012                                            !
 ! COMMENTS           : COMMENTS                                                !
 !                                                                              !
 ! EXPLICIT KEYWORDS..:                                                         !
 !                                                                              !
 ! SESSION NUMBER.....: 0721        LIBRARY......: BMS    LOCK....:             !
 !                                                                              !
 ! O: C1 CH: $7NPROGR1                   ACTION:                                !
 --------------------------------------------------------------------------------
```

DESCRIPTION OF THE ENTITY'S ENDEVOR ENVIRONMENTS

To a VA Pac entity can correspond several generated objects managed in several ENDEVOR environments. This description allows the user to:

- consult the list of ENDEVOR environments of the entity,

- define other possible ENDEVOR target environments.

The interface recognizes the first target environment whose validation indicator is set.

NOTE: The definition of the same environment on two different description lines can have unpredictable results.

The description of ENDEVOR entity target environments contains the following information:

. VALIDATION indicator (v or blank)

. ENVIRONMENT code

. STAGE code (display only)

. SYSTEM code

. SUB-SYSTEM code

. VA Pac ELEMENT TYPE

. VA Pac ELEMENT nature and code

. presence indicator of INFOPAC ELEMENT characteristics

The presence indicator of INFOPAC ELEMENT characteristics is automatically displayed if the VA Pac ELEMENT is present in this environment.

The Stage is not recognized if a target environment other than the environment defined by default is validated. It will be completed, for information, once the generated entity is taken into account in ENDEVOR.

--> VA Pac Choice: $7N eeeeee D1 (with eeeeee: UEO occurrence code).

```
--------------------------------------------------------------------------------
!                                                                              !
!                       DOCUMENTATION                       *PTXX.PDEV.BMS.651 !
! ENVIRONMENT                    7N PROGR1 1 COMMAND CHECK                     !
!                                                                              !
! A LIN : V ENVIRON. S SYSTEM   SUBSYST. TYPE     NAT.+ELEM *                  !
!   001 :   DEMO     P FINANCE  ACCTREC  PCOBOL   PPROGR1EX *                  !
!       : _ ________ _ ________ ________ ________ _________ _                  !
!       : _ ________ _ ________ ________ ________ _________ _                  !
!       : _ ________ _ ________ ________ ________ _________ _                  !
!       : _ ________ _ ________ ________ ________ _________ _                  !
!       : _ ________ _ ________ ________ ________ _________ _                  !
!       : _ ________ _ ________ ________ ________ _________ _                  !
!       : _ ________ _ ________ ________ ________ _________ _                  !
!       : _ ________ _ ________ ________ ________ _________ _                  !
!       : _ ________ _ ________ ________ ________ _________ _                  !
!       : _ ________ _ ________ ________ ________ _________ _                  !
!       : _ ________ _ ________ ________ ________ _________ _                  !
!       : _ ________ _ ________ ________ ________ _________ _                  !
!       : _ ________ _ ________ ________ ________ _________ _                  !
!       : _ ________ _ ________ ________ ________ _________ _                  !
!       : _ ________ _ ________ ________ ________ _________ _                  !
!       : _ ________ _ ________ ________ ________ _________ _                  !
!       : _ ________ _ ________ ________ ________ _________ _                  !
!                                                                              !
! O: C1 CH: $7NPROGR1 D1                                                       !
--------------------------------------------------------------------------------
```

DESCRIPTION OF THE LAST ACTION ON THE ENTITY OBJECTS

This description is updated upon Endevor return. So it is in read-only mode. It memorizes the characteristics related to the last action carried out on the entity's generated objects.

It comprises the following information:

```
. the code of the last action

. the date of the last action

. the time of the last action

. the user code who carried out the last action
```

To each description line of the Endevor target environment where the "presence indicator of the INFOPAC element characteristics" is displayed, corresponds a description line (with the same number) of the VA Pac entity in Endevor.

--> VA Pac Choice: $7N eeeeee D2 (with eeeeee: OEU occurrence code).

```
--------------------------------------------------------------------------------
!                                                                              !
!                       DOCUMENTATION                       *PTXX.PDEV.BMS.651 !
! Last Action                    7N PROGR1 2 COMMAND CHECK                     !
!                                                                              !
! A LIN : Action   Date       Time  User                                       !
!   001 : ADD      1999/10/23 11:12 PTXX                                       !
!       : ________ __________ _____ ________                                   !
!       : ________ __________ _____ ________                                   !
!       : ________ __________ _____ ________                                   !
!       : ________ __________ _____ ________                                   !
!       : ________ __________ _____ ________                                   !
!       : ________ __________ _____ ________                                   !
!       : ________ __________ _____ ________                                   !
!       : ________ __________ _____ ________                                   !
!       : ________ __________ _____ ________                                   !
!       : ________ __________ _____ ________                                   !
!       : ________ __________ _____ ________                                   !
!       : ________ __________ _____ ________                                   !
!       : ________ __________ _____ ________                                   !
!       : ________ __________ _____ ________                                   !
!       : ________ __________ _____ ________                                   !
!       : ________ __________ _____ ________                                   !
!       : ________ __________ _____ ________                                   !
! *** END ***                                                                  !
! O: C1 CH: $7Nprogr1 D2                                                       !
--------------------------------------------------------------------------------
```

DESCRIPTION OF THE VA Pac ENTITY IN ENDEVOR

Endevor automatically updates this description which is therefore in read-only mode. This description permits the storage of the characteristics of a VA Pac generated entity in ENDEVOR.

The description contains the following information:

. Update date in ENDEVOR

. Update time and Version in ENDEVOR

. Entity generation date

. Entity generation time

. VA Pac User Code

. Library code and generation session number

The library code is useful if the generation library is different from that in which the information about the ENDEVOR environments is stored.

For each description line of the ENDEVOR target environment where the "presence indicator of INFOPAC ELEMENT characteristics" is displayed, there is a VA Pac entity description line (with the same number) in Endevor.

--> VA Pac Choice: $7N eeeeee D9 (with eeeeee: UEO occurrence code).

```
--------------------------------------------------------------------------------
!                                                                              !
!                       DOCUMENTATION                       *PTXX.PDEV.BMS.651 !
! INFORMATION                    7N PROGR1 9 COMMAND CHECK                     !
!                                                                              !
! A LIN : NDV.DATE   NDV.T VV.LL GEN.DATE   GEN.TIME USER     LIB SESSI       !
!   001 : 1999/10/23 11:12 01.00 23/10/1999 11:10:21 PTXX     BMS 0730         !
!       : __________ ___________ __________ ________ ________ _________        !
!       : __________ ___________ __________ ________ ________ _________        !
!       : __________ ___________ __________ ________ ________ _________        !
!       : __________ ___________ __________ ________ ________ _________        !
!       : __________ ___________ __________ ________ ________ _________        !
!       : __________ ___________ __________ ________ ________ _________        !
!       : __________ ___________ __________ ________ ________ _________        !
!       : __________ ___________ __________ ________ ________ _________        !
!       : __________ ___________ __________ ________ ________ _________        !
!       : __________ ___________ __________ ________ ________ _________        !
!       : __________ ___________ __________ ________ ________ _________        !
!       : __________ ___________ __________ ________ ________ _________        !
!       : __________ ___________ __________ ________ ________ _________        !
!       : __________ ___________ __________ ________ ________ _________        !
!       : __________ ___________ __________ ________ ________ _________        !
!       : __________ ___________ __________ ________ ________ _________        !
!       : __________ ___________ __________ ________ ________ _________        !
! *** END ***                                                                  !
! O: C1 CH: $7NPROGR1 D9                                                       !
--------------------------------------------------------------------------------
```

2.8. PREDEFINED USER RELATIONSHIP

.NDVRL RELATIONSHIP

The .NDVRL predefined User Relationship which is supplied at installation creates links between the VA Pac entity and the occurrence of the .NDENV or .NDVLM User Entity.

The existence of these links at the entity level allows the user to know if the generated results are managed in ENDEVOR.

This relationship is set by the presence of the entity code in the description of the default environment when creating the occurrence of the .NDENV or .NDVLM User Entity.

The VA Pac entity must be defined in a library whose level is equal to or higher than that of the UEO, and in the same session (See "Target Libraries/Sessions Table").

```
--> VA Pac choice : P pppppp XQ (with pppppp: prog. code)
                    O oooooo XQ (with oooooo: screen code)
                    ...
```

2.9. INTERFACE PROCESSORS

INTERFACE PROCESSORS

Warning:

The processors supplied by the interface may be used without modification. However, the user may adapt them after having understood the functionalities described below. A bad adaptation of these processors can have unpredictable consequences.

INTRODUCTION

Any action on the VA Pac ELEMENT generally triggers a similar action on the INFOPAC ELEMENT and archives the transaction in the interface's journal file. These tasks are performed by the Interface's ENDEVOR PROCESSORS which are supplied at installation. The processors belong to Processor Groups related to the ACTION and to the TYPEs of VA Pac objects.

The interface has five processor types:

- MOVE PROCESSOR, activated by default by the MOVE action on VA Pac ELEMENTs. It can also be activated by the TRANSFER action on the same elements.

- DELETE PROCESSOR, activated by default by the DELETE action on VA Pac ELEMENTs. It can also be activated by the GENERATE action with the choice of a different Group of Processors, or by the MOVE/TRANSFER actions on option on the same elements.

- GENERATE PROCESSOR of source-type VA Pac ELEMENTs, activated by the ADD/GENERATE/TRANSFER/UPDATE actions on VA Pac ELEMENTs.

- GENERATE PROCESSOR of load-type VA Pac ELEMENTs, activated by the ADD/GENERATE/TRANSFER/UPDATE actions on VA Pac ELEMENTs.

- GENERATE PROCESSOR of INFOPAC ELEMENTs activated by the ADD/UPDATE actions on INFOPAC ELEMENTs.

1. MOVE PROCESSOR

The interface MOVE processor, supplied at installation in the PRCSMOVP member of the SY library, must be the MOVE processor of the processor groups which can be activated by the MOVE action on VA Pac ELEMENTs.

The MOVE PROCESSOR moves the INFOPAC ELEMENT into the same ENDEVOR environment as the VA Pac ELEMENT and at the same time. Also, the transaction is archived in the interface journal.

2. DELETE PROCESSOR

The interface DELETE processor, supplied in the PRCSDELP member of the SY library, must be the DELETE processor of the processor groups which can be activated by the DELETE action on VA Pac ELEMENTs.
The DELETE PROCESSOR deletes the INFOPAC ELEMENT in the same ENDEVOR environment as the VA Pac ELEMENT. Also, the transaction is archived in the interface journal.

3. GENERATE PROCESSOR OF SOURCE-TYPE VA Pac ELEMENTS

The interface GENERATE processor of source-type VA Pac ELEMENTs supplied in the PRCSGEPP member of the SY library, can be the GENERATE processor of the processor groups which can be activated by the ADD/GENERATE/TRANSFER/UPDATE actions on VA Pac ELEMENTs. The different steps of a processor in operation are conditioned by the TRANSFER or GENERATE action.

This processor moves the INFOPAC ELEMENT into the same Endevor environment as the VA Pac ELEMENT in the case of the TRANSFER action, and archives the transaction in the interface journal file.

4. GENERATE PROCESSOR OF LOAD-TYPE VA Pac ELEMENTS

The interface GENERATE processor of load-type INFOPAC ELEMENTs supplied in the PRCSGEPP member of the SY library must be the GENERATE processor of the processor groups which can be activated by the ADD/GENERATE/TRANSFER/UPDATE actions on the VA Pac ELEMENTs.

This processor compiles the VA Pac ELEMENT, moves the INFOPAC ELEMENT into the same Endevor environment as the VA Pac ELEMENT in the case of the TRANSFER action, and archives the transaction in the interface journal file.

5. GENERATE PROCESSOR OF INFOPAC ELEMENTS

The interface GENERATE processor of INFOPAC ELEMENTs supplied by the PRCSGENP member of the SY library, must be the GENERATE processor of the processor groups which can be activated by the ADD/UPDATE actions on the INFOPAC ELEMENTs.

This processor archives the transaction of the VA Pac ELEMENT in the interface journal file when being imported in Endevor.

6. OTHER PROCESSORS

The user can define several GENERATE processors which are activated by the ADD/UPDATE actions on VA Pac ELEMENTS. Each fills a different function related to the nature of the generated source. The user must carefully add the steps related to the processing of the corresponding INFOPAC ELEMENTs. These processors can be a call to pre-processors, a compilation, or a linkedit for example, and can be used when importing a VA Pac ELEMENT into ENDEVOR.

These processors are to be allocated to processor groups (see Chapter INSTALLATION, Subchapter ENVIRONMENT PREPARATION, Paragraph ALLOCATION OF PROCESSORS TO THE PROCESSOR GROUP).

In the definition of the processor groups related to the TYPEs of INFOPAC ELEMENTs, the MOVE action must use the MOVE processor, and the TRANSFER action the GENERATE processor.

In the definition of the processor groups related to the TYPEs of VA Pac ELEMENTs, the MOVE action must use the MOVE processor, and the TRANSFER action may use either the GENERATE processor or the MOVE processor.

In VA Pac, the choice of the processor group used to import a VA Pac ELEMENT into ENDEVOR must be indicated in the default target description of the .NDVLM User Entity occurrence or the .NDENV User Entity occurrence before the generation.

2.10. DESCRIPTION OF INTERFACE PROCESSORS

DESCRIPTION OF THE INTERFACE PROCESSORS

1.DESCRIPTION OF THE MOVE PROCESSOR

The MOVE processor of the processor groups which can be activated by the MOVE/TRANSFER action on the VA Pac ELEMENTs comprises the following steps:

- Allocation of message files: BC1PDSIN

- Preparation of the source context if TRANSFER action: PNTRAN

This step is executed only if the processor is activated by the TRANSFER action. It retrieves the parameters of the Endevor source context memorized by EXIT2 in the UP file in order to archive the action. The source context is not supplied in the symbolic parameter blocks during the TRANSFER action.

- Preparation of the source context if MOVE action: IEBGENER

Retrieval of the source context parameters from the Endevor symbolic parameters if the processor is activated by the MOVE action.

- Preparation of the action on the INFOPAC ELEMENT: PNPR10

With the parameters of the source context supplied by one of the two previous steps and those of the target context supplied by Endevor symbolic parameter blocks, the INFOPAC ELEMENT write commands are generated for the following step, as well as the action on the INFOPAC ELEMENT itself.

- Writing of the contents of the INFOPAC ELEMENT: CONWRITE

Execution of the commands generated by the PNPR10 step with the writing of the contents of the INFOPAC ELEMENT in a sequential file to prepare the action archiving.

- Memorization of the INFOPAC ELEMENT contents: PNPR11

Retrieval of the INFOPAC ELEMENT contents from the sequential file and memorization in the UP workfile which will be copied via EXIT3 in the UQ file, to archive the action.

- Execution of the action on the INFOPAC ELEMENT: C1BM300

Execution of the action on the INFOPAC ELEMENT with the commands prepared by the PNPR10 step.

- Printing of the messages from the various steps of the processor: CONLIST

2.DESCRIPTION OF THE DELETE PROCESSOR

The DELETE processor of the processor groups which can be activated by the DELETE/GENERATE/MOVE/TRANSFER actions on VA Pac ELEMENTS comprises the following steps:

- Allocation of message files: BC1PDSIN

- Preparation of the action on the INFOPAC ELEMENT: PNPR10

  Generation of the print commands of the INFOPAC ELEMENT for the next step, as well as the action on the INFOPAC ELEMENT itself.

- Writing of the INFOPAC ELEMENT contents: CONWRITE

  Execution of the commands generated by the PNPR10 step with the writing of the INFOPAC ELEMENT contents in a sequential file to prepare the action archiving.

- Memorization of the INFOPAC ELEMENT contents: PNPR11

  Retrieval of the INFOPAC ELEMENT contents from the sequential file and memorization in the UP workfile which will be copied into the UQ file, via EXIT3, for journal archiving.

- Execution of the action on the INFOPAC ELEMENT: C1BM300

  With the commands prepared by the PNPR10 step, execution of the action on the INFOPAC ELEMENT if the processor is not activated by the GENERATE action.

- Printing of the messages from the various steps of the processor: CONLIST

The various steps of the processor are executed only if the COMMENT field does not contain the '*IBM*' value. This makes it possible, in the execution of the 'Integrity Validation' procedure, not to delete the INFOPAC ELEMENT.

3.DESCRIPTION OF THE GENERATE PROCESSOR OF THE SOURCE TYPE VA Pac ELEMENTS

The GENERATE processor of the processor groups which can be activated by the ADD/UPDATE/GENERATE/TRANSFER actions on the Source-type VA Pac ELEMENTs is effective only for the TRANSFER or GENERATE action. The different steps of the processor are executed only if the processor is activated by the TRANSFER or GENERATE action. It comprises the following steps:

- Allocation of the message files: BC1PDSIN

- Preparation of the source context: PNTRAN

This step is executed only if the processor is activated by the GENERATE or TRANSFER action. The Endevor source context parameters memorized via EXIT2 in the UP file are retrieved in order to archive the action. The source context is not supplied in the symbolic parameter blocks during the TRANSFER action.

- Preparation of the action on the INFOPAC ELEMENT: PNPR10

This step is executed only if the processor is activated by the GENERATE or TRANSFER action. With the parameters of the source context supplied by the previous steps and those of the target context supplied by Endevor symbolic parameters, the action on the INFOPAC ELEMENT itself is generated.

- Writing of the INFOPAC ELEMENT contents: CONWRITE

This step is executed only if the processor is activated by the GENERATE action. Execution of the commands generated by the previous step with the writing of the contents of the INFOPAC ELEMENT in a sequential file to prepare for archiving. In the case of the TRANSFER action, there is no need to archive the action as this is done by the MOVE processor.

- Memorization of the INFOPAC ELEMENT contents: PNPR11

This step is executed only if the processor is activated by the GENERATE action. The INFOPAC ELEMENT contents are retrieved from the sequential file and the element is memorized in the UP workfile. This file will be copied, via EXIT3, in the UQ journal file.

- Execution of the action on the INFOPAC ELEMENT: C1BM300

This step is executed only if the processor is activated by the TRANSFER action. The action is executed on the INFOPAC ELEMENT with the commands prepared by the PNPR10 step.

- Pringing of the messages of the various steps of the processor: CONLIST

This step is executed only if the processor is activated by the GENERATE or TRANSFER action.

This processor cannot be activated by the MOVE action as the symbolic parameters of the source environment used in the processor are not supplied by Endevor.

4.DESCRIPTION OF THE GENERATE PROCESSOR OF LOAD TYPE VA Pac ELEMENTS

The GENERATE processor of the processor groups which can be activated by the ADD/UPDATE/GENERATE/MOVE/TRANSFER actions on the Load-type VA Pac ELEMENTs comprises the following steps:

- Compilation of the VA Pac ELEMENT: IKFCBL00

- Linkedit of the VA Pac ELEMENT: IEWL

- Allocation of the messages files: BC1PDSIN

  This step is executed only if the processor is activated by the GENERATE or TRANSFER action.

- Preparation of the source context: PNTRAN

  This step is executed only if the processor is activated by the GENERATE or TRANSFER action. It retrieves the Endevor source context parameters memorized via EXIT2 in the UP file in order to archive the action. The source context is not supplied in the symbolic parameter blocks during the TRANSFER action.

- Preparation of the action on the INFOPAC ELEMENT: PNPR10

  This step is executed only if the processor is activated by the GENERATE or TRANSFER action. With the parameters of the source context supplied by the previous steps and those of the target context supplied by Endevor symbolic parameters, the action on the INFOPAC ELEMENT itself is prepared.

- Writing of the INFOPAC ELEMENT contents: CONWRITE

  This step is executed only if the processor is activated by the GENERATE action. Execution of the commands generated by the previous step with the writing of the INFOPAC ELEMENT contents in a sequential file to prepare for archiving. In the case of the TRANSFER action, there is no need to archive the action as this is done by the MOVE processor.

- Memorization of the INFOPAC ELEMENT contents: PNPR11

This step is executed only if the processor is activated by the GENERATE action. The INFOPAC ELEMENT contents are retrieved from the sequential file and the element is memorized in the UP workfile. This file will be copied, via EXIT3, in the UQ journal file.

- Execution of the action on the INFOPAC ELEMENT: C1BM300

This step is executed only if the processor is activated by the TRANSFER action. The action is executed on the INFOPAC ELEMENT with the commands prepared by the PNPR10 step.

- Printing of the messages of the various steps of the processor: CONLIST

This step is executed only if the processor is activated by the GENERATE or TRANSFER action.

This processor cannot be activated by the MOVE action as the symbolic parameters of the source environment used in the processor are not supplied by Endevor.

5.DESCRIPTION OF THE GENERATE PROCESSOR OF THE INFOPAC ELEMENTS

The GENERATE processor of the processor groups which can be activated by the ADD/UPDATE actions on the INFOPAC ELEMENTS include the following steps:

- Allocation of the message files: BC1PDSIN

- Preparation of the source context: PNTRAN

  Retrieval of the ENDEVOR source context parameters memorized, via EXIT2, in the UP file, and deletion of the source context memorized in UP file thanks to the 'I' value in the last parameter of the step.

- Preparation of the action on INFOPAC ELEMENT: PNPR10

  With the parameters of the source context supplied by the preceding step and those of the target context supplied by Endevor symbolic parameter blocks, generation of the write commands of the INFOPAC ELEMENT for next step.

- Writing of the contents of the INFOPAC ELEMENT: CONWRITE

  Execution of the commands generated by the previous step with the writing of the INFOPAC ELEMENT contents in a sequential file to prepare for archiving.

- Memorization of the INFOPAC ELEMENT contents: PNPR11

  The contents of the INFOPAC ELEMENT are retrieved from the sequential file and the element is memorized in the UP workfile. This file will be copied, via EXIT3, in the UQ journal file.

- Printing of the messages from the various steps of the processor: CONLIST

The various steps of the processor are executed only if the COMMENT field does not contain the '*RND*' value. This makes it possible, for an action generated by the Interface during a retrieval of existing data, not to delete the INFOPAC ELEMENT.

2.11. EXITS OF THE INTERFACE

THE INTERFACE EXITS

The Interface has two exits: EXIT2 and EXIT3. They allow to control the different Endevor actions that are carried out on the VA Pac ELEMENTs or the INFOPAC ELEMENTs, and to finish the tasks started by the processors, such as the archiving of transactions. They are supplied respectively under the C1UEXT02 and C1UEXT03 members in the Interface load-modules library.

EXIT2 FUNCTIONS: C1UEXT02

Executed before the processing of the Endevor action, C1UXT02 controls the validity of all the actions performed on the VA Pac ELEMENTs recognized via the Types table.

It rejects all Endevor actions on INFOPAC ELEMENTs except the actions generated by the Interface procedures.

It rejects the ADD action on VA Pac ELEMENTs if it is not generated by the Interface procedures.

In the case of the TRANSFER action, it memorizes the source context of the element to be processed in the UP workfile in order to transmit it to the processors. This context is not supplied in the symbolic parameter blocks for archiving at the time of this action.

EXIT3 FUNCTIONS: C1UEXT03

Executed after the Endevor action is processed, C1UEXT03 retrieves the transaction corresponding to the Endevor action which is being memorized in the UP workfile by the different processors, deletes it from the UP file and records this transaction in the UQ Interface Journal file.

3. FUNCTIONALITIES

3.1. INTRODUCTION

INTRODUCTION

The VA Pac/ENDEVOR Interface permits:

- the automatic recognition and acceptance in ENDEVOR of VA Pac generated objects.

- the standard management of these objects in ENDEVOR.

- On-line consultation in VA Pac of the ENDEVOR environments in which the objects are managed.

- consultation in ENDEVOR of the VA Pac source environments of these objects.

The interface has two phases:

- Generation of VA Pac entities: import into ENDEVOR, and archiving of the transactions in the interface's journal.

- Impact of these transactions into VA Pac: formatting of the interface's journal in the form of batch update transactions and actual VA Pac batch update performed by the UPDT procedure.

3.2. GENERATION AND IMPORT OF A VA PAC ELEMENT

GENERATION AND IMPORT OF A VA Pac ELEMENT

VA Pac objects are imported or manipulated in ENDEVOR by ENDEVOR Actions. All transactions concerning these objects are archived in an interface journal file. These transactions are reformatted as VA Pac batch update transactions by the MEND procedure. The batch update brings back into VA Pac, the ENDEVOR environments where the generated entities are managed.

IMPORT INTO ENDEVOR

VA Pac objects are imported into ENDEVOR directly by a VA Pac GPRT generation procedure which produces a generated source. The source is preceded and followed by JCL lines which come from the VA Pac Before/After command options (see the VA Pac USER'S MANUAL).

The user must indicate the following JCL lines in the After program commands:

- the execution of the UPND procedure, followed by:

- the $PACPDS parameter containing the name of the temporary library in which the generated VA Pac ELEMENT is found as follows:

$PACPDS='-'

- the $PACINF parameter containing the contents of the INFOPAC ELEMENT as follows:

$PACINF=-

The GPND procedure uses the file resulting from the generation, completes it with ENDEVOR ADD actions before submitting it for execution using the parameters indicated above. The procedure imports two objects into ENDEVOR: the VA Pac ELEMENT and the INFOPAC ELEMENT.

The ADD action of the INFOPAC ELEMENT stores the VA Pac ELEMENT transaction in the Journal file of the Interface.

As a reminder, the following options are authorized for this Action:

- the OVERRIDE SIGNOUT option to import a new version of the VA Pac ELEMENT under a user code which is different from that assigned to the current version of the same element;

- the DELETE INPUT SOURCE option to delete the source object once it is imported into Endevor;

- the NEW VERSION option to assign a version number to the imported VA Pac ELEMENT;

- the name of the processor group to select a processor other than that defined by default at the element TYPE level;

- the CCID number;

- and the comments of the COMMENT option.

VA Pac Before/After Command Example

The following is an example of the use of the Before/After commands in a batch program (see the USER MANUAL, Chapter DATABASE MANAGEMENT; or the OPERATIONS MANUAL, Chapter CONTROL CARDS FOR GENERATED JCL for more detail on 'x' option types and the examples of 'y' option codes defined below).

.COMMANDS AT THE BEGINNING OF THE GENERATED PROGRAMS FLOW

They permit to allocate a temporary storage PDS before all generations.

```
         1         2         3         4        7    7    8
(x) (y) 8-0---------0---------0---------0----//--0----5----0

 A   E  //- JOB (-),-,CLASS=-,MSGCLASS=-,            12U34-
 A   E  // NOTIFY=-                                  U    -
 A   E  //*
 A   E  //*--- TEMPORARY PDS CREATION ---
 A   E  //*
 A   E  //STEP01   EXEC PGM=IEFBR14
 A   E  //DD1      DD   DSN=-,DISP=(,CATLG,DELETE),   5    -
 A   E  //             UNIT=3390,VOL=SER=STM480,
 A   E  //             DCB=(RECFM=FB,LRECL=80,
 A   E  //                  BLKSIZE=9440,DSORG=PO),
 A   E  //             SPACE=(TRK,(20,05,05),RLSE)
```

.GENERATED PROGRAM BEFORE COMMANDS

They permit to store the generated source in the temporary PDS allocated here above.

```
         1         2         3         4        7    7    8
(x) (y) 8-0---------0---------0---------0----//--0----5----0

 D   E  //*
 D   E  //*--- SOURCE BACKUP ---
 D   E  //*
 D   E  //STEP02   EXEC PGM=IEBUPDTE,PARM=NEW
 D   E  //SYSUT1   DD   DSN=-,DISP=SHR               1    -
 D   E  //SYSUT2   DD   DSN=-,DISP=SHR               1    -
 D   E  //SYSPRINT DD SYSOUT=*
 D   E  //SYSIN    DD *
 D   E  ./ADD NAME=-                                 P    -
```

.GENERATED PROGRAM AFTER COMMANDS

They permit to execute the import preparation procedure for generated objects in ENDEVOR UPND. The user must define the ACPDS parameter to indicate the name of the temporary PDS, and ACINF to indicate the contents of the INFOPAC ELEMENT. The value of the INFOPAC ELEMENT contents parameter is 'L'.

```
          1         2         3         4        7    7    8
(x) (y) 8-0---------0---------0---------0----//--0----5----0

 F   E  //*
 F   E  //*--- IMPORT INTO ENDEVOR ---
 F   E  //*
 F   E  //UPND     EXEC NDVRUPND
 F   E  //*
 F   E  $PACPDS='-'                                   1    -
 F   E  $PACINF=-                                     L    -
```

.COMMANDS AT THE END OF THE GENERATED PROGRAMS FLOW

They permit to delete the temporary storage PDS.

```
          1         2         3         4        7    7    8
(x) (y) 8-0---------0---------0---------0----//--0----5----0

 Z   E  //*
 Z   E  //*--- CLEAR TEMPORARY STORAGE ---
 Z   E  //*
 Z   E  //STEP03   EXEC PGM=IEHPROGM,COND=EVEN
 Z   E  //DD1      DD   UNIT=3390,VOL=SER=STM480,DISP=SHR
 Z   E  //SYSIN    DD   *
 Z   E   UNCATLG DSNAME=-                             5    -
 Z   E   SCRATCH DSNAME=-,VOL=3390=STM480             5    -
 Z   E  /*
 Z   E  //SYSPRINT DD   SYSOUT=*
 Z   E  //*
 Z   E  //
```

EXAMPLE: Job generation submit screens

1. Validated JCL lines for the execution of the generation procedure ('C4' operation code),

2. Corresponding generation command lines of the screen obtained with the 'GP' choice ('C1' operation code).

See the 'USER'S MANUAL', Chapter 'GENERATION PRINT' for more details on job generation submit screens.

GPRT (see the JCL lines below) initiates the generation of the VA Pac element.
GPND (see same JCL) submits a job which imports into Endevor the VA Pac element and the INFOPAC element.

```
!-------------------------------------------------------------------------------!
!                                                                               !
!                      DOCUMENTATION                    *PTXX.PDEV.BMS.651 !
! JCL LINES FOR THE COMMANDS                            USER: PTXX             !
!                                                                               !
! A    COM  LIGNE  :    V C CONTINUATION OF THE REQUEST                        !
!      JCL  000001 :    V   //PTXXGPRT JOB (111,1111,XX,CLASS=X,MSGCLASS=X,    !
!      JCL  000002 :    V   //      NOTIFY=PTXX                                !
!      JCL  000003 :    V   /*JOBPARM LINES=100                                !
!      JCL  000004 :    V   //GPRT    EXEC D280GPRT,ROOT=D2,FILE=80,OUT=*,     !
!      JCL  000005 :    V   //        INDUV='PT$PDV.PAC802',                   !
!      JCL  000006 :    V   //        COND.LNK=(00,LE,PAC)                     !
!      JCL  700000 :    V   //*                                                !
!      JCL  700010 :    V   //GPND    EXEC NDVRGPND,USER=PTXX,                 !
!      JCL  700020 :    V   //        PAC7GB=PAC7GB                            !
!      JCL  700030 :    V   //PNDV10.PAC7JB DD DSN=*.GPRT.PAC.PAC7GB,          !
!      JCL  700040 :    V   //        DISP=(OLD,PASS)                          !
!                  :                                                           !
!                  :                                                           !
!                  :                                                           !
!                  :                                                           !
!                  :                                                           !
!                  :                                                           !
!                  :                                                           !
! *** END ***                                                                  !
! O: C4 CH:                      JOB:                                          !
!                                                                               !
---------------------------------------------------------------------------------
```

```
!-------------------------------------------------------------------------------!
!                                                                               !
!                      DOCUMENTATION                    *PTXX.PDEV.BMS.651 !
! GENERATION AND PRINT COMMANDS                         USER: PTXX             !
!                                                                               !
! A SO COM  ENTITY : OP V C CONTINUATION OF THE REQUEST                        !
!   90 FLP         : C1 V * PROGRAM JOB CARD / JOB DELIM  ENV: _ (CCF:E CCB:E) !
!                  :        1=PTXXGPND_2=111,1111_3=X_4=X_5=PTXX.NDVR.TEMP     !
!   90 GCP  PTU001 : C1 V * SOURCE CODE FOR SELECTED PROGRAM     (CCF:E CCB:E) !
!                  :        1=PTXX.NDVR.TEMP                                   !
!   90 GCP  PTU002 : C1   * SOURCE CODE FOR SELECTED PROGRAM     (CCF:E CCB:E) !
!                  :        1=PTXX.NDVR.TEMP                                   !
!                  :                                                           !
!                  :                                                           !
!                  :                                                           !
!                  :                                                           !
!                  :                                                           !
!                  :                                                           !
!                  :                                                           !
!                  :                                                           !
!                  :                                                           !
!                  :                                                           !
!                  :                                                           !
! *** END ***                                                                  !
! O: C1 CH:                      JOB:                                          !
!                                                                               !
---------------------------------------------------------------------------------
```

3.3. MANAGEMENT OF A VA PAC ELEMENT IN ENDEVOR

MANAGEMENT OF A VA Pac ELEMENT IN ENDEVOR

TRANSFER OF VA Pac ELEMENT IN ENDEVOR

Once imported into ENDEVOR, only VA Pac ELEMENTs can be manipulated by ENDEVOR Actions. They can be transferred from one Stage to another with the Endevor 'MOVE' Action, or from one environment to another with the Endevor 'TRANSFER' Action. The interface accompanies the transfer of the VA Pac ELEMENT with the transfer of the corresponding INFOPAC ELEMENT to the same ENDEVOR target environment using the various MOVE or GENERATE processors of the interface.

The user can consult the contents of the INFOPAC ELEMENT, but cannot perform ENDEVOR Actions on it.

The 'MOVE' or 'TRANSFER' Action memorizes the VA Pac ELEMENT transaction in the interface's journal file.

The following options are authorized for the 'MOVE' action:

- the DELETE 'FROM' ELEMENT option to delete the VA Pac ELEMENT from the source Stage after the transfer. Only "Y", the option's default value, is valid. The element is deleted from the source Stage;

- the RETAIN SIGNOUT option to keep the user code assigned to the VA Pac ELEMENT in the target Stage. By default, the element loses the assignment after the transfer;

- the SIGNOUT TO option to assign a user code other than that being used by the VA Pac ELEMENT in the target Stage;

- the CCID number;

- the comments of the COMMENT option.

The following options are authorized for the 'TRANSFER' action:

- the DELETE 'FROM' ELEMENT option to delete the VA Pac ELEMENT from the source Stage after the transfer. Only "Y", the option's default value, is valid. The element is deleted from the source Stage;

- the GENERATE ELEMENT option to regenerate or recompile the VA Pac ELEMENT after transfer;

- the OVERRIDE SIGNOUT option to allow the user to transfer the VA Pac ELEMENT carrying a user code different from her/his own;

- the PROCESSOR GROUP option to associate a group of processors to the VA Pac ELEMENT to be transferred;

- the WITH HISTORY option;

- the RETAIN SIGNOUT option to keep the user code assigned to the VA Pac ELEMENT in the target Stage. By default the element loses the assignment after the transfer;

- the SIGNOUT TO option to assign to the VA Pac ELEMENT of the target Stage a user code different from that of the source Stage;

- the CCID number;

- and the comments of the COMMENT option.

DELETING VA Pac ELEMENTS IN ENDEVOR

The VA Pac ELEMENT can be deleted from the ENDEVOR environment with the 'DELETE' Action. The interface accompanies the deletion of the VA Pac ELEMENT with the deletion of the corresponding INFOPAC ELEMENT using the 'PACD' processor.

The 'DELETE' Action stores the VA Pac ELEMENT transaction in the interface's journal file.

The following options are authorized for this Action:

- the OVERRIDE SIGNOUT option to allow the user to delete the VA Pac ELEMENT assigned to a user code different from the one in use;

- the CCID number;

- and the comments of the COMMENT option.

3.4. UPDATE OF THE ENDEVOR CONTEXT IN VA PAC

UPDATE OF THE ENDEVOR CONTEXT IN VA Pac

The VA Pac ELEMENT transactions in ENDEVOR by the ADD/UPDATE, DELETE, GENREATE, MOVE and TRANSFER Actions are archived in the interface's journal file.

To return the ENDEVOR information to VA Pac, the administrator must prepare the VA Pac batch update transactions based on the journal file using the MEND procedure, followed by the VA Pac update procedure, UPDT.

The ENDEVOR environments of the generated VA Pac objects are memorized in the description of the .NDVML User Entity Occurrence. The occurrence has the same code as the generated entity.

The choice of the library and session to save the ENDEVOR information can be parameterized in the TARGET LIBRARY/SESSION TABLE. By default, the ENDEVOR contexts are saved in the same library and the same session as that used for generation.

IMPORTANT NOTE: The VA Pac Database must be closed.

4. INTEGRITY VALIDATION

4.1. INTRODUCTION

INTRODUCTION

Integrity validation enables you to test the validity of the data related to the ENDEVOR environments where the VA Pac and INFOPAC ELEMENTS are managed, and those recorded in VA Pac as .NDVLM User Entity Occurrences.

There are two types of integrity validation:

- Intra-ENDEVOR integrity validation,

- Inter-Environment integrity validation.

4.2. INTRA-ENDEVOR INTEGRITY VALIDATION

INTRA-ENDEVOR INTEGRITY VALIDATION

In the same ENDEVOR environment, a VA Pac ELEMENT must be accompanied by an INFOPAC ELEMENT having the same code. This validation checks the presence of these two elements in the same ENDEVOR context.

The Intra-ENDEVOR integrity validation procedure produces an error report, and generates a transaction file of the deleted VA Pac ELEMENTS or INFOPAC ELEMENTS that are wrongly present.

REMINDER: The VA Pac and INFOPAC ELEMENTS can only be deleted using transactions generated by the interface.

4.3. INTER-ENVIRONMENT INTEGRITY VALIDATION

INTER-ENVIRONMENT INTEGRITY VALIDATION

The administrator updates

the Endevor data related to the contexts where VA Pac generated objects are managed. The transfer of VA Pac ELEMENTS in ENDEVOR and the update of the .NDVLM User Entity Occurrences are performed in two distinct phases.

For this reason, the inter-environment integrity validation checks the validity of the ENDEVOR data stored in VA Pac against the real contexts where the VA Pac ELEMENTS are generated in ENDEVOR.

This procedure produces an error report and proposes corrective transactions for the invalid data stored in the .NDVLM User Entity Occurrences.

These corrections are VA Pac batch update transactions used by the UPDT procedure.

5. EXISTING DATA RETRIEVAL FACILITY

RETRIEVAL FACILITY

For sites which operated with both VA Pac and ENDEVOR without the interface described in this manual, it is possible to use the Retrieval Facility.

This facility recognizes VA Pac ELEMENTs (batch or on-line programs) managed in ENDEVOR and creates:

- VA Pac batch update transactions which contain the ENDEVOR data related to these elements. These transactions are to be taken into account in VA Pac by the UPDT procedure,

- INFOPAC ELEMENTs creation transactions related to these VA Pac ELEMENTs in ENDEVOR. They are contained in a batch job constituted of ENDEVOR ADD actions which the user must complete before submission.

The user must supply the list of ENDEVOR environments where the generated objects coming from VA Pac are managed. The following procedures must then be executed in order:

- Search of the VA Pac ELEMENTs managed in ENDEVOR and preparation of the creation job of the corresponding INFOPAC ELEMENTs (RRND procedure),

- Update and submission of the job mentioned in the preceding step in order to actually create the INFOPAC ELEMENTs in ENDEVOR (RIND procedure),

- Generation of VA Pac update transactions in order to create the .NDVLM user entity occurrences which contain data about the ENDEVOR context of these VA Pac ELEMENTs (RPND procedure).

NOTE: The source of the VA Pac objects managed in ENDEVOR must be available. The retrieval procedure detects the source objects coming from VA Pac thanks to the presence of the 'PACBASE-CONSTANTS' string.

The VA Pac TYPE Table and the TARGET LIBRARY/SESSION Table must be updated beforehand.

6. RETRIEVAL 1.6 ---> 2.5

INTERFACE RETRIEVAL 1.6 -> 2.5

The 2.5 release of the interface ensures the Year 2000 compatibility with Endevor 3.7.2 (including the Year2000 PTF) and 3.8 releases.

After the product is installed, two retrievals must be performed:

1. Retrieval of the 'QU' archived journal (sse the JJND procedure).

2. Retrieval of Endevor User Entity Occurrences in Va Pac.

The 2.5 release takes into account the year 2000 roll over for the user occurrences managed by Endevor.

So the 'REND' procedure recognizes all the $7N occurrences of the .NDVLM user entity from the 1.6 release and applies the new date format to them:

1. last action CCYY/MM/DD instead of YY/MM/DD

2. current action CCYY/MM/DD instead of YY/MM/DD

3. element generation DD/MM/CCYY instead of DD/MM/YY

Note: All libraries and sessions are updated.

Before executing this procedure:

1. upon installation, you must execute the '$PRFJ.VINS' job under the ===MOD NDVRVINS module name. This job updates the new descriptions of the Endevor User Entities, release Year2000.

2. it is highly recommended to save the VA Pac database.

For more detail, see the 'REND' procedure.

7. THE BATCH PROCEDURES

7.1. INTRODUCTION

INTRODUCTION

BATCH processing is divided into various procedures. The following sub-chapters present each of the procedures with their specific execution conditions.

The description of each procedure includes:

. A general presentation with:

- an introduction
- the execution conditions
- the actions to be taken in case of abnormal execution.

. A description of user input, processing executed, and results, plus specific recommendations.

. A description of steps with:

- symbolics and parameters,
- list of the files used (temporary and permanent),
- JCL lines.

PROCEDURE CLASSIFICATION

The procedures associated with batch operations are classified into various categories:

IMPORT PROCEDURES OF ELEMENTS GENERATED IN ENDEVOR

. Retrieval of the result of VA Pac generation-print procedure (GPRT), import preparation of the elements generated in ENDEVOR (GPND) and actual import in ENDEVOR (UPND).

GENERATION PROCEDURES OF VA Pac UPDATE TRANSACTIONS

. Formatting of VA Pac batch update transactions, archiving and re-initialization of UQ journal (MEND).

UTILITY PROCEDURES

. Updating of the element TYPES table managed by the interface (TYND).

. Updating of TARGET LIBRARY/SESSION table (LSND).

. List of the interface program dates (INND)

. Retrieval of the journal transactions already archived (JRND).

INTEGRITY VALIDATION PROCEDURES

. Intra-ENDEVOR integrity validation (CEND).

. Inter-Environment integrity validation (CIND).

RETRIEVAL PROCEDURES OF PREVIOUS RELEASES

. Retrieval of the 8.0.2 V02I archived journal file (JJND).

. Retrieval of TS and TY files from 1.6 release (RPTS).

7.2. CEND: INTRA-ENDEVOR INTEGRITY VALIDATION

CEND: INTRA-ENDEVOR INTEGRITY VALIDATION

The CEND procedure validates the consistency and existence of INFOPAC ELEMENTs and the corresponding VA Pac ELEMENTs in the same ENDEVOR environment. Each VA Pac ELEMENT must have an INFOPAC ELEMENT which includes the VA Pac generation context.

EXECUTION CONDITION

The interface must not be in use.

USER INPUT

. One identification line per environment.

ENVIRONMENT PARAMETER LINE

Pos.	Length	Value	Meaning
11	8	eeeeeeee	ENDEVOR environment

REPORT PRINTING

This procedure prints a report of the Intra-ENDEVOR integrity validation showing detected anomalies.

```
DESCRIPTION OF STEPS

This procedure comprises four steps:

 . Recognition of user input                  : PTU001

 . Preparation of Endevor 'LIST' requests of
   VA Pac and INFOPAC ELEMENTs                : PNCI10

 . Listing of VA Pac ELEMENTs and INFOPAC
   ELEMENTs                                   : NDVRC1

 . Validation, printing of anomalies and generation
   of update requests for Endevor             : PNCI20



RECOGNITION OF USER INPUT                     : PTU001

. List of environments                        : CARTE
  DSN=SYSIN

. Output file                                 : PAC7MB
  DSN=&&CENDMB

PREPARATION OF ENDEVOR LISTING REQUESTS       : PNCI10

. Output files
 - Work file                                  : PAC7IN
   DSN=&&CENDMB
 - Interface TYPES file                       : PAC7TY
   DSN=&INDEX..&ROOT.&FILE.TY

. Output file:
 - Endevor LIST requests for VA Pac and INFOPAC
   ELEMENTs                                   : PAC7BS
   DSN=&&PAC7BS


LISTING OF VA Pac and INFOPAC ELEMENTs       : NDVRC1

. Input file
 - Interface TYPES file                       : PAC7TY
   DSN=&INDEX..&ROOT.&FILE.TY

. Input-output files
 - Interface work file                        : PAC7PU
   DSN=&INDEX..&ROOT.&FILE.UP
 - Interface work file                        : PAC7UP
   DSN=&INDEX..&ROOT.&FILE.UP

. Output files:
 - Printing of Endevor 'LIST' requests        : C1MSGS1
   DSN=&&C1MSGS1
```

```
INTRA-ENDEVOR INTEGRITY VALIDATION            : PNCI20

. Input files:
 - Printing of Endevor 'LIST' requests        : PAC7MS
   DSN=&&C1MSGS1
 - Interface TYPES file                       : PAC7TY
   DSN=&INDEX..&ROOT.&FILE.TY
 - VA Pac error messages file                 : PAC7AE
   DSN=&INDEXX..&ROOTX.&ROOTX.AE

. Output files:
 - Endevor update requests                    : PAC7BS
   DSN=&&PAC7BT
 - Integrity validation report                : PAC7CR
   SYSOUT=&OUT
```

```
//*********************************************************************
//* VA PAC-ENDEVOR 2.5   : INTRA-ENDEVOR INTEGRITY VALIDATION         *
//*********************************************************************
//$RADP.CEND PROC FILE=$FILE,    NUMBER OF PHYSICAL DATABASE
//    ROOT=$ROOT,                ROOT OF THE SYSTEM
//    ROOTX=$ROOTX,              ROOT OF 'VA PAC' SYSTEM
//    INDEX='$INDEX',            INDEX OF VSAM SYSTEM FILES
//    INDEXP='$INDEXP',          INDEX OF NON VSAM SYSTEM FILES
//    INDEXX='$INDEXX',          INDEX OF 'VA PAC' VSAM SYSTEM FILES
//*:  VSAMCAT='$CATU',           USER VSAM CATALOG
//*:  SYSTCAT='$CATV',           SYSTEM VSAM CATALOG
//    STEPLIB='$MODB',           LIBRARY OF LOAD-MODULES
//    LOADLIB='$LDLIB',          ENDEVOR LOADLIB
//    CONLIB='$CONLIB',          ENDEVOR CONLIB
//    SORTLIB='$BIBT',           SORT LIBRARY
//    CYL=1,                     SORT WORKS SIZE
//    OUT=$OUT,                  OUTPUT CLASS
//    UWK=$UWK,                  WORK UNIT
//    SPABS='(TRK,(02,01),RLSE)',  SPACE TRANSACTIONS FOR 'BSTIPT01'
//    SPAMS='(TRK,(05,05),RLSE)',  SPACE 'C1MSGS1 & C1PRINT' NDVRC1
//    PSBLIB='$PSBLIB',     PSB LIBRARY
//    DBDLIB='$DBDLIB',     DBD LIBRARY
//    RESLIB='$RESLIB',     RESLIB IMS
//    PROCLIB='$PRCLIB',    PROCLIB IMS
// BUF=40,SPIE=0,TEST=0,EXCPVR=0,RST=0,PRLD=,SRCH=0,
// CKPTID=,MON=N,LOGA=0,FMTO=T,DBRC=$DBRC,IRLM=$IRLM
//*------------------------------------------------------------------*
//*
//INPUT   EXEC PGM=PTU001
//STEPLIB  DD DSN=&STEPLIB,DISP=SHR
//CARTE    DD DDNAME=SYSIN,DCB=BLKSIZE=80
//PAC7MB   DD DSN=&&CENDMB,DISP=(,PASS),
//            UNIT=&UWK,SPACE=(TRK,(1,1),RLSE),
//            DCB=(RECFM=FB,LRECL=80,BLKSIZE=80)
//*
//PNCI10  EXEC PGM=PNCI10,PARM=' '
//STEPLIB  DD DSN=&STEPLIB,DISP=SHR
//*:STEPCAT  DD DSN=&VSAMCAT,DISP=SHR
//PAC7TY   DD DSN=&INDEX..&ROOT.&FILE.TY,DISP=SHR
//PAC7IN   DD DSN=&&CENDMB,DISP=(OLD,PASS)
//PAC7BS   DD DSN=&&PAC7BS,DISP=(,PASS),UNIT=&UWK,
//            DCB=(RECFM=FB,LRECL=80,BLKSIZE=6080),
//            SPACE=&SPABS
//SYSOUT   DD SYSOUT=&OUT
//SYSOUX   DD SYSOUT=&OUT
//SYSUDUMP DD SYSOUT=&OUT
//*
//NDVRC1  EXEC PGM=NDVRC1,DYNAMNBR=1500,REGION=4096K,PARM='C1BM3000'
//STEPLIB  DD DSN=&LOADLIB,DISP=SHR
//CONLIB   DD DSN=&CONLIB,DISP=SHR
//SYSPRINT DD SYSOUT=&OUT
//SYSUDUMP DD SYSOUT=&OUT
//SORTWK01 DD UNIT=SYSDA,SPACE=(CYL,(2,1))
//SORTWK02 DD UNIT=SYSDA,SPACE=(CYL,(2,1))
//SORTWK03 DD UNIT=SYSDA,SPACE=(CYL,(2,1))
//SORTWK04 DD UNIT=SYSDA,SPACE=(CYL,(2,1))
//C1TPDD01 DD UNIT=SYSDA,SPACE=(CYL,5),
//            DCB=(RECFM=VB,LRECL=260,BLKSIZE=6160)
//C1TPDD02 DD UNIT=SYSDA,SPACE=(CYL,5),
//            DCB=(RECFM=VB,LRECL=260,BLKSIZE=6160)
//C1TPLSIN DD UNIT=SYSDA,SPACE=(CYL,5),
//            DCB=(RECFM=FB,LRECL=80,BLKSIZE=6160)
//C1TPLSOU DD UNIT=SYSDA,SPACE=(CYL,5)
//PAC7TY   DD DSN=&INDEX..&ROOT.&FILE.TY,DISP=SHR
```

```
//PAC7PU   DD DSN=&INDEX..&ROOT.&FILE.UP,DISP=SHR
//PAC7UP   DD DSN=&INDEX..&ROOT.&FILE.UP,DISP=SHR
//PAC7UQ   DD DSN=&INDEX..&ROOT.&FILE.UQ,DISP=SHR
//C1PLMSGS DD SYSOUT=&OUT
//*1MSGS1  DD SYSOUT=&OUT
//C1MSGS1  DD DSN=&&C1MSGS1,DISP=(,PASS),UNIT=&UWK,
//            DCB=(RECFM=FBA,LRECL=133,BLKSIZE=26600),
//            SPACE=&SPAMS
//C1PRINT  DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,BLKSIZE=6171)
//SYSABEND DD SYSOUT=&OUT
//SYSOUT   DD SYSOUT=&OUT
//BSTIPT01 DD DSN=&&PAC7BS,DISP=(OLD,PASS)
//*
//PNCI20  EXEC PGM=DFSRRC00,REGION=$REGSIZ,
//        PARM=(DLI,PNCI20,PNCI20$SUG,&BUF,
//        &SPIE&TEST&EXCPVR&RST,&PRLD,
//        &SRCH,&CKPTID,&MON,&LOGA,&FMTO,,,&DBRC,&IRLM)
//STEPLIB  DD DSN=&RESLIB,DISP=SHR
//         DD DSN=&STEPLIB,DISP=SHR
//DFSRESLB DD DSN=&RESLIB,DISP=SHR
//IMS      DD DSN=&PSBLIB,DISP=SHR
//         DD DSN=&DBDLIB,DISP=SHR
//DDSNAP   DD SYSOUT=&OUT
//PROCLIB  DD DSN=&PROCLIB,DISP=SHR
//IEFRDER  DD DUMMY,
//            DCB=(RECFM=VB,BLKSIZE=1920,LRECL=1916,BUFNO=2)
//SYSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//            BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//            BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSMON   DD DUMMY
//DFSVSAMP DD DSN=&INDEXP..&ROOT.&ROOT.SY(DFSVSAM8),DISP=SHR
//SORTLIB  DD DSN=&SORTLIB,DISP=SHR
//SORTWK01 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//SORTWK02 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//SORTWK03 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//*:STEPCAT  DD DSN=&VSAMCAT,DISP=SHR
//PAC7AE$SUF DD DSN=&INDEXX..&ROOTX.&ROOTX.AE,DISP=SHR
//PAC7TY   DD DSN=&INDEX..&ROOT.&FILE.TY,DISP=SHR
//PAC7MS   DD DSN=&&C1MSGS1,DISP=(OLD,PASS)
//PAC7BS   DD DSN=&&PAC7BT,DISP=(,PASS),UNIT=&UWK,
//            DCB=(RECFM=FB,LRECL=80,BLKSIZE=6080),
//            SPACE=&SPABS
//PAC7CR   DD SYSOUT=&OUT
//SYSOUT   DD SYSOUT=&OUT
//SYSOUX   DD SYSOUT=&OUT
//SYSUDUMP DD SYSOUT=&OUT
//SYSPRINT DD SYSOUT=&OUT
//*
```

7.3. CIND: INTER-ENVIRONMENT INTEGRITY VALIDATION

INTER-ENVIRONMENT INTEGRITY VALIDATION

The CIND procedure validates the integrity of ENDEVOR data stored in VA Pac and the actual presence of VA Pac ELEMENTs in these environments. This procedure performs appropriate VA Pac update transactions.

EXECUTION CONDITION

The interface must not be in use.

USER INPUT

FIRST INPUT FLOW

. One administrator identification line.

USER PARAMETER LINE

Pos.	Length	Value	Meaning
2	1	'*'	Line code
3	8	uuuuuuuu	User code
11	8	mmmmmmmm	Password

SECOND INPUT FLOW

. One identification line per environment.

ENVIRONMENT PARAMETER LINE

Pos.	Length	Value	Meaning
11	8	eeeeeeee	ENDEVOR environment

REPORT PRINTING

This procedure prints an inter-environment integrity validation report.

RESULT

The procedure generates a VA Pac UPDT batch update transactions file to update the ENDEVOR information stored in VA Pac.

DESCRIPTION OF STEPS

This procedure comprises ten steps:

. Recognition of the administrator identification : PTU001
. Recognition of the list of Endevor environments : PTU001
. Extraction of $7N call-type occurrences : PNDC10
. Preparation of Endevor listing requests : PNDC50
. Listing of VA Pac ELEMENTs in Endevor : NDVRC1
. Formatting of the Endevor print-out obtained : PNDC60
. Listing of VA Pac ELEMENTs in Endevor : NDVRC1
. Retrieval of the date of the last action : PNDC70
. Comparison between VA Pac and Endevor environments and generation of possible corrective transactions : PNDC90
. Formatting of the VA Pac update transactions resulting from the comparison : PNDM10

RECOGNITION OF THE ADMINISTRATOR CODE: PTU001

. Input data : CARTE
DSN=SYSIN

. Output file : PAC7MB
DSN=&&CINDMB1

RECONITION OF THE LIST OF ENVIRONMENTS: PTU001

. Input data : CARTE
DSN=SYSIN

. Output file : PAC7MB
DSN=&&CINDMB2

```
EXTRACTION OF $7N CALL-TYPE OCCURRENCES: PNDC10

. Input files:
 - Administrator identification              : PAC7CA
   DSN=&&CINDMB1
 - VA Pac error messages file               : PAC7AE
   DSN=&INDEXX..&ROOTX.&ROOTX.AE
 - VA Pac index file                        : PAC7AN
   DSN=&INDEXX..&ROOTX.&FILEX.AN
 - VA Pac data file                         : PAC7AR
   DSN=&INDEXX..&ROOTX.&FILEX.AR
 - LIBRARIES and SESSIONS file               : PAC7TS
   DSN=&INDEX..&ROOT.&FILE.TS

. Output files:
 - Extracted transactions file               : PAC7RT
   DSN=&&PAC7RT
 - Reports of detected errors                : PAC7ET
   SYSOUT=&OUT


PREPARATION OF VA Pac ELEMENTS LISTING REQUESTS: PNDC50

. Input files
 - List of Endevor environments              : PAC7IN
   DSN=&&CINDMB2
 - Interface TYPES file                      : PAC7TY
   DSN=&INDEX..&ROOT.&FILE.TY

. Output file
 - Endevor 'LIST' requests of INFOPAC ELEMENTs: PAC7BS
   DSN=&&PAC7BS
 - Endevor 'LIST' requests of VA Pac ELEMENTs : PAC7BT
   DSN=&&PAC7BT
```

LISTING OF VA Pac ELEMENTS IN ENDEVOR: NDVRC1

```
. Input file
 - Interface TYPES file                       : PAC7TY
   DSN=&INDEX..&ROOT.&FILE.TY
 - Endevor 'LIST' requests of VA Pac ELEMENTs : BSTIPT01
   DSN=&&PAC7BS

. Input-Output files
 - Interface work file                        : PAC7PU
   DSN=&INDEX..&ROOT.&FILE.UP
 - Interface work file                        : PAC7UP
   DSN=&INDEX..&ROOT.&FILE.UP

. Output files
 - 'C1MSGS1' print file                       : C1MSGS1
   DSN=&&C1MSGS1
 - 'C1PRINT' print file                       : C1PRINT
   DSN=&&C1PRINT
```

FORMATTING OF THE PRINTOUTS OUTPUT BY NDVRC1: PNDC60

```
. Input files
 - Interface TYPES file                       : PAC7TY
   DSN=&INDEX..&ROOT.&FILE.TY
 - 'C1PRINT' output file from NDVRC1          : PAC7LT
   DSN=&&C1PRINT
 - 'C1MSGS1' output file from NDVRC1          : PAC7PT
   DSN=&&C1MSGS1

. Output file
 - Formtted VA Pac ELEMENTs file              : PAC7RS
   DSN=&&PAC7RR
```

LISTING OF VA Pac ELEMENTS IN ENDEVOR: NDVRC1

```
. Input file
 - Interface TYPES file                    : PAC7TY
   DSN=&INDEX..&ROOT.&FILE.TY
 - Endevor 'LIST' requests of VA Pac ELEMENTs: BSTIPT01
   DSN=&&PAC7BS

. Input-Output files
 - Interface work file                     : PAC7PU
   DSN=&INDEX..&ROOT.&FILE.UP
 - Interface work file                     : PAC7UP
   DSN=&INDEX..&ROOT.&FILE.UP

. Output files
 - 'C1MSGS1' print file                    : C1MSGS1
   DSN=&&C1MSGS1
 - 'C1PRINT' print file                    : C1PRINT
   DSN=&&C1PRINT
```

FORMATTING OF THE PRINTOUTS OUTPUT BY NDVRC1: PNDC70

```
. Input files
 - Interface TYPES file                    : PAC7TY
   DSN=&INDEX..&ROOT.&ROOT.TY
 - 'C1MSGS1' output file from NDVRC1       : PAC7LT
   DSN=&&C1MSGS1P
 - 'C1PRINT' output file from NDVRC1       : PAC7PT
   DSN=&&C1PRINTP
 - INFOPAC elements file                   : PAC7RR
   DSN=&&PAC7RR

. Output file
 - Completed INFOPAC ELEMENTs file         : PAC7RS
   DSN=&&PAC7RS
```

```
INTER-ENVIRONNEMENT VALIDATION: PNDC90

. Input files
 - Administrator identification transaction : PAC7CA
   DSN=&&CINDMB1
 - VA Pac error messages file               : PAC7AE
   DSN=&INDEXX..&ROOTX.&ROOTX.AE
 - LIBRARIES AND SESSIONS file              : PAC7TS
   DSN=&INDEX..&ROOT.&FILE.TS
 - List of VA Pac ELEMENTs in Endevor       : PAC7UN
   DSN=&&PAC7RT
 - VA Pac data file                         : PAC7AR
   DSN=&INDEXX..&ROOTX.&FILEX.AR
 - List of occurrences stored in VA Pac     : PAC7UM
   DSN=&&PAC7RS

. Output files
 - Corrective transactions file             : PAC7UR
   DSN=&&PAC7UR
 - Report of detected errors                : PAC7ET
   SYSOUT=&OUT


FORMATTING OF VA Pac BATCH UPDATE TRANSACTIONS: PNDM10

. Input files:
 - Corrective transactions file             : PAC7UR
   DSN=&&PAC7UR
 - VA Pac index file                        : PAC7AN
   DSN=&INDEXX..&ROOTX.&FILEX.AN
 - VA Pac data file                         : PAC7AR
   DSN=&INDEXX..&ROOTX.&FILEX.AR
 - VA Pac error messages file               : PAC7AE
   DSN=&INDEXX..&ROOTX.&ROOTX.AE

. Output files:
 - VA Pac update transactions file          : PAC7MV
   DSN=&&PAC7MV
 - Validation report                        : PAC7ET
   DSN=&OUT
```

```
//*********************************************************************
//* VA PAC-ENDEVOR 2.5   : INTER-ENVIRONMENT INTEGRITY VALIDATION     *
//*********************************************************************
//$RADP.CIND PROC FILE=$FILE,         NUMBER OF PHYSICAL DATABASE
//         FILEX=$FILEX,              NUMBER OF VA PAC PHYSICAL DBASE
//         ROOT=$ROOT,                ROOT OF THE SYSTEM
//         ROOTX=$ROOTX,              ROOT OF 'VA PAC' SYSTEM
//         INDEX='$INDEX',            INDEX OF VSAM SYSTEM FILES
//         INDEXX='$INDEXX',          INDEX OF VA PAC VSAM SYSTEMFILES
//         INDEXP='$INDEXP',          INDEX OF NON VSAM SYSTEM FILES
//*:       VSAMCAT='$CATU',           USER VSAM CATALOG
//*:       SYSTCAT='$CATV',           SYSTEM VSAM CATALOG
//         STEPLIB='$MODB',           LIBRARY OF LOAD-MODULES
//         LOADLIB='$LDLIB',          ENDEVOR LOADLIB
//         CONLIB='$CONLIB',          ENDEVOR CONLIB
//         SORTLIB='$BIBT',           SORT LIBRARY
//         BCOB='$BCOB',              COBOL/VS ROUTINES LIBRARY
//         CYL=1,                     SORT WORKS SIZE
//         OUT=$OUT,                  OUTPUT CLASS
//         UWK=$UWK,                  WORK UNIT
//         SPAMB='(TRK,(02,01),RLSE)', TRANSACTIONS SPACE FOR BSTIPT01
//         SPAMS='(TRK,(05,05),RLSE)', SPACE 'C1MSGS1 & C1PRINT'
//         SPART='(TRK,(02,01),RLSE)', TEMPORARY FILE SPACE
//         SPAMV='(TRK,(05,01),RLSE)', SPACE OF UPDATE TRANSACTION FILE
//    PSBLIB='$PSBLIB',     PSB LIBRARY
//    DBDLIB='$DBDLIB',     DBD LIBRARY
//    RESLIB='$RESLIB',     RESLIB IMS
//    PROCLIB='$PRCLIB',    PROCLIB IMS
// BUF=40,SPIE=0,TEST=0,EXCPVR=0,RST=0,PRLD=,SRCH=0,
// CKPTID=,MON=N,LOGA=0,FMTO=T,DBRC=$DBRC,IRLM=$IRLM
//*-------------------------------------------------------------------*
//*
//INPUT1  EXEC PGM=PTU001
//STEPLIB  DD DSN=&STEPLIB,DISP=SHR
//CARTE    DD DDNAME=SYSIN,DCB=BLKSIZE=80
//PAC7MB   DD DSN=&&CINDMB1,DISP=(,PASS),
//            UNIT=&UWK,SPACE=(TRK,(1,1),RLSE),
//            DCB=(RECFM=FB,LRECL=80,BLKSIZE=80)
//*
//INPUT2  EXEC PGM=PTU001
//STEPLIB  DD DSN=&STEPLIB,DISP=SHR
//CARTE    DD DDNAME=SYSIN,DCB=BLKSIZE=80
//PAC7MB   DD DSN=&&CINDMB2,DISP=(,PASS),
//            UNIT=&UWK,SPACE=(TRK,(1,1),RLSE),
//            DCB=(RECFM=FB,LRECL=80,BLKSIZE=80)
//*
//PNDC10  EXEC PGM=DFSRRC00,REGION=$REGSIZ,
//        PARM=(DLI,PNDC10,PNDC10$SUG,&BUF,
//        &SPIE&TEST&EXCPVR&RST,&PRLD,
//        &SRCH,&CKPTID,&MON,&LOGA,&FMTO,,,&DBRC,&IRLM)
//STEPLIB  DD DSN=&RESLIB,DISP=SHR
//         DD DSN=&STEPLIB,DISP=SHR
//         DD DSN=&BCOB,DISP=SHR
//DFSRESLB DD DSN=&RESLIB,DISP=SHR
//IMS      DD DSN=&PSBLIB,DISP=SHR
//         DD DSN=&DBDLIB,DISP=SHR
//DDSNAP   DD SYSOUT=&OUT
//PROCLIB  DD DSN=&PROCLIB,DISP=SHR
//IEFRDER  DD DUMMY,
//            DCB=(RECFM=VB,BLKSIZE=1920,LRECL=1916,BUFNO=2)
//SYSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//            BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//            BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
```

```
//IMSMON   DD DUMMY
//DFSVSAMP DD DSN=&INDEXP..&ROOT.&ROOT.SY(DFSVSAM8),DISP=SHR
//SORTLIB  DD DSN=&SORTLIB,DISP=SHR
//SORTWK01 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//SORTWK02 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//SORTWK03 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//*:STEPCAT  DD DSN=&VSAMCAT,DISP=SHR
//*:         DD DSN=&SYSTCAT,DISP=SHR
//PAC7AE$SUF DD DSN=&INDEXX..&ROOTX.&ROOTX.AE,DISP=SHR
//PAC7AN$SUF DD DSN=&INDEXX..&ROOTX.&FILEX.AN,DISP=SHR
//PAC7AR$SUF DD DSN=&INDEXX..&ROOTX.&FILEX.AR,DISP=SHR
//PAC7TS   DD DSN=&INDEX..&ROOT.&FILE.TS,DISP=SHR
//PAC7PA$SUF DD DSN=&INDEXX..&ROOTX.&ROOTX.PA,DISP=SHR
//PAC7P1$SUF DD DSN=&INDEXX..&ROOTX.&ROOTX.P1,DISP=SHR
//PAC7CA   DD DSN=&&CINDMB1,DISP=(OLD,PASS)
//PAC7RT   DD DSN=&&PAC7RT,DISP=(,PASS),UNIT=&UWK,
//            DCB=(RECFM=FB,LRECL=170,BLKSIZE=25500),
//            SPACE=&SPART
//PAC7ET   DD SYSOUT=&OUT
//SYSOUT   DD SYSOUT=&OUT
//SYSOUX   DD SYSOUT=&OUT
//SYSPRINT DD SYSOUT=&OUT
//SYSUDUMP DD SYSOUT=&OUT
//*
//PNDC50  EXEC PGM=PNDC50
//STEPLIB  DD DSN=&STEPLIB,DISP=SHR
//*:STEPCAT  DD DSN=&VSAMCAT,DISP=SHR
//PAC7TY   DD DSN=&INDEX..&ROOT.&FILE.TY,DISP=SHR
//PAC7IN   DD DSN=&&CINDMB2,DISP=(OLD,PASS)
//PAC7BS   DD DSN=&&PAC7BS,DISP=(,PASS),UNIT=&UWK,
//            DCB=(RECFM=FB,LRECL=80,BLKSIZE=6080),
//            SPACE=&SPAMB
//PAC7BT   DD DSN=&&PAC7BT,DISP=(,PASS),UNIT=&UWK,
//            DCB=(RECFM=FB,LRECL=80,BLKSIZE=6080),
//            SPACE=&SPAMB
//SYSOUT   DD SYSOUT=&OUT
//SYSOUX   DD SYSOUT=&OUT
//SYSUDUMP DD SYSOUT=&OUT
//*
//NDVRC1A EXEC PGM=NDVRC1,DYNAMNBR=1500,REGION=4096K,PARM='C1BM3000'
//STEPLIB  DD DSN=&LOADLIB,DISP=SHR
//CONLIB   DD DSN=&CONLIB,DISP=SHR
//SYSPRINT DD SYSOUT=&OUT
//SYSUDUMP DD SYSOUT=&OUT
//SORTWK01 DD UNIT=SYSDA,SPACE=(CYL,(2,1))
//SORTWK02 DD UNIT=SYSDA,SPACE=(CYL,(2,1))
//SORTWK03 DD UNIT=SYSDA,SPACE=(CYL,(2,1))
//SORTWK04 DD UNIT=SYSDA,SPACE=(CYL,(2,1))
//C1TPDD01 DD UNIT=SYSDA,SPACE=(CYL,5),
//            DCB=(RECFM=VB,LRECL=260,BLKSIZE=6160)
//C1TPDD02 DD UNIT=SYSDA,SPACE=(CYL,5),
//            DCB=(RECFM=VB,LRECL=260,BLKSIZE=6160)
//C1TPLSIN DD UNIT=SYSDA,SPACE=(CYL,5),
//            DCB=(RECFM=FB,LRECL=80,BLKSIZE=6160)
//C1TPLSOU DD UNIT=SYSDA,SPACE=(CYL,5)
//PAC7TY   DD DSN=&INDEX..&ROOT.&FILE.TY,DISP=SHR
//PAC7PU   DD DSN=&INDEX..&ROOT.&FILE.UP,DISP=SHR
//PAC7UP   DD DSN=&INDEX..&ROOT.&FILE.UP,DISP=SHR
//PAC7UQ   DD DSN=&INDEX..&ROOT.&FILE.UQ,DISP=SHR
//C1PLMSGS DD SYSOUT=&OUT
//*1MSGS1  DD SYSOUT=&OUT
//C1MSGS1  DD DSN=&&C1MSGS1I,DISP=(,PASS),UNIT=&UWK,
//            DCB=(RECFM=FBA,LRECL=133,BLKSIZE=26600),
//            SPACE=&SPAMS
```

```
//*1PRINT  DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,BLKSIZE=6171)
//C1PRINT  DD DSN=&&C1PRINTI,DISP=(,PASS),UNIT=&UWK,
//            DCB=(RECFM=FBA,LRECL=133,BLKSIZE=26600),
//            SPACE=&SPAMS
//SYSABEND DD SYSOUT=&OUT
//SYSOUT   DD SYSOUT=&OUT
//BSTIPT01 DD DSN=&&PAC7BS,DISP=(OLD,PASS)
//*
//PNDC60  EXEC PGM=PNDC60
//STEPLIB  DD DSN=&STEPLIB,DISP=SHR
//SORTLIB  DD DSN=&SORTLIB,DISP=SHR
//SORTWK01 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//SORTWK02 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//SORTWK03 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//*:STEPCAT  DD DSN=&VSAMCAT,DISP=SHR
//PAC7TY   DD DSN=&INDEX..&ROOT.&FILE.TY,DISP=SHR
//PAC7LT   DD DSN=&&C1MSGS1I,DISP=(OLD,PASS)
//PAC7PT   DD DSN=&&C1PRINTI,DISP=(OLD,PASS)
//PAC7RS   DD DSN=&&PAC7RR,DISP=(,PASS),UNIT=&UWK,
//            DCB=(RECFM=FB,LRECL=170,BLKSIZE=25500),
//            SPACE=&SPART
//SYSOUT   DD SYSOUT=&OUT
//SYSOUX   DD SYSOUT=&OUT
//SYSUDUMP DD SYSOUT=&OUT
//*
//NDVRC1B EXEC PGM=NDVRC1,DYNAMNBR=1500,REGION=4096K,PARM='C1BM3000'
//STEPLIB  DD DSN=&LOADLIB,DISP=SHR
//CONLIB   DD DSN=&CONLIB,DISP=SHR
//SYSPRINT DD SYSOUT=&OUT
//SYSUDUMP DD SYSOUT=&OUT
//SORTWK01 DD UNIT=SYSDA,SPACE=(CYL,(2,1))
//SORTWK02 DD UNIT=SYSDA,SPACE=(CYL,(2,1))
//SORTWK03 DD UNIT=SYSDA,SPACE=(CYL,(2,1))
//SORTWK04 DD UNIT=SYSDA,SPACE=(CYL,(2,1))
//C1TPDD01 DD UNIT=SYSDA,SPACE=(CYL,5),
//            DCB=(RECFM=VB,LRECL=260,BLKSIZE=6160)
//C1TPDD02 DD UNIT=SYSDA,SPACE=(CYL,5),
//            DCB=(RECFM=VB,LRECL=260,BLKSIZE=6160)
//C1TPLSIN DD UNIT=SYSDA,SPACE=(CYL,5),
//            DCB=(RECFM=FB,LRECL=80,BLKSIZE=6160)
//C1TPLSOU DD UNIT=SYSDA,SPACE=(CYL,5)
//PAC7TY   DD DSN=&INDEX..&ROOT.&FILE.TY,DISP=SHR
//PAC7PU   DD DSN=&INDEX..&ROOT.&FILE.UP,DISP=SHR
//PAC7UP   DD DSN=&INDEX..&ROOT.&FILE.UP,DISP=SHR
//PAC7UQ   DD DSN=&INDEX..&ROOT.&FILE.UQ,DISP=SHR
//C1PLMSGS DD SYSOUT=&OUT
//*1MSGS1  DD SYSOUT=&OUT
//C1MSGS1  DD DSN=&&C1MSGS1P,DISP=(,PASS),UNIT=&UWK,
//            DCB=(RECFM=FBA,LRECL=133,BLKSIZE=26600),
//            SPACE=&SPAMS
//*1PRINT  DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,BLKSIZE=6171)
//C1PRINT  DD DSN=&&C1PRINTP,DISP=(,PASS),UNIT=&UWK,
//            DCB=(RECFM=FBA,LRECL=133,BLKSIZE=26600),
//            SPACE=&SPAMS
//SYSABEND DD SYSOUT=&OUT
//SYSOUT   DD SYSOUT=&OUT
//BSTIPT01 DD DSN=&&PAC7BT,DISP=(OLD,PASS)
//*
//PNDC70  EXEC PGM=PNDC70
//STEPLIB  DD DSN=&STEPLIB,DISP=SHR
//SORTLIB  DD DSN=&SORTLIB,DISP=SHR
//SORTWK01 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//SORTWK02 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//SORTWK03 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
```

```
//*:STEPCAT  DD DSN=&VSAMCAT,DISP=SHR
//PAC7TY   DD DSN=&INDEX..&ROOT.&FILE.TY,DISP=SHR
//PAC7LT   DD DSN=&&C1MSGS1P,DISP=(OLD,PASS)
//PAC7PT   DD DSN=&&C1PRINTP,DISP=(OLD,PASS)
//PAC7RR   DD DSN=&&PAC7RR,DISP=(OLD,PASS)
//PAC7RS   DD DSN=&&PAC7RS,DISP=(,PASS),UNIT=&UWK,
//            DCB=(RECFM=FB,LRECL=170,BLKSIZE=25500),
//            SPACE=&SPART
//SYSOUT   DD SYSOUT=&OUT
//SYSOUX   DD SYSOUT=&OUT
//SYSUDUMP DD SYSOUT=&OUT
//*
//PNDC90  EXEC PGM=DFSRRC00,REGION=$REGSIZ,
//        PARM=(DLI,PNDC90,PNDC90$SUG,&BUF,
//        &SPIE&TEST&EXCPVR&RST,&PRLD,
//        &SRCH,&CKPTID,&MON,&LOGA,&FMTO,,,&DBRC,&IRLM)
//STEPLIB  DD DSN=&RESLIB,DISP=SHR
//         DD DSN=&STEPLIB,DISP=SHR
//DFSRESLB DD DSN=&RESLIB,DISP=SHR
//IMS      DD DSN=&PSBLIB,DISP=SHR
//         DD DSN=&DBDLIB,DISP=SHR
//DDSNAP   DD SYSOUT=&OUT
//PROCLIB  DD DSN=&PROCLIB,DISP=SHR
//IEFRDER  DD DUMMY,
//            DCB=(RECFM=VB,BLKSIZE=1920,LRECL=1916,BUFNO=2)
//SYSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//            BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//            BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSMON   DD DUMMY
//DFSVSAMP DD DSN=&INDEXP..&ROOT.&ROOT.SY(DFSVSAM8),DISP=SHR
//*:STEPCAT  DD DSN=&VSAMCAT,DISP=SHR
//*:         DD DSN=&SYSTCAT,DISP=SHR
//SORTLIB  DD DSN=&SORTLIB,DISP=SHR
//SORTWK01 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//SORTWK02 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//SORTWK03 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//PAC7AE$SUF DD DSN=&INDEXX..&ROOTX.&ROOTX.AE,DISP=SHR
//PAC7AR$SUF DD DSN=&INDEXX..&ROOTX.&FILEX.AR,DISP=SHR
//PAC7TS   DD DSN=&INDEX..&ROOT.&FILE.TS,DISP=SHR
//PAC7CA   DD DSN=&&CINDMB1,DISP=(OLD,PASS)
//PAC7UM   DD DSN=&&PAC7RT,DISP=(OLD,PASS)
//PAC7UN   DD DSN=&&PAC7RS,DISP=(OLD,PASS)
//PAC7UR   DD DSN=&&PAC7UR,DISP=(,PASS),UNIT=&UWK,
//            DCB=(RECFM=FB,LRECL=183,BLKSIZE=18300),
//            SPACE=&SPAMV
//PAC7ET   DD SYSOUT=&OUT
//SYSOUT   DD SYSOUT=&OUT
//SYSOUX   DD SYSOUT=&OUT
//SYSPRINT DD SYSOUT=&OUT
//SYSUDUMP DD SYSOUT=&OUT
//*
//PNDM10  EXEC PGM=DFSRRC00,REGION=$REGSIZ,
//        PARM=(DLI,PNDM10,PNDM10$SUG,&BUF,
//        &SPIE&TEST&EXCPVR&RST,&PRLD,
//        &SRCH,&CKPTID,&MON,&LOGA,&FMTO,,,&DBRC,&IRLM)
//STEPLIB  DD DSN=&RESLIB,DISP=SHR
//         DD DSN=&STEPLIB,DISP=SHR
//         DD DSN=&BCOB,DISP=SHR
//DFSRESLB DD DSN=&RESLIB,DISP=SHR
//IMS      DD DSN=&PSBLIB,DISP=SHR
//         DD DSN=&DBDLIB,DISP=SHR
//DDSNAP   DD SYSOUT=&OUT
//PROCLIB  DD DSN=&PROCLIB,DISP=SHR
```

```
//IEFRDER  DD DUMMY,
//            DCB=(RECFM=VB,BLKSIZE=1920,LRECL=1916,BUFNO=2)
//SYSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//            BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//            BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSMON   DD DUMMY
//DFSVSAMP DD DSN=&INDEXP..&ROOT.&ROOT.SY(DFSVSAM8),DISP=SHR
//*:STEPCAT  DD DSN=&VSAMCAT,DISP=SHR
//*:         DD DSN=&SYSTCAT,DISP=SHR
//PAC7AE$SUF DD DSN=&INDEXX..&ROOTX.&ROOTX.AE,DISP=SHR
//PAC7AN$SUF DD DSN=&INDEXX..&ROOTX.&FILEX.AN,DISP=SHR
//PAC7AR$SUF DD DSN=&INDEXX..&ROOTX.&FILEX.AR,DISP=SHR
//PAC7PA$SUF DD DSN=&INDEXX..&ROOTX.&ROOTX.PA,DISP=SHR
//PAC7P1$SUF DD DSN=&INDEXX..&ROOTX.&ROOTX.P1,DISP=SHR
//PAC7UR   DD DSN=&&PAC7UR,DISP=(OLD,PASS)
//PAC7MV   DD DSN=&&PAC7MV,DISP=(,PASS),UNIT=&UWK,
//            DCB=(RECFM=FB,LRECL=80,BLKSIZE=8800),
//            SPACE=&SPAMV
//PAC7ET   DD SYSOUT=&OUT
//SYSOUT   DD SYSOUT=&OUT
//SYSOUX   DD SYSOUT=&OUT
//SYSPRINT DD SYSOUT=&OUT
//SYSUDUMP DD SYSOUT=&OUT
//*
```

7.4. GPND: PREPARATION FOR IMPORT TO ENDEVOR

GPND: PREPARATION FOR IMPORT TO ENDEVOR

The GPND procedure completes the file resulting from VA Pac generation-print procedure (GPRT) and prepares the import job into ENDEVOR of the generated VA Pac and INFOPAC ELEMENTs to be processed.

EXECUTION CONDITION

GPRT procedure must be executed first.

RESULT

This procedure formats a transactions list for the update of ENDEVOR by the UPND procedure (executed automatically when indicated in the control commands after the program).

DESCRIPTION OF STEPS

This procedure comprises one step:

. Completion of the file resulting from generation and preparation for import into Endevor: PNDV10

```
PREPARATION FOR IMPORT INTO ENDEVOR: PNDV10

. Input files:
 - VA Pac errror messages file                : PAC7AE
   DSN=&INDEXX..&ROOTX.&ROOTX.AE
 - VA Pac index file                          : PAC7AN
   DSN=&INDEXX..&ROOTX.&FILEX.AN
 - VA Pac data file                           : PAC7AR
   DSN=&INDEXX..&ROOTX.&FILEX.AR
 - Interface TYPES file                       : PAC7TY
   DSN=&INDEX..&ROOT.&FILE.TY
 - LIBRARIES and SESSIONS file                : PAC7TS
   DSN=&INDEX..&ROOT.&FILE.TS
 - VA Pac generated code file                 : PAC7JB
   DSN=&&PAC7JB

. Output files:
 - Execution errors report                    : PAC7ET
   DSN=&OUT
 - Import file to be executed                 : PAC7BJ
   DSN=&&PAC7BJ
```

```
//******************************************************************
//* VA PAC-ENDEVOR 2.5   : PARAMETERS SPECIFICATION                *
//******************************************************************
//$RADP.GPND PROC FILE=$FILE,         NUMBER OF PHYSICAL DATABASE
//          FILEX=$FILEX,             NUMBER OF VA PAC PHYSICAL DBASE
//          ROOT=$ROOT,               ROOT OF THE SYSTEM
//          ROOTX=$ROOTX,             ROOT OF 'VA PAC' SYSTEM
//          INDEX='$INDEX',           INDEX OF VSAM SYSTEM FILES
//          INDEXX='$INDEXX',         INDEX OF VA PAC SYSTEM VSAMFILES
//          INDEXP='$INDEXP',         INDEX OF NON VSAM SYSTEM FILES
//*:        VSAMCAT='$CATU',          USER VSAM CATALOG
//*:        SYSTCAT='$CATV',          SYSTEM VSAM CATALOG
//          STEPLIB='$MODB',          LIBRARY OF LOAD-MODULES
//          OUT=$OUT,                 OUTPUT CLASS
//          UWK=$UWK,                 WORK UNIT
//          PAC7GB='????',            'PAC7GB' DSN FROM GPRT PROC
//          SPABJ='(TRK,(02,01),RLSE)',     GENERATION FILE SPACE
//     PSBLIB='$PSBLIB',     PSB LIBRARY
//     DBDLIB='$DBDLIB',     DBD LIBRARY
//     RESLIB='$RESLIB',     RESLIB IMS
//     PROCLIB='$PRCLIB',    PROCLIB IMS
// BUF=40,SPIE=0,TEST=0,EXCPVR=0,RST=0,PRLD=,SRCH=0,
// CKPTID=,MON=N,LOGA=0,FMTO=T,DBRC=$DBRC,IRLM=$IRLM
//*----------------------------------------------------------------*
//*
//PNDV10  EXEC PGM=DFSRRC00,REGION=$REGSIZ,
//         PARM=(DLI,PNDV10,PNDV10$SUG,&BUF,
//         &SPIE&TEST&EXCPVR&RST,&PRLD,
//         &SRCH,&CKPTID,&MON,&LOGA,&FMTO,,,&DBRC,&IRLM)
//STEPLIB  DD DSN=&RESLIB,DISP=SHR
//         DD DSN=&STEPLIB,DISP=SHR
//         DD DSN=$BCOB,DISP=SHR
//DFSRESLB DD DSN=&RESLIB,DISP=SHR
//IMS      DD DSN=&PSBLIB,DISP=SHR
//         DD DSN=&DBDLIB,DISP=SHR
//DDSNAP   DD SYSOUT=&OUT
//PROCLIB  DD DSN=&PROCLIB,DISP=SHR
//IEFRDER  DD DUMMY,
//            DCB=(RECFM=VB,BLKSIZE=1920,LRECL=1916,BUFNO=2)
//SYSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//            BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//            BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSMON   DD DUMMY
//DFSVSAMP DD DSN=&INDEXP..&ROOT.&ROOT.SY(DFSVSAM8),DISP=SHR
//*:STEPCAT  DD DSN=&VSAMCAT,DISP=SHR
//*:         DD DSN=&SYSTCAT,DISP=SHR
//PAC7AE$SUF DD DSN=&INDEXX..&ROOTX.&ROOTX.AE,DISP=SHR
//PAC7AN$SUF DD DSN=&INDEXX..&ROOTX.&FILEX.AN,DISP=SHR
//PAC7AR$SUF DD DSN=&INDEXX..&ROOTX.&FILEX.AR,DISP=SHR
//PAC7PA$SUF DD DSN=&INDEXX..&ROOTX.&ROOTX.PA,DISP=SHR
//PAC7P1$SUF DD DSN=&INDEXX..&ROOTX.&ROOTX.P1,DISP=SHR
//PAC7TS   DD DSN=&INDEX..&ROOT.&FILE.TS,DISP=SHR
//PAC7TY   DD DSN=&INDEX..&ROOT.&FILE.TY,DISP=SHR
//PAC7JB   DD DSN=&PAC7GB,DISP=(OLD,PASS)
//PAC7BJ   DD DSN=&&PAC7BJ,DISP=(,PASS),UNIT=&UWK,
//            DCB=(RECFM=FB,LRECL=80,BLKSIZE=8000),
//            SPACE=&SPABJ
//PAC7ET   DD SYSOUT=&OUT
//SYSOUT   DD SYSOUT=&OUT
//SYSOUX   DD SYSOUT=&OUT
//SYSPRINT DD SYSOUT=&OUT
//SYSUDUMP DD SYSOUT=&OUT
//*
```

```
//IEBGN   EXEC PGM=IEBGENER
//SYSUT1   DD DSN=&&PAC7BJ,DISP=(OLD,PASS)
//SYSUT2   DD SYSOUT=(&OUT,INTRDR)
//SYSIN    DD DUMMY
//SYSOUT   DD SYSOUT=&OUT
//SYSPRINT DD SYSOUT=&OUT
//*
```

7.5. INDD: LIST OF PROGRAM DATES

INDD: LIST OF PROGRAM DATES

The INDD procedure supplies the list of the interface programs' generation dates.

EXECUTION CONDITION

None.

DESCRIPTION OF STEPS

This procedure comprises one step:

. List of program dates: PNXDAT

```
LIST OF PROGRAM DATES: PNXDAT

. Input files:
 - VA Pac error messages file                : PAC7AE
   DSN=&INDEXX..&ROOTX.&ROOTX.AE
 - Transaction file                           : PAC7MB
   DUMMY

. Output file:
 - List of program dates report               : PACDDS
   DSN=&OUT
```

```
//*******************************************************************
//* VA PAC-ENDEVOR 2.5   : DATES OF LOAD MODULES                    *
//*******************************************************************
//$RADP.INND PROC ROOTX=$ROOTX,      ROOT OF 'VA PAC' SYSTEM
//         ROOT=$ROOT,               ROOT OF VA PAC ENDEVOR
//         INDEXX='$INDEXX',         INDEX OF VA PAC VSAM SYSTEM FILES
//         INDEXP='$INDEXP',         INDEX OF NON VSAM SYSTEM FILES
//*:       SYSTCAT='$CATV',          VSAM SYSTEM CATALOG
//         STEPLIB='$MODB',          LIBRARY OF LOAD-MODULES
//         OUT=$OUT,                 OUTPUT CLASS
//    PSBLIB='$PSBLIB',     PSB LIBRARY
//    DBDLIB='$DBDLIB',     DBD LIBRARY
//    RESLIB='$RESLIB',     RESLIB IMS
//    PROCLIB='$PRCLIB',    PROCLIB IMS
// BUF=40,SPIE=0,TEST=0,EXCPVR=0,RST=0,PRLD=,SRCH=0,
// CKPTID=,MON=N,LOGA=0,FMTO=T,DBRC=$DBRC,IRLM=$IRLM
//*******************************************************************
//PNXDAT  EXEC PGM=DFSRRC00,REGION=$REGSIZ,
//        PARM=(DLI,PNXDAT,PNXDAT$SUG,&BUF,
//        &SPIE&TEST&EXCPVR&RST,&PRLD,
//        &SRCH,&CKPTID,&MON,&LOGA,&FMTO,,,&DBRC,&IRLM)
//STEPLIB  DD DSN=&RESLIB,DISP=SHR
//         DD DSN=&STEPLIB,DISP=SHR
//DFSRESLB DD DSN=&RESLIB,DISP=SHR
//IMS      DD DSN=&PSBLIB,DISP=SHR
//         DD DSN=&DBDLIB,DISP=SHR
//DDSNAP   DD SYSOUT=&OUT
//PROCLIB  DD DSN=&PROCLIB,DISP=SHR
//IEFRDER  DD DUMMY,
//            DCB=(RECFM=VB,BLKSIZE=1920,LRECL=1916,BUFNO=2)
//SYSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//            BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//            BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSMON   DD DUMMY
//DFSVSAMP DD DSN=&INDEXP..&ROOT.&ROOT.SY(DFSVSAM8),DISP=SHR
//*:STEPCAT  DD DSN=&VSAMCAT,DISP=SHR
//*:         DD DSN=&SYSTCAT,DISP=SHR
//PAC7AE$SUF DD DSN=&INDEXX..&ROOTX.&ROOTX.AE,DISP=SHR
//SYSOUT   DD SYSOUT=&OUT
//SYSOUX   DD SYSOUT=&OUT
//PACDDS   DD SYSOUT=&OUT
//PAC7MB   DD DUMMY
//SYSUDUMP DD SYSOUT=&OUT
//SYSPRINT DD SYSOUT=&OUT
//*
```

7.6. JJND: RETRIEVAL OF ARCHIVED JOURNAL FILE

JJND: RETRIEVAL OF ARCHIVED JOURNAL FILE

The JJND procedure retrieves the archived journal file from release 1.6 and transforms it into a journal operational in 2.5. It adapts the date of the last action to make it compatible with the Year 2000 reqirements.

EXECUTION CONDITION

This procedure is executed when switching from release 1.6 to release 2.5.

```
DESCRIPTION OF STEPS


This procedure comprises one step:
 . Conversion of the archived journal         : PNRJ10


RETRIEVAL OF ARCHIVED JOURNAL: PNRJ10

. Input files:
 - Old archived journal file (release 1.6)    : PAC7QI
   DSN: indicated when giving a value to
   &JNARCH parameter
   (Record length = 135)

. Output file:
 - Archived journal file of release 2.5       : PAC7QR
   (generation +1)
   DSN=&INDEXQ..&ROOT.&FILE.QU(+1)
   (Record length = 187)
```

```
//*******************************************************************
//* VA PAC-ENDEVOR 2.5   : CONVERTING THE ARCHIVED JOURNAL            *
//*******************************************************************
//$RADP.JJND PROC FILE=$FILE,      NUMBER OF PHYSICAL DATABASE
//   ROOT=$ROOT,                   ROOT OF THE SYSTEM
//   INDEXQ='$INDEXQ',             INDEX OF THE USER NON VSAM FILLES
//   STEPLIB='$MODB',              LOAD-MODULES LIBRARY
//   OUT=$OUT,                     OUTPUT CLASS
//   UNITS=$UNITO,                 'QU' SAVE UNIT
//   VOLS='SER=$VOLO',             'QU' SAVE VOLUME
//   SPAQU='(TRK,(10,05),RLSE)',   'QU' SAVE SPACE
//   JNARCH='????????'             SAVE OF THE ARCHIVED JOURNAL
//*                                FROM A PREVIOUS RELEASE
//*-------------------------------------------------------------
//PNRJ10  EXEC PGM=PNRJ10
//STEPLIB  DD DSN=&STEPLIB,DISP=SHR
//PAC7QI   DD DSN=&JNARCH,DISP=SHR
//PAC7QR   DD DSN=&INDEXQ..&ROOT.&FILE.QU(+1),DISP=(,CATLG,DELETE),
//            UNIT=&UNITS,VOL=&VOLS,SPACE=&SPAQU,
//            DCB=(&INDEXQ..DSCB.&ROOT.&FILE.QU,
//            RECFM=FB,LRECL=187,BLKSIZE=18700)
//SYSOUT   DD SYSOUT=&OUT
//SYSOUX   DD SYSOUT=&OUT
//SYSUDUMP DD SYSOUT=&OUT
//*
```

7.7. JRND: RETRIEVAL OF ARCHIVED JOURNAL TRANSACTIONS

INTRODUCTION

The JRND procedure retrieves the journal transactions which have already been archived in QU and prepares the VA Pac Batch update transactions by selecting the archiving date and time.

EXECUTION CONDITION

None.

USER INPUT

.One administrator identification line.

Pos.	Len.	Value	Meaning
2	1	'*'	Line code
3	8	uuuuuuuu	User code
11	8	mmmmmmmm	Password

.One selection line of the archived transactions to be retrieved.

Pos.	Len.	Value	Meaning
1	8		Selection starting date, formatted CCYYMMDD
9	6		Seletion starting time, formatted HHMMSS
15	8		Selection ending date, formatted CCYYMMDD
23	6		Selection ending time, formatted HHMMSS

OUTPUT REPORT

Generation result report.

RESULT

This procedure generates a VA Pac Batch update transactions file for the UPDT procedure.

DESCRIPTION OF STEPS

This procedure comprised seven steps:

- Recognition of user input : PTU001
- Allocation of UU selected transactions file : IDCAMS
- Initialization of UU selected transactions file: PNINUQ
- Retrieval of QU journal transactions on UU : PNDJ10
- Preparation for update transactions generation : PNDM05
- Generation of VA Pac update transactions : PNDM10
- Deletion of the selected transactions file : IDCAMS

RECOGNITION OF USER INPUT: PTU001

. Input data : CARTE
DSN=SYSIN

. Output file : PAC7MB
DSN=&&JRNDMB

ALLOCATION OF UU SELECTED TRANSACTIONS FILE: IDCAMS

. Input file:
- Definition of selected transactions file : SYSIN
DSN=&INDEXP..&ROOT.&ROOT.SY(DF&ROOT.&FILE.UU)

INITIALIZATION OF UU FILE: PNINUQ

. Output file:
- UU selected transactions file : PAC7UQ
DSN=&INDEX..&ROOT.&FILE.UU

```
RETRIEVAL OF ARCHIVED JOURNAL TRANSACTIONS: PNDJ10

. Input files:
 - LIBRARY and SESSIONS file                    : PAC7IN
   DSN=&INDEX..&ROOT.&FILE.TS
 - Archived journal file                        : PAC7US
   DSN=&INDEXQ..&ROOT.&FILE.QU(0)
 - VA Pac error messages file                   : PAC7AE
   DSN=&INDEXX..&ROOTX.&ROOTX.AE
 - User input file                              : PAC7MB
   DSN=&&JRNDMB

. Input-output file:
 - UU selected transactions file                : PAC7UQ
   DSN=&INDEX..&ROOT.&FILE.UU

. Output file:
 - Administrator identification transaction     : PAC7MV
   DSN=&INPUT
 - Anomaly report                              : PAC7ET
   DSN=&OUT


PREPARATION FOR UPDATE TRANSACTIONS GENERATION: PNDM05

. Input files:
 - VA Pac data file                            : PAC7AR
   DSN=&INDEXX..&ROOTX.&FILEX.AR
 - LIBRARIES and SESSIONS file                  : PAC7TS
   DSN=&INDEX..&ROOT.&ROOT.TS
 - UU selected transactions file                : PAC7UQ
   DSN=&INDEX..&ROOT.&FILE.UU
 - Administrator identification transaction     : PAC7MV
   DSN=&INPUT

. Output file:
 - Endevor transactions file                    : PAC7UR
   DSN=&&PAC7UR
```

```
GENERATION OF VA Pac UPDATE TRANSACTIONS: PNDM10

. Input files:
 - Endevor transactions file                    : PAC7UR
   DSN=&&PAC7UR
 - VA Pac index file                           : PAC7AN
   DSN=&INDEXX..&ROOTX.&FILEX.AN
 - VA Pac data file                            : PAC7AR
   DSN=&INDEXX..&ROOTX.&FILEX.AR
 - VA Pac error messages file                  : PAC7AE
   DSN=&INDEXX..&ROOTX.&ROOTX.AE

. Output file:
 - VA Pac update transactions file             : PAC7MV
   DSN=&&PAC7MV
 - Anomaly report                              : PAC7ET
   DSN=&OUT


DELETION OF THE UU SELECTED TRANSACTIONS FILE: IDCAMS


. Input file:
 - Delete command of UU file                    : SYSIN
  DSN=&INDEXP..&ROOT.&ROOT.SY(DL&ROOT.&FILE.UU)
```

```
//**********************************************************************
//* VA PAC-ENDEVOR 2.5   : RETRIEVAL OF ARCHIVED JOURNAL TRANSACTTIONS*
//**********************************************************************
//$RADP.JRND PROC FILE=$FILE,         NUMBER OF PHYSICAL DATABASE
//         FILEX=$FILEX,              NUMBER OF VA PAC PHYSICAL DBASE
//         ROOT=$ROOT,                ROOT OF THE SYSTEM
//         ROOTX=$ROOTX,              ROOT OF 'VA PAC' SYSTEM
//         INDEX='$INDEX',            INDEX OF VSAM SYSTEM FILES
//         INDEXX='$INDEXX',          INDEX OF VA PAC VSAM SYSTEMFILES
//         INDEXP='$INDEXP',          INDEX OF NON VSAM SYSTEM FILES
//         INDEXQ='$INDEXQ',          INDEX OF NON VSAM USER FILES
//*:       VSAMCAT='$CATU',           USER VSAM CATALOG
//*:       SYSTCAT='$CATV',           SYSTEM VSAM CATALOG
//         STEPLIB='$MODB',           LIBRARY OF LOAD-MODULES
//         SORTLIB='$BIBT',           SORT LIBRARY
//         CYL=3,                     SORT WORKS SIZE
//         OUT=$OUT,                  OUTPUT CLASS
//         UWK=$UWK,                  WORK UNIT
//         SPAUR='(TRK,(02,01),RLSE)', TEMPORARY FILE SPACE
//         SPAMV='(TRK,(05,01),RLSE)', UPDATE TRANS. FILE SPACE
//    PSBLIB='$PSBLIB',     PSB LIBRARY
//    DBDLIB='$DBDLIB',     DBD LIBRARY
//    RESLIB='$RESLIB',     RESLIB IMS
//    PROCLIB='$PRCLIB',    PROCLIB IMS
// BUF=40,SPIE=0,TEST=0,EXCPVR=0,RST=0,PRLD=,SRCH=0,
// CKPTID=,MON=N,LOGA=0,FMTO=T,DBRC=$DBRC,IRLM=$IRLM
//*--------------------------------------------------------------------*
//*
//INPUT   EXEC PGM=PTU001
//STEPLIB  DD DSN=&STEPLIB,DISP=SHR
//CARTE    DD DDNAME=SYSIN,DCB=BLKSIZE=80
//PAC7MB   DD DSN=&&JRNDMB,DISP=(,PASS),
//            UNIT=&UWK,SPACE=(TRK,(1,1),RLSE),
//            DCB=(RECFM=FB,LRECL=80,BLKSIZE=80)
//*
//DEFINE  EXEC PGM=IDCAMS
//*:STEPCAT DD DSN=&VSAMCAT,DISP=SHR
//SYSPRINT DD SYSOUT=&OUT
//SYSIN    DD DSN=&INDEXP..&ROOT.&ROOT.SY(DF&ROOT.&FILE.UU),DISP=SHR
//*
//PNINUQ EXEC PGM=PNINUQ
//STEPLIB  DD DSN=&STEPLIB,DISP=SHR
//PAC7UQ   DD DSN=&INDEX..&ROOT.&FILE.UU,DISP=SHR
//SYSOUT   DD SYSOUT=*
//SYSUDUMP DD SYSOUT=*
//*
//PNDJ10  EXEC PGM=DFSRRC00,REGION=$REGSIZ,
//        PARM=(DLI,PNDJ10,PNDJ10$SUG,&BUF,
//        &SPIE&TEST&EXCPVR&RST,&PRLD,
//        &SRCH,&CKPTID,&MON,&LOGA,&FMTO,,,&DBRC,&IRLM)
//STEPLIB  DD DSN=&RESLIB,DISP=SHR
//         DD DSN=&STEPLIB,DISP=SHR
//DFSRESLB DD DSN=&RESLIB,DISP=SHR
//IMS      DD DSN=&PSBLIB,DISP=SHR
//         DD DSN=&DBDLIB,DISP=SHR
//DDSNAP   DD SYSOUT=&OUT
//PROCLIB  DD DSN=&PROCLIB,DISP=SHR
//IEFRDER  DD DUMMY,
//            DCB=(RECFM=VB,BLKSIZE=1920,LRECL=1916,BUFNO=2)
//SYSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//            BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//            BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSMON   DD DUMMY
```

```
//DFSVSAMP DD DSN=&INDEXP..&ROOT.&ROOT.SY(DFSVSAM8),DISP=SHR
//*:STEPCAT DD DSN=&VSAMCAT,DISP=SHR
//*:        DD DSN=&SYSTCAT,DISP=SHR
//PAC7IN   DD DSN=&INDEX..&ROOT.&FILE.TS,DISP=SHR
//PAC7UQ   DD DSN=&INDEX..&ROOT.&FILE.UU,DISP=SHR
//PAC7US   DD DSN=&INDEXQ..&ROOT.&FILE.QU(0),DISP=SHR
//PAC7AE$SUF DD DSN=&INDEXX..&ROOTX.&ROOTX.AE,DISP=SHR
//PAC7MB   DD DSN=&&JRNDMB,DISP=(OLD,DELETE)
//PAC7MV   DD DSN=&&JRNDMV,DISP=(,PASS),
//            UNIT=&UWK,SPACE=(TRK,(1,1),RLSE),
//            DCB=(RECFM=FB,LRECL=80,BLKSIZE=80)
//PAC7ET   DD SYSOUT=&OUT
//SYSOUT   DD SYSOUT=&OUT
//SYSOUX   DD SYSOUT=&OUT
//SYSPRINT DD SYSOUT=&OUT
//SYSUDUMP DD SYSOUT=&OUT
//*
//PNDM05  EXEC PGM=DFSRRC00,REGION=$REGSIZ,
//        PARM=(DLI,PNDM05,PNDM05$SUG,&BUF,
//        &SPIE&TEST&EXCPVR&RST,&PRLD,
//        &SRCH,&CKPTID,&MON,&LOGA,&FMTO,,,&DBRC,&IRLM)
//STEPLIB  DD DSN=&RESLIB,DISP=SHR
//         DD DSN=&STEPLIB,DISP=SHR
//DFSRESLB DD DSN=&RESLIB,DISP=SHR
//IMS      DD DSN=&PSBLIB,DISP=SHR
//         DD DSN=&DBDLIB,DISP=SHR
//DDSNAP   DD SYSOUT=&OUT
//PROCLIB  DD DSN=&PROCLIB,DISP=SHR
//IEFRDER  DD DUMMY,
//            DCB=(RECFM=VB,BLKSIZE=1920,LRECL=1916,BUFNO=2)
//SYSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//            BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//            BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSMON   DD DUMMY
//DFSVSAMP DD DSN=&INDEXP..&ROOT.&ROOT.SY(DFSVSAM8),DISP=SHR
//SORTLIB  DD DSN=&SORTLIB,DISP=SHR
//SORTWK01 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//SORTWK02 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//SORTWK03 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//*:STEPCAT DD DSN=&VSAMCAT,DISP=SHR
//*:        DD DSN=&SYSTCAT,DISP=SHR
//PAC7AR$SUF DD DSN=&INDEXX..&ROOTX.&FILEX.AR,DISP=SHR
//PAC7TS   DD DSN=&INDEX..&ROOT.&FILE.TS,DISP=SHR
//PAC7UQ   DD DSN=&INDEX..&ROOT.&FILE.UU,DISP=SHR
//PAC7MV   DD DSN=&&JRNDMV,DISP=(OLD,DELETE)
//PAC7UR   DD DSN=&&PAC7UR,DISP=(,PASS),UNIT=&UWK,
//            DCB=(RECFM=FB,LRECL=183,BLKSIZE=18300),
//            SPACE=&SPAUR
//SYSOUT   DD SYSOUT=&OUT
//SYSOUX   DD SYSOUT=&OUT
//SYSPRINT DD SYSOUT=&OUT
//SYSUDUMP DD SYSOUT=&OUT
//*
//PNDM10  EXEC PGM=DFSRRC00,REGION=$REGSIZ,
//        PARM=(DLI,PNDM10,PNDM10$SUG,&BUF,
//        &SPIE&TEST&EXCPVR&RST,&PRLD,
//        &SRCH,&CKPTID,&MON,&LOGA,&FMTO,,,&DBRC,&IRLM)
//STEPLIB  DD DSN=&RESLIB,DISP=SHR
//         DD DSN=&STEPLIB,DISP=SHR
//         DD DSN=$BCOB,DISP=SHR
//DFSRESLB DD DSN=&RESLIB,DISP=SHR
//IMS      DD DSN=&PSBLIB,DISP=SHR
//         DD DSN=&DBDLIB,DISP=SHR
```

```
//DDSNAP   DD SYSOUT=&OUT
//PROCLIB  DD DSN=&PROCLIB,DISP=SHR
//IEFRDER  DD DUMMY,
//            DCB=(RECFM=VB,BLKSIZE=1920,LRECL=1916,BUFNO=2)
//SYSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//            BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//            BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSMON   DD DUMMY
//DFSVSAMP DD DSN=&INDEXP..&ROOT.&ROOT.SY(DFSVSAM8),DISP=SHR
//*:STEPCAT DD DSN=&VSAMCAT,DISP=SHR
//*:        DD DSN=&SYSTCAT,DISP=SHR
//PAC7AE$SUF DD DSN=&INDEXX..&ROOTX.&ROOTX.AE,DISP=SHR
//PAC7AN$SUF DD DSN=&INDEXX..&ROOTX.&FILEX.AN,DISP=SHR
//PAC7AR$SUF DD DSN=&INDEXX..&ROOTX.&FILEX.AR,DISP=SHR
//PAC7PA$SUF DD DSN=&INDEXX..&ROOTX.&ROOTX.PA,DISP=SHR
//PAC7UR   DD DSN=&&PAC7UR,DISP=(OLD,PASS)
//PAC7P1$SUF DD DSN=&INDEXX..&ROOTX.&ROOTX.P1,DISP=SHR
//PAC7MV   DD DSN=&&PAC7MV,DISP=(,PASS),UNIT=&UWK,
//            DCB=(RECFM=FB,LRECL=80,BLKSIZE=8800),
//            SPACE=&SPAMV
//PAC7ET   DD SYSOUT=&OUT
//SYSOUT   DD SYSOUT=&OUT
//SYSOUX   DD SYSOUT=&OUT
//SYSPRINT DD SYSOUT=&OUT
//SYSUDUMP DD SYSOUT=&OUT
//*
//DELETE  EXEC PGM=IDCAMS,COND=((00,NE,PNDM05),(00,NE,PNDM10))
//*:STEPCAT DD DSN=&VSAMCAT,DISP=SHR
//SYSPRINT DD SYSOUT=&OUT
//SYSIN    DD DSN=&INDEXP..&ROOT.&ROOT.SY(DL&ROOT.&FILE.UU),DISP=SHR
//*
```

7.8. LSND: LIBRARY/SESSION UPDATES

LSND: LIBRARY/SESSION UPDATE

The LSND procedure updates the TARGET LIBRARIES and SESSIONS table used by the interface.

EXECUTION CONDITION

None.

USER INPUT

. One line per update request.

SESSION UPDATE PARAMETER LINE

Pos.	Length	Value	Meaning
1	1		Transaction code
		'C'	Creation
		'M'	Modification
		'D'	Deletion
		'X'	Creation or Modification
2	4	base	VA Pac Database code
6	2	'NS'	Line code
8	4	ssss	Target session number
12	1		Target session status
		'Z'	'9999' current session
		'T'	Frozen session
13	3	nnn	Line number
16	5	sssss	Starting session number
21	5	sssss	Ending session number
26	36		Comments

Starting and ending sessions may have a 'Z' or 'T' status.
'T' status is included in 'Z' for the same session.

LIBRARY UPDATE PARAMETER LINE

Pos.	Length	Value	Meaning
1	1		Transaction code
		'C'	Creation
		'M'	Modification
		'D'	Deletion
		'X'	Creation or Modification
2	4	base	VA Pac Database code
6	2	'NB'	Line code
8	3	bbb	Target library code
13	3	nnn	Line number
16	3	bbb	VA Pac generation library code
19	36		Comments

OUTPUT REPORTS

An execution report with the detected anomalies and a report which lists the TYPES, TARGET SESSIONS and LIBRARIES defined on the site.

DESCRIPTION OF STEPS

This procedrue comprises three steps:

. Recognition of input : PTU001

. Update of LIBRARIES and SESSIONS file : PNDU10

. Printing of TYPEs file and of TARGET LIBRARIES
and SESSIONS file : PNDU30

RECOGNITION OF INPUT: PTU001

. Input data : CARTE
DSN=SYSIN

. Output file : PAC7MB
DSN=&&LSNDMB

LIBRARIES AND SESSIONS UPDATE: PNDU10

. Input/output file:
- LIBRARIES and SESSIONS file : PAC7TS
DSN=&INDEX..&ROOT.&FILE.TS

. Input files:
- VA Pac error messages file : PAC7AE
DSN=&INDEXX..&ROOTX.&ROOTX.AE
- Update transactions : PAC7MV
DSN=&INPUT

. Output report:
- Update report : PAC7ET
DSN=&OUT

PRINTING OF TYPES AND OF TARGET LIBRARIES/SESSIONS
FILES: PNDU30

. Input files:
- TYPES file : PAC7TY
DSN=&INDEX..&ROOT.&FILE.TY
- LIBRARIES and SESSIONS file : PAC7TS
DSN=&INDEX..&ROOT.&FILE.TS
- VA Pac error messages file : PAC7AE
DSN=&INDEXX..&ROOTX.&ROOTX.AE

. Output file:
- Printout of interface TYPES and of TARGET
LIBRARIES/SESSIONS : PAC7ET
DSN=&OUT

```
//*********************************************************************
//* VA PAC-ENDEVOR 2.5   : UPDATING LIBRARIES/SESSIONS TABLE          *
//*********************************************************************
//$RADP.LSND PROC FILE=$FILE,         NUMBER OF PHYSICAL DATABASE
//         ROOT=$ROOT,                ROOT OF THE SYSTEM
//         ROOTX=$ROOTX,              ROOT OF 'VA PAC' SYSTEM
//         INDEX='$INDEX',            INDEX OF VSAM SYSTEM FILES
//         INDEXP='$INDEXP',          INDEX OF NON VSAM SYSTEM FILES
//         INDEXX='$INDEXX',          INDEX OF VA PAC VSAM SYSTEMFILES
//*:       VSAMCAT='$CATU',           USER VSAM CATALOG
//*:       SYSTCAT='$CATV',           SYSTEM VSAM CATALOG
//         STEPLIB='$MODB',           LIBRARY OF LOAD-MODULES
//         SORTLIB='$BIBT',           SORT LIBRARY
//         CYL=3,                     SORT WORKS SIZE
//         OUT=$OUT,                  OUTPUT CLASS
//         UWK=$UWK,                  WORK UNIT
//    PSBLIB='$PSBLIB',     PSB LIBRARY
//    DBDLIB='$DBDLIB',     DBD LIBRARY
//    RESLIB='$RESLIB',     RESLIB IMS
//    PROCLIB='$PRCLIB',    PROCLIB IMS
// BUF=40,SPIE=0,TEST=0,EXCPVR=0,RST=0,PRLD=,SRCH=0,
// CKPTID=,MON=N,LOGA=0,FMTO=T,DBRC=$DBRC,IRLM=$IRLM
//*--------------------------------------------------------------
//*
//INPUT   EXEC PGM=PTU001
//STEPLIB  DD DSN=&STEPLIB,DISP=SHR
//CARTE    DD DDNAME=SYSIN,DCB=BLKSIZE=80
//PAC7MB   DD DSN=&&LSNDMB,DISP=(,PASS),
//            UNIT=&UWK,SPACE=(TRK,(1,1),RLSE),
//            DCB=(RECFM=FB,LRECL=80,BLKSIZE=80)
//*
//PNDU10  EXEC PGM=DFSRRC00,REGION=$REGSIZ,
//        PARM=(DLI,PNDU10,PNDU10$SUG,&BUF,
//        &SPIE&TEST&EXCPVR&RST,&PRLD,
//        &SRCH,&CKPTID,&MON,&LOGA,&FMTO,,,&DBRC,&IRLM)
//STEPLIB  DD DSN=&RESLIB,DISP=SHR
//         DD DSN=&STEPLIB,DISP=SHR
//DFSRESLB DD DSN=&RESLIB,DISP=SHR
//IMS      DD DSN=&PSBLIB,DISP=SHR
//         DD DSN=&DBDLIB,DISP=SHR
//DDSNAP   DD SYSOUT=&OUT
//PROCLIB  DD DSN=&PROCLIB,DISP=SHR
//IEFRDER  DD DUMMY,
//            DCB=(RECFM=VB,BLKSIZE=1920,LRECL=1916,BUFNO=2)
//SYSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//            BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//            BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSMON   DD DUMMY
//DFSVSAMP DD DSN=&INDEXP..&ROOT.&ROOT.SY(DFSVSAM8),DISP=SHR
//SORTLIB  DD DSN=&SORTLIB,DISP=SHR
//SORTWK01 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//SORTWK02 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//SORTWK03 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//*:STEPCAT  DD DSN=&VSAMCAT,DISP=SHR
//*:         DD DSN=&SYSTCAT,DISP=SHR
//PAC7AE$SUF DD DSN=&INDEXX..&ROOTX.&ROOTX.AE,DISP=SHR
//PAC7TS   DD DSN=&INDEX..&ROOT.&FILE.TS,DISP=SHR
//PAC7MV   DD DSN=&&LSNDMB,DISP=(OLD,DELETE)
//PAC7ET   DD SYSOUT=&OUT
//SYSOUT   DD SYSOUT=&OUT
//SYSOUX   DD SYSOUT=&OUT
//SYSUDUMP DD SYSOUT=&OUT
//*
```

```
//PNDU30  EXEC PGM=DFSRRC00,REGION=$REGSIZ,
//        PARM=(DLI,PNDU30,PNDU30$SUG,&BUF,
//        &SPIE&TEST&EXCPVR&RST,&PRLD,
//        &SRCH,&CKPTID,&MON,&LOGA,&FMTO,,,&DBRC,&IRLM)
//STEPLIB  DD DSN=&RESLIB,DISP=SHR
//         DD DSN=&STEPLIB,DISP=SHR
//DFSRESLB DD DSN=&RESLIB,DISP=SHR
//IMS      DD DSN=&PSBLIB,DISP=SHR
//         DD DSN=&DBDLIB,DISP=SHR
//DDSNAP   DD SYSOUT=&OUT
//PROCLIB  DD DSN=&PROCLIB,DISP=SHR
//IEFRDER  DD DUMMY,
//            DCB=(RECFM=VB,BLKSIZE=1920,LRECL=1916,BUFNO=2)
//SYSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//            BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//            BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSMON   DD DUMMY
//DFSVSAMP DD DSN=&INDEXP..&ROOT.&ROOT.SY(DFSVSAM8),DISP=SHR
//*:STEPCAT  DD DSN=&VSAMCAT,DISP=SHR
//*:         DD DSN=&SYSTCAT,DISP=SHR
//PAC7AE$SUF DD DSN=&INDEXX..&ROOTX.&ROOTX.AE,DISP=SHR
//PAC7TS   DD DSN=&INDEX..&ROOT.&FILE.TS,DISP=SHR
//PAC7TY   DD DSN=&INDEX..&ROOT.&FILE.TY,DISP=SHR
//PAC7ET   DD SYSOUT=&OUT
//SYSOUT   DD SYSOUT=&OUT
//SYSOUX   DD SYSOUT=&OUT
//SYSPRINT DD SYSOUT=&OUT
//SYSUDUMP DD SYSOUT=&OUT
//*
```

7.9. MEND: GENERATION OF VA PAC UPDATE TRANSACTIONS

MEND: GENERATION OF VA PAC UPDATE TRANSACTIONS

The MEND procedure prepares VA Pac batch update transactions from the UQ interface journal and archives this journal. Only the transactions of the current VA Pac Database are selected.

EXECUTION CONDITION

The interface must be closed to use.

USER INPUT

. One administrator parameter line.

Pos.	Length	Value	Meaning
2	1	'*'	Line code
3	8	uuuuuuuu	User code
11	8	mmmmmmmm	Password

REPORT PRINTING

Printing of generation result.

RESULT

This procedure generates a VA Pac batch update transactions file for the UPDT procedure.

DESCRIPTION OF STEPS

This procedure comprises the following steps:

```
 . Recognition of input                           : PTU001

 . Preparation for the generation of update trans.: PNDM05

 . Generation of VA Pac update transactions       : PNDM10

 . Archiving of the UQ interface journal          : PNDM50

 . Reinitialization of the UQ journal             : PNDM60
```

RECOGNITION OF INPUT: PTU001

```
. Input data                                    : CARTE
  DSN=SYSIN

. Output file                                   : PAC7MB
  DSN=&&MENDMB
```

PREPARATION FOR THE GENERATION OF UPDATE TRANS. : PNDM05

```
. Input files:
 - VA Pac data file                             : PAC7AR
   DSN=&INDEXX..&ROOTX.&FILEX.AR
 - LIBRARIES and SESSIONS file                  : PAC7TS
   DSN=&INDEX..&ROOT.&FILE.TS
 - Interface journal file                       : PAC7UQ
   DSN=&INDEX..&ROOT.&FILE.UQ
 - Administrator identification transaction     : PAC7MV
   DSN=&INPUT

. Output file:
 - Endevor transactions file                    : PAC7UR
   DSN=&&PAC7UR
```

GENERATION OF VA PAC UPDATE TRANSACTIONS: PNDM10

```
. Input files:
 - Endevor transactions file                    : PAC7UR
   DSN=&&PAC7UR
 - VA Pac index file                            : PAC7AN
   DSN=&INDEXX..&ROOTX.&FILEX.AN
 - VA Pac data file                             : PAC7AR
   DSN=&INDEXX..&ROOTX.&FILEX.AR
 - VA Pac error message file                    : PAC7AE
   DSN=&INDEXX..&ROOTX.&ROOTX.AE

. Output file:
 - VA Pac update transactions file              : PAC7MV
   DSN=&&PAC7MV
 - Anomaly report                               : PAC7ET
   DSN=&OUT
```

JOURNAL ARCHIVING: PNDM50

```
. Input files:
 - VA Pac data  file                          : PAC7AR
   DSN=&INDEXX..&ROOTX.&FILEX.AR
 - Interface jounal file                      : PAC7UQ
   DSN=&INDEX..&ROOT.&FILE.UQ
 - Old archived journal file                  : PAC7US
   DSN=&INDEXQ..&ROOT.&FILE.QU(0)

. Output file:
 - New archived journal file                  : PAC7SU
   DSN=&INDEXQ..&ROOT.&FILE.QU(+1)
  - Transactions from other Databases         : PAC7QU
    DSN=&&PAC7QU
  - Counter of transactions from other Databases: PAC7CC
    DSN=&&PAC7CC
```

FORMATTING OF JOURNAL FILE: IDCAMS

```
. Input file:
 - Definition of journal file                 : SYSIN
   DSN=&INDEXP..&ROOT.&ROOT.SY(DF&ROOT.&FILE.UQ)
```

JOURNAL FILE REINITIALIZATION: PNDM60

```
 . Input files:
  - Counter of transactions from other Databases: PAC7CC
    DSN=&&PAC7CC
  - Transactions from other Databases         : PAC7QU
    DSN=&&PAC7QU

 . Output file:
  - Interface journal file                    : PAC7UQ
    DSN=&INDSV..&ROOT.&FILE.UQ
```

```
//*********************************************************************
//* VA PAC-ENDEVOR 2.5   : IMPOSITION TRANSACTIONS FOR UPDATING       *
//*********************************************************************
//$RADP.MEND PROC FILE=$FILE,         NUMBER OF PHYSICAL DATABASE
//         FILEX=$FILEX,              NUMBER OF VA PAC PHYSICAL DBASE
//         ROOT=$ROOT,                ROOT OF THE SYSTEM
//         ROOTX=$ROOTX,              ROOT OF 'VA PAC' SYSTEM
//         INDEX='$INDEX',            INDEX OF VSAM SYSTEM FILES
//         INDEXX='$INDEXX',          INDEX VA PAC VSAM SYSTEM FILES
//         INDEXP='$INDEXP',          INDEX OF NON VSAM SYSTEM FILES
//         INDEXQ='$INDEXQ',          INDEX OF NON VSAM USER FILES
//*:       VSAMCAT='$CATU',           USER VSAM CATALOG
//*:       SYSTCAT='$CATV',           SYSTEM VSAM CATALOG
//         STEPLIB='$MODB',           LIBRARY OF LOAD-MODULES
//         SORTLIB='$BIBT',           SORT LIBRARY
//         CYL=1,                     SORT WORKS SIZE
//         OUT=$OUT,                  OUTPUT CLASS
//         UWK=$UWK,                  WORK UNIT
//         UNITS=$UNITO,              'UQ' BACKUP UNIT
//         VOLS='SER=$VOLO',          'UQ' BACKUP VOLUME
//         SPAUR='(TRK,(02,01),RLSE)', TEMPORARY FILE SPACE
//         SPAMV='(TRK,(05,01),RLSE)', UPDATE TRANS. FILE SPACE
//         SPASU='(TRK,(10,05),RLSE)', 'UQ' BACKUP SPACE
//         SPAQU='(TRK,(10,05),RLSE)', SPACE FOR TRANS. FROM OTHER BASE
//         SPACC='(TRK,(01,01),RLSE)', SPACE FOR TRANS. COUNT ("     " )
//    PSBLIB='$PSBLIB',     PSB LIBRARY
//    DBDLIB='$DBDLIB',     DBD LIBRARY
//    RESLIB='$RESLIB',     RESLIB IMS
//    PROCLIB='$PRCLIB',    PROCLIB IMS
// BUF=40,SPIE=0,TEST=0,EXCPVR=0,RST=0,PRLD=,SRCH=0,
// CKPTID=,MON=N,LOGA=0,FMTO=T,DBRC=$DBRC,IRLM=$IRLM
//*-------------------------------------------------------------------*
//*
//INPUT   EXEC PGM=PTU001
//STEPLIB  DD DSN=&STEPLIB,DISP=SHR
//CARTE    DD DDNAME=SYSIN,DCB=BLKSIZE=80
//PAC7MB   DD DSN=&&MENDMB,DISP=(,PASS),
//            UNIT=&UWK,SPACE=(TRK,(1,1),RLSE),
//            DCB=(RECFM=FB,LRECL=80,BLKSIZE=80)
//*
//PNDM05  EXEC PGM=DFSRRC00,REGION=$REGSIZ,
//        PARM=(DLI,PNDM05,PNDM05$SUG,&BUF,
//        &SPIE&TEST&EXCPVR&RST,&PRLD,
//        &SRCH,&CKPTID,&MON,&LOGA,&FMTO,,,&DBRC,&IRLM)
//STEPLIB  DD DSN=&RESLIB,DISP=SHR
//         DD DSN=&STEPLIB,DISP=SHR
//DFSRESLB DD DSN=&RESLIB,DISP=SHR
//IMS      DD DSN=&PSBLIB,DISP=SHR
//         DD DSN=&DBDLIB,DISP=SHR
//DDSNAP   DD SYSOUT=&OUT
//PROCLIB  DD DSN=&PROCLIB,DISP=SHR
//IEFRDER  DD DUMMY,
//            DCB=(RECFM=VB,BLKSIZE=1920,LRECL=1916,BUFNO=2)
//SYSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//            BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//            BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSMON   DD DUMMY
//DFSVSAMP DD DSN=&INDEXP..&ROOT.&ROOT.SY(DFSVSAM8),DISP=SHR
//SORTLIB  DD DSN=&SORTLIB,DISP=SHR
//SORTWK01 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//SORTWK02 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//SORTWK03 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//*:STEPCAT  DD DSN=&VSAMCAT,DISP=SHR
```

```
//*:         DD DSN=&SYSTCAT,DISP=SHR
//PAC7AR$SUF DD DSN=&INDEXX..&ROOTX.&FILEX.AR,DISP=SHR
//PAC7TS   DD DSN=&INDEX..&ROOT.&FILE.TS,DISP=SHR
//PAC7UQ   DD DSN=&INDEX..&ROOT.&FILE.UQ,DISP=SHR
//PAC7MV   DD DSN=&&MENDMB,DISP=(OLD,DELETE)
//PAC7UR   DD DSN=&&PAC7UR,DISP=(,PASS),UNIT=&UWK,
//            DCB=(RECFM=FB,LRECL=183,BLKSIZE=18300),
//            SPACE=&SPAUR
//SYSOUT   DD SYSOUT=&OUT
//SYSOUX   DD SYSOUT=&OUT
//SYSPRINT DD SYSOUT=&OUT
//SYSUDUMP DD SYSOUT=&OUT
//*
//PNDM10  EXEC PGM=DFSRRC00,REGION=$REGSIZ,
//        COND=(00,NE,PNDM05),
//        PARM=(DLI,PNDM10,PNDM10$SUG,&BUF,
//        &SPIE&TEST&EXCPVR&RST,&PRLD,
//        &SRCH,&CKPTID,&MON,&LOGA,&FMTO,,,&DBRC,&IRLM)
//STEPLIB  DD DSN=&RESLIB,DISP=SHR
//         DD DSN=&STEPLIB,DISP=SHR
//         DD DSN=$BCOB,DISP=SHR
//DFSRESLB DD DSN=&RESLIB,DISP=SHR
//IMS      DD DSN=&PSBLIB,DISP=SHR
//         DD DSN=&DBDLIB,DISP=SHR
//DDSNAP   DD SYSOUT=&OUT
//PROCLIB  DD DSN=&PROCLIB,DISP=SHR
//IEFRDER  DD DUMMY,
//            DCB=(RECFM=VB,BLKSIZE=1920,LRECL=1916,BUFNO=2)
//SYSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//            BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//            BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSMON   DD DUMMY
//DFSVSAMP DD DSN=&INDEXP..&ROOT.&ROOT.SY(DFSVSAM8),DISP=SHR
//*:STEPCAT  DD DSN=&VSAMCAT,DISP=SHR
//*:         DD DSN=&SYSTCAT,DISP=SHR
//PAC7AE$SUF DD DSN=&INDEXX..&ROOTX.&ROOTX.AE,DISP=SHR
//PAC7AN$SUF DD DSN=&INDEXX..&ROOTX.&FILEX.AN,DISP=SHR
//PAC7AR$SUF DD DSN=&INDEXX..&ROOTX.&FILEX.AR,DISP=SHR
//PAC7PA$SUF DD DSN=&INDEXX..&ROOTX.&ROOTX.PA,DISP=SHR
//PAC7P1$SUF DD DSN=&INDEXX..&ROOTX.&ROOTX.P1,DISP=SHR
//PAC7UR   DD DSN=&&PAC7UR,DISP=(OLD,DELETE)
//PAC7MV   DD DSN=&&PAC7MV,DISP=(,PASS),UNIT=&UWK,
//            DCB=(RECFM=FB,LRECL=80,BLKSIZE=8800),
//            SPACE=&SPAMV
//PAC7ET   DD SYSOUT=&OUT
//SYSOUT   DD SYSOUT=&OUT
//SYSOUX   DD SYSOUT=&OUT
//SYSPRINT DD SYSOUT=&OUT
//SYSUDUMP DD SYSOUT=&OUT
//*
//PNDM50  EXEC PGM=DFSRRC00,REGION=$REGSIZ,
//        COND=((00,NE,PNDM05),(00,NE,PNDM10)),
//        PARM=(DLI,PNDM50,PNDM50$SUG,&BUF,
//        &SPIE&TEST&EXCPVR&RST,&PRLD,
//        &SRCH,&CKPTID,&MON,&LOGA,&FMTO,,,&DBRC,&IRLM)
//STEPLIB  DD DSN=&RESLIB,DISP=SHR
//         DD DSN=&STEPLIB,DISP=SHR
//DFSRESLB DD DSN=&RESLIB,DISP=SHR
//IMS      DD DSN=&PSBLIB,DISP=SHR
//         DD DSN=&DBDLIB,DISP=SHR
//DDSNAP   DD SYSOUT=&OUT
//PROCLIB  DD DSN=&PROCLIB,DISP=SHR
//IEFRDER  DD DUMMY,
//            DCB=(RECFM=VB,BLKSIZE=1920,LRECL=1916,BUFNO=2)
//SYSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//            BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//            BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSMON   DD DUMMY
//DFSVSAMP DD DSN=&INDEXP..&ROOT.&ROOT.SY(DFSVSAM8),DISP=SHR
//*:STEPCAT  DD DSN=&VSAMCAT,DISP=SHR
//*:         DD DSN=&SYSTCAT,DISP=SHR
//PAC7AR$SUF DD DSN=&INDEXX..&ROOTX.&FILEX.AR,DISP=SHR
//PAC7US   DD DSN=&INDEXQ..&ROOT.&FILE.QU(0),DISP=SHR
//PAC7UQ   DD DSN=&INDEX..&ROOT.&FILE.UQ,DISP=SHR
//PAC7SU   DD DSN=&INDEXQ..&ROOT.&FILE.QU(+1),DISP=(,CATLG,DELETE),
//            VOL=&VOLS,
//            UNIT=&UNITS,SPACE=&SPASU,
//            DCB=&INDEXQ..DSCB.&ROOT.&FILE.QU
//PAC7QU   DD DSN=&&PAC7QU,DISP=(,PASS),UNIT=&UWK,
//            DCB=(RECFM=FB,LRECL=170,BLKSIZE=1700),
//            SPACE=&SPAQU
//PAC7CC   DD DSN=&&PAC7CC,DISP=(,PASS),UNIT=&UWK,
//            DCB=(RECFM=FB,LRECL=170,BLKSIZE=170),
//            SPACE=&SPACC
//SYSOUT   DD SYSOUT=&OUT
//SYSOUX   DD SYSOUT=&OUT
//SYSUDUMP DD SYSOUT=&OUT
//*
//DEFINUQ EXEC PGM=IDCAMS,COND=((00,NE,PNDM05),(00,NE,PNDM10))
//*:STEPCAT  DD DSN=&VSAMCAT,DISP=SHR
//SYSPRINT DD SYSOUT=&OUT
//SYSIN    DD DSN=&INDEXP..&ROOT.&ROOT.SY(DF&ROOT.&FILE.UQ),DISP=SHR
//*
//PNDM60  EXEC PGM=PNDM60,COND=((00,NE,PNDM05),(00,NE,PNDM10))
//STEPLIB  DD DSN=&STEPLIB,DISP=SHR
//PAC7QU   DD DSN=&&PAC7QU,DISP=(OLD,DELETE)
//PAC7CC   DD DSN=&&PAC7CC,DISP=(OLD,DELETE)
//PAC7UQ   DD DSN=&INDEX..&ROOT.&FILE.UQ,DISP=SHR
//SYSOUT   DD SYSOUT=*
//SYSUDUMP DD SYSOUT=*
//*
```

7.10. REND: RETRIEVAL OF VA PAC U.E. 1.6 -> 2.5

REND : RETRIEVAL OF VA PAC U.E. 1.6 --> 2.5

The REND procedure retrieves all the $7N occurrences of the .NDVLM User Entity in release 1.6 and applies the new formats to dates:

1. last action	CCYY/MM/DD instead of YY/MM/DD
1. current action	CCYY/MM/DD instead of YY/MM/DD
1. element generation	DD/MM/CCYY instead of DD/MM/YY

EXECUTION CONDITIONS

The VA Pac VINS procedure must be executed first.

The VA Pac AR data file must be closed to on-line use.

```
DESCRIPTION OF STEPS

This procedure comprises the following step:

 . U.E. update in VA Pac Database            : PNRPEU

. Input-output file:
 - VA Pac data file                          : PAC7AR
   DSN=&INDEXX..&ROOTX.&FILEX.AR

. Output display:
 - Counter of updated occurrences
```

```
//***************************************************************
//* VA PAC-ENDEVOR 2.5   : USER ENTITIES RETRIEVAL
//***************************************************************
//$RADP.REND PROC ROOTX=$ROOTX,      ROOT OF 'VA PAC' SYSTEM
//         FILEX=$FILEX,             NUMBER OF VA PAC DATABASE
//         ROOT=$ROOT,               SYSTEM ROOT
//         INDEXX='$INDEXX',         INDEX OF VA PAC VSAM SYSTEM FILES
//         INDEXP='$INDEXP',         INDEX OF NON VSAM SYSTEM FILLES
//*:       SYSTCAT='$CATV',           SYSTEM VSAM CATALOG
//         STEPLIB='$MODB',           LIBRARY OF LOAD-MODULES
//         OUT=$OUT,                  OUTPUT CLASS
//    PSBLIB='$PSBLIB',     PSB LIBRARY
//    DBDLIB='$DBDLIB',     DBD LIBRARY
//    RESLIB='$RESLIB',     RESLIB IMS
//    PROCLIB='$PRCLIB',    PROCLIB IMS
// BUF=40,SPIE=0,TEST=0,EXCPVR=0,RST=0,PRLD=,SRCH=0,
// CKPTID=,MON=N,LOGA=0,FMTO=T,DBRC=$DBRC,IRLM=$IRLM
//***************************************************************
//PNRPEU  EXEC PGM=DFSRRC00,REGION=$REGSIZ,
//        PARM=(DLI,PNRPEU,PNRPEU$SUG,&BUF,
//        &SPIE&TEST&EXCPVR&RST,&PRLD,
//        &SRCH,&CKPTID,&MON,&LOGA,&FMTO,,,&DBRC,&IRLM)
//STEPLIB  DD DSN=&RESLIB,DISP=SHR
//         DD DSN=&STEPLIB,DISP=SHR
//DFSRESLB DD DSN=&RESLIB,DISP=SHR
//IMS      DD DSN=&PSBLIB,DISP=SHR
//         DD DSN=&DBDLIB,DISP=SHR
//DDSNAP   DD SYSOUT=&OUT
//PROCLIB  DD DSN=&PROCLIB,DISP=SHR
//IEFRDER  DD DUMMY,
//            DCB=(RECFM=VB,BLKSIZE=1920,LRECL=1916,BUFNO=2)
//SYSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//            BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//            BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSMON   DD DUMMY
//DFSVSAMP DD DSN=&INDEXP..&ROOT.&ROOT.SY(DFSVSAM8),DISP=SHR
//*:STEPCAT  DD DSN=&VSAMCAT,DISP=SHR
//*:         DD DSN=&SYSTCAT,DISP=SHR
//PAC7AR$SUF DD DSN=&INDEXX..&ROOTX.&FILEX.AR,DISP=SHR
//SYSOUT   DD SYSOUT=&OUT
//SYSOUX   DD SYSOUT=&OUT
//SYSUDUMP DD SYSOUT=&OUT
//SYSPRINT DD SYSOUT=&OUT
//*
```

7.11. RIND: INFOPAC ELT. CREATION (EXISTING RETRIEVAL)

RIND: INFOPAC ELEMENT CREATION

The RIND procedure allows the user to create, in ENDEVOR, the INFOPAC ELEMENTS which correspond to the VA Pac entities created in ENDEVOR before the installation of the VA Pac-ENDEVOR interface on the site.

EXECUTION CONDITION

This procedure is prepared by the 'RRND: Existing Data Retrieval' procedure.

DESCRIPTION OF STEPS

This procedure comprises the following steps:

```
 . Creation of temporary PDS for INFOPAC elts.   : IEFBR14

 . Loading of INFOPAC elements in the PDS        : IEBUPDTE

 . Creation (ADD) of INFOPAC elements in ENDEVOR : NDVRC1
```

CREATION OF A TEMPORARY PDS FOR INOFPAC ELTS.: IEFBR14

```
. Output file:
 - PDS containing INFOPAC elements           : DD1
   DSN=&INDEXQ..&ROOT.&FILE.BB
```

LOADING INFOPAC ELEMENTS IN THE PDS: IEBUPDTE

```
. Input file:
 - 'INFOPAC' contents of each element        : SYSIN
   ./ add name=Innnnnnn

. Input/Output file:
 - PDS containing INFOPAC elements           : SYSUT1/SYSUT2
   DSN=&INDEXQ..&ROOT.&FILE.BB
```

CREATION OF INFOPAC ELEMENTS IN ENDEVOR: NDVRC1

```
. Input files:
 - TYPES file                                : PAC7TY
   DSN=&INDEX..&ROOT.&FILE.TY
 - PDS containing INFOPAC elements           : INFOPAC
   DSN=*.STEP01.DD1
 - Endevor ADD query of INFOPAC elements     : BSTIPT01

. Input/Output files:
 - Interface work file                       : PAC7PU
   DSN=&INDEX..&ROOT.&FILE.UP
 - Interface work file                       : PAC7UP
   DSN=&INDEX..&ROOT.&FILE.UP

. Output file:
 - Interface journal file                    : PAC7UQ
   DSN=&INDEX..&ROOT.&FILE.UQ
```

```
//*********************************************************************
//* VA PAC-ENDEVOR 2.5   : CREATE 'INFOPAC'     (RETRIEVAL ELEME NTS) *
//*********************************************************************
//$RADP.RIND PROC FILE=$FILE,         NUMBER OF PHYSICAL DATABASE
//         ROOT=$ROOT,                ROOT OF THE SYSTEM
//         INDEX='$INDEX',            INDEX OF VSAM SYSTEM FILES
//         INDEXQ='$INDEXQ',          INDEX OF NON VSAM USER FIL ES
//*:       VSAMCAT='$CATU',           USER VSAM CATALOG
//         LOADLIB='$LDLIB',          ENDEVOR LOADLIB
//         CONLIB='$CONLIB',          ENDEVOR CONLIB
//         OUT=$OUT,                  OUTPUT CLASS
//         VOLS='SER=$VOLO',          NON VSAM USER FILES VOLUME
//         UNITS=$UNITO,              NON VSAM USER FILES UNIT
//         SPABB='(TRK,(20,10,20),RLSE)' SPACE OF TEMPORARY PDS
//*-------------------------------------------------------------------*
//*
//**********************************************
//* CREATE TEMPORARY PDS FILE                  *
//**********************************************
//*
//STEP01  EXEC PGM=IEFBR14
//DD1      DD DSN=&INDEXQ..&ROOT.&FILE.BB,DISP=(,CATLG,DELETE),
//            UNIT=&UNITS,
//            VOL=&VOLS,
//            DCB=(RECFM=FB,LRECL=80,BLKSIZE=80,DSORG=PO),
//            SPACE=&SPABB
//*
//**********************************************
//* LOADING 'INFOPAC' ELEMENTS                 *
//**********************************************
//*
//STEP02  EXEC PGM=IEBUPDTE,PARM=NEW
//SYSPRINT DD SYSOUT=&OUT
//SYSUT1   DD DSN=*.STEP01.DD1,DISP=(SHR,KEEP,DELETE)
//SYSUT2   DD DSN=*.STEP01.DD1,DISP=(SHR,KEEP,DELETE)
//SYSIN    DD DUMMY
//*
//**********************************************
//* ENDEVOR 'ADD' ACTION FOR 'INFOPAC' ELEMENT *
//**********************************************
//*
//STEP03  EXEC PGM=NDVRC1,DYNAMNBR=1500,REGION=4096K,PARM='C1BM3000'
//STEPLIB  DD DSN=&LOADLIB,DISP=SHR
//CONLIB   DD DSN=&CONLIB,DISP=SHR
//*:STEPCAT  DD DSN=&VSAMCAT,DISP=SHR
//SYSPRINT DD SYSOUT=&OUT
//SYSUDUMP DD SYSOUT=&OUT
//SORTWK01 DD UNIT=SYSDA,SPACE=(CYL,(2,1))
//SORTWK02 DD UNIT=SYSDA,SPACE=(CYL,(2,1))
//SORTWK03 DD UNIT=SYSDA,SPACE=(CYL,(2,1))
//SORTWK04 DD UNIT=SYSDA,SPACE=(CYL,(2,1))
//C1TPDD01 DD UNIT=SYSDA,SPACE=(CYL,5),
//            DCB=(RECFM=VB,LRECL=260,BLKSIZE=6160)
//C1TPDD02 DD UNIT=SYSDA,SPACE=(CYL,5),
//            DCB=(RECFM=VB,LRECL=260,BLKSIZE=6160)
//C1TPLSIN DD UNIT=SYSDA,SPACE=(CYL,5),
//            DCB=(RECFM=FB,LRECL=80,BLKSIZE=6160)
//C1TPLSOU DD UNIT=SYSDA,SPACE=(CYL,5)
//PAC7TY   DD DSN=&INDEX..&ROOT.&FILE.TY,DISP=SHR
//PAC7PU   DD DSN=&INDEX..&ROOT.&FILE.UP,DISP=SHR
//PAC7UP   DD DSN=&INDEX..&ROOT.&FILE.UP,DISP=SHR
//PAC7UQ   DD DSN=&INDEX..&ROOT.&FILE.UQ,DISP=SHR
//INFOPAC  DD DSN=*.STEP01.DD1,DISP=(SHR,DELETE,DELETE)
//C1PLMSGS DD SYSOUT=&OUT
```

```
//C1MSGS1  DD SYSOUT=&OUT
//C1PRINT  DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,BLKSIZE=6171)
//PTRACE   DD SYSOUT=&OUT
//SYSABEND DD SYSOUT=&OUT
//SYSOUT   DD SYSOUT=&OUT
//BSTIPT01 DD DUMMY
//*
```

7.12. RPND: VA PAC ELT. CREATION (EXISTING RETRIEVAL)

RPND: CREATION OF VA Pac ELEMENT

The RPND procedure prepares the VA Pac batch update transactions from the 'UQ' interface journal prepared by the RRND procedure.

EXECUTION CONDITION

The RRND procedure must be executed first.

USER INPUT

. One administrator identification line.

Pos.	Length	Value	Meaning
2	1	'*'	Line code
3	8	uuuuuuuu	User code
11	8	mmmmmmmm	Password

REPORT PRINTING

Printing of generation result.

RESULT

This procedure generates a VA Pac batch update transaction file for the UPDT procedure.

DESCRIPTION OF STEPS

This procedure comprises the following steps:

. Recognition of input : PTU001

. Preparation for generation of update trans. : PNDM05

. Generation of VA Pac update trans. (EU def.) : PNDR40

. Generation of VA Pac update trans. (-D1 -D9) : PNDM10

. Redefinition of UQ journal file : DEFINE

. Reinitialization of UQ journal file : PNINUQ

```
RECOGNITION OF INPUT: PTU001

. Input data                                   : CARTE
  DSN=SYSIN

. Output file                                  : PAC7MB
  DSN=&&RPNDMB


PREPARATION FOR GENERATION OF UPDATE TRANSACTIONS: PNDM05

. Input files:
 - LIBRARIES and SESSIONS file                 : PAC7TS
   DSN=&INDEX..&ROOT.&FILE.TS
 - Interface journal file                      : PAC7UQ
   DSN=&INDEX..&ROOT.&FILE.UQ
 - Administrator identification transaction   : PAC7MV
   DSN=&&RPNDMB
 - VA Pac data file                            : PAC7AR
   DSN=&INDEXX..&ROOTX.&FILEX.AR

. Output file:
 - ENDEVOR transactions file                   : PAC7UR
   DSN=&&PAC7UR

. Sort files                                   : SORTWK01
                                               : SORTWK02
                                               : SORTWK03
```

```
GENERATION OF VA Pac UPDATE TRANS. (EU def.): PNDR40

. Input file:
 - Endevor transactions file                 : PAC7LT
   DSN=&&PAC7UR

. Output file:
 - VA Pac updating transactions file         : PAC7MU
   DSN=&&PAC7MU


GENERATION OF VA Pac UPDATE TRANS. (-D1 -D9): PNDM10

. Input files:
 - Endevor transactions file                 : PAC7UR
   DSN=&&PAC7UR
 - VA Pac index file                         : PAC7AN
   DSN=&INDEXX..&ROOTX.&FILEX.AN
 - VA Pac data file                          : PAC7AR
   DSN=&INDEXX..&ROOTX.&FILEX.AR
 - VA Pac error messages file                : PAC7AE
   DSN=&INDEXX..&ROOTX.&ROOTX.AE

. Output file:
 - VA Pac update transactions file           : PAC7MV
   DSN=&&PAC7MU --> DISP=(MOD,PASS))
 - Anomaly report                            : PAC7ET
   DSN=&OUT


REDEFINITION OF 'UG' JOURNAL: IDCAMS

. Input file:
 - Definition of journal file                : SYSIN
   DSN=&INDEXP..&ROOT.&ROOT.SY(DF&ROOT.&FILE.UQ)



REINITIALIZATION OF 'UQ' JOURNAL: PNINUQ

. Output file:
 - Interface journal file                    : PAC7UQ
   DSN=&INDEX..&ROOT.&FILE.UQ
```

```
//**********************************************************************
//* VA PAC-ENDEVOR 2.5   : CREATE 'VA PAC'   (RETRIEVING OF ELEMENTS) *
//**********************************************************************
//$RADP.RPND PROC FILE=$FILE,         NUMBER OF PHYSICAL DATABASE
//         FILEX=$FILEX,              NUMBER OF VA PAC PHYSICAL DBASE
//         ROOT=$ROOT,                ROOT OF THE SYSTEM
//         ROOTX=$ROOTX,              ROOT OF 'VA PAC' SYSTEM
//         INDEX='$INDEX',            INDEX OF VSAM SYSTEM FILES
//         INDEXX='$INDEXX',          INDEX OF VA PAC VSAM SYSTEM FILES
//         INDEXP='$INDEXP',          INDEX OF SYSTEM NON VSAM FI
//*:       VSAMCAT='$CATU',           USER VSAM CATALOG
//*:       SYSTCAT='$CATV',           SYSTEM VSAM CATALOG
//         STEPLIB='$MODB',           LIBRARY OF LOAD-MODULES
//         SORTLIB='$BIBT',           SORT LIBRARY
//         CYL=1,                     SORT WORKS SIZE
//         OUT=$OUT,                  OUTPUT CLASS
//         UWK=$UWK,                  WORK UNIT
//         SPAUR='(TRK,(02,01),RLSE)', SPACE OF TEMPORARY FILE
//         SPAMV='(TRK,(05,01),RLSE)', SPACE OF UPDATE TRANS. FILE
//    PSBLIB='$PSBLIB',     PSB LIBRARY
//    DBDLIB='$DBDLIB',     DBD LIBRARY
//    RESLIB='$RESLIB',     RESLIB IMS
//    PROCLIB='$PRCLIB',    PROCLIB IMS
// BUF=40,SPIE=0,TEST=0,EXCPVR=0,RST=0,PRLD=,SRCH=0,
// CKPTID=,MON=N,LOGA=0,FMTO=T,DBRC=$DBRC,IRLM=$IRLM
//*--------------------------------------------------------------------*
//*
//INPUT   EXEC PGM=PTU001
//STEPLIB  DD DSN=&STEPLIB,DISP=SHR
//CARTE    DD DDNAME=SYSIN,DCB=BLKSIZE=80
//PAC7MB   DD DSN=&&RPNDMB,DISP=(,PASS),
//            UNIT=&UWK,SPACE=(TRK,(1,1),RLSE),
//            DCB=(RECFM=FB,LRECL=80,BLKSIZE=80)
//*
//PNDM05  EXEC PGM=DFSRRC00,REGION=$REGSIZ,
//        PARM=(DLI,PNDM05,PNDM05$SUG,&BUF,
//        &SPIE&TEST&EXCPVR&RST,&PRLD,
//        &SRCH,&CKPTID,&MON,&LOGA,&FMTO,,,&DBRC,&IRLM)
//STEPLIB  DD DSN=&RESLIB,DISP=SHR
//         DD DSN=&STEPLIB,DISP=SHR
//DFSRESLB DD DSN=&RESLIB,DISP=SHR
//IMS      DD DSN=&PSBLIB,DISP=SHR
//         DD DSN=&DBDLIB,DISP=SHR
//DDSNAP   DD SYSOUT=&OUT
//PROCLIB  DD DSN=&PROCLIB,DISP=SHR
//IEFRDER  DD DUMMY,
//            DCB=(RECFM=VB,BLKSIZE=1920,LRECL=1916,BUFNO=2)
//SYSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//            BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//            BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSMON   DD DUMMY
//DFSVSAMP DD DSN=&INDEXP..&ROOT.&ROOT.SY(DFSVSAM8),DISP=SHR
//SORTLIB  DD DSN=&SORTLIB,DISP=SHR
//SORTWK01 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//SORTWK02 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//SORTWK03 DD UNIT=&UWK,SPACE=(CYL,&CYL,,CONTIG)
//*:STEPCAT  DD DSN=&VSAMCAT,DISP=SHR
//*:         DD DSN=&SYSTCAT,DISP=SHR
//PAC7AR$SUF DD DSN=&INDEXX..&ROOTX.&FILEX.AR,DISP=SHR
//PAC7TS   DD DSN=&INDEX..&ROOT.&FILE.TS,DISP=SHR
//PAC7UQ   DD DSN=&INDEX..&ROOT.&FILE.UQ,DISP=SHR
//PAC7MV   DD DSN=&&RPNDMB,DISP=(OLD,DELETE)
//PAC7UR   DD DSN=&&PAC7UR,DISP=(,PASS),UNIT=&UWK,
//            DCB=(RECFM=FB,LRECL=183,BLKSIZE=18300),
//            SPACE=&SPAUR
//SYSOUT   DD SYSOUT=&OUT
//SYSOUX   DD SYSOUT=&OUT
//SYSPRINT DD SYSOUT=&OUT
//SYSUDUMP DD SYSOUT=&OUT
//*
//PNDR40  EXEC PGM=PNDR40,COND=(00,NE,PNDM05)
//STEPLIB  DD DSN=&STEPLIB,DISP=SHR
//*:STEPCAT  DD DSN=&VSAMCAT,DISP=SHR
//*:         DD DSN=&SYSTCAT,DISP=SHR
//PAC7LT   DD DSN=&&PAC7UR,DISP=(OLD,PASS)
//PAC7MU   DD DSN=&&PAC7MU,DISP=(,PASS),UNIT=&UWK,
//            DCB=(RECFM=FB,LRECL=80,BLKSIZE=8800),
//            SPACE=&SPAMV
//SYSOUT   DD SYSOUT=&OUT
//SYSOUX   DD SYSOUT=&OUT
//SYSPRINT DD SYSOUT=&OUT
//SYSUDUMP DD SYSOUT=&OUT
//*
//PNDM10  EXEC PGM=DFSRRC00,REGION=$REGSIZ,
//        COND=(00,NE,PNDM05),
//        PARM=(DLI,PNDM10,PNDM10$SUG,&BUF,
//        &SPIE&TEST&EXCPVR&RST,&PRLD,
//        &SRCH,&CKPTID,&MON,&LOGA,&FMTO,,,&DBRC,&IRLM)
//STEPLIB  DD DSN=&RESLIB,DISP=SHR
//         DD DSN=&STEPLIB,DISP=SHR
//         DD DSN=$BCOB,DISP=SHR
//DFSRESLB DD DSN=&RESLIB,DISP=SHR
//IMS      DD DSN=&PSBLIB,DISP=SHR
//         DD DSN=&DBDLIB,DISP=SHR
//DDSNAP   DD SYSOUT=&OUT
//PROCLIB  DD DSN=&PROCLIB,DISP=SHR
//IEFRDER  DD DUMMY,
//            DCB=(RECFM=VB,BLKSIZE=1920,LRECL=1916,BUFNO=2)
//SYSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//            BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//            BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSMON   DD DUMMY
//DFSVSAMP DD DSN=&INDEXP..&ROOT.&ROOT.SY(DFSVSAM8),DISP=SHR
//*:STEPCAT  DD DSN=&VSAMCAT,DISP=SHR
//*:         DD DSN=&SYSTCAT,DISP=SHR
//PAC7AE$SUF DD DSN=&INDEXX..&ROOTX.&ROOTX.AE,DISP=SHR
//PAC7AN$SUF DD DSN=&INDEXX..&ROOTX.&FILEX.AN,DISP=SHR
//PAC7AR$SUF DD DSN=&INDEXX..&ROOTX.&FILEX.AR,DISP=SHR
//PAC7PA$SUF DD DSN=&INDEXX..&ROOTX.&ROOTX.PA,DISP=SHR
//PAC7P1$SUF DD DSN=&INDEXX..&ROOTX.&ROOTX.P1,DISP=SHR
//PAC7UR   DD DSN=&&PAC7UR,DISP=(OLD,PASS)
//PAC7MV   DD DSN=&&PAC7MU,DISP=(MOD,PASS)
//PAC7ET   DD SYSOUT=&OUT
//SYSOUT   DD SYSOUT=&OUT
//SYSOUX   DD SYSOUT=&OUT
//SYSPRINT DD SYSOUT=&OUT
//SYSUDUMP DD SYSOUT=&OUT
//*
//*
//DEFINE  EXEC PGM=IDCAMS,COND=(00,NE,PNDM05)
//*:STEPCAT  DD DSN=&VSAMCAT,DISP=SHR
//SYSPRINT DD SYSOUT=&OUT
//SYSIN    DD DSN=&INDEXP..&ROOT.&ROOT.SY(DF&ROOT.&FILE.UQ),DISP=SHR
//*
//PNINUQ  EXEC PGM=PNINUQ,COND=(00,NE,PNDM05)
//STEPLIB  DD DSN=&STEPLIB,DISP=SHR
```

```
//PAC7UQ   DD DSN=&INDEX..&ROOT.&FILE.UQ,DISP=SHR
//SYSOUT   DD SYSOUT=*
//SYSUDUMP DD SYSOUT=*
//*
```

7.13. RRND: RETRIEVAL OF ENDEVOR ELEMENTS

RRND: EXISTING DATA RETRIEVAL

The RRND procedure retrieves ENDEVOR ELEMENTS coming from VA Pac and generated before the acquisition of VA Pac-Endevor interface, and associates an INFOPAC ELEMENT type with each one.

EXECUTION CONDITION

None.

RESULT

Generated execution JCL for the RIND procedure (creation of INFOPAC-type elements).

'UQ' VSAM file which contains the update transactions for VA Pac. This file is retrieved as it is by the RPND procedure.

IMPORTANT NOTE

The execution of the RRND procedure MUST be followed by the execution of the RIND and RPND procedures.

USER INPUT

You can enter as many lines as there are environments which can possibly contain one or more ELEMENTS coming from the VA Pac system.

ENVIRONMENT PARAMETER LINE

Pos.	Length	Value	Meaning
1	10	'xxxxxxxxxx'	ENDEVOR Element (1)
11	8	'eeeeeeee'	ENDEVOR Environment
19	8	'ssssssss'	ENDEVOR System (1)
27	8	'subssubs'	ENDEVOR Sub-system (1)
35	8	'tttttttt'	ENDEVOR Type (1)
43	1	's'	ENDEVOR Stage (1)

(1) Each field can be entered in a generic manner using an asterisk immediately after the last significant character (ex: 'env*   '). The value '   ' is equivalent to the value '*   '.

DESCRIPTION OF STEPS

This procedure comprises the following steps:

. Recognition of user input	: PTU001
. Preparation of the elements 'LIST' SYSIN	: PNCI10
. List of ENDEVOR elements	: NDVRC1A
. Preparation of the elements 'PRINT' SYSIN	: PNDR20
. Printing of ENDEVOR elements	: NDVRC1B
. Creation of 'INFOPAC' from COBOL source	: PNDR30
. Redefinition of 'UQ' journal file	: DEFINE
. Reinitialization of 'UQ' journal file	: PNINUQ
. 'UQ' loading for VA Pac update	: PNDR35

RECOGNITION OF INPUT: PTU001

```
. Input data                              : CARTE
  DDNAME=SYSIN

. Output file                             : PAC7MB
  DSN=&&RRNDMB
```

PREPARATION OF THE ELEMENTS 'LIST' SYSIN: PNCI10

This program prepares the extraction of the COBOL-type ENDEVOR elements generated by VA Pac and identified by the working constants: 'CONSTANTES-PAC' or 'PACBASE-CONSTANT'.

```
. Input files:
 - User input: elements to be extracted       : PAC7IN
   DSN=&&RRNDMB
 - Interface type files                       : PAC7TY
   DSN=&INDEX..&ROOT.&FILE.TY

. Fichier en sortie :
 - 'LIST' SYSIN of the elements to be selected: PAC7BS
   DSN=&&PAC7BS
```

LIST OF SELECTED ENDEVOR ELEMENTS: NDVRC1A

```
. Input file:
 - 'LIST' SYSIN of elements to be selected   : BSTIPT01
   DSN=&&PAC7BS

. Output file:
 - SYSOUT of the ENDEVOR list                : C1MSGS1
   DSN=&&C1MSGS1A
```

PREPARATION FOR THE ELEMENTS 'PRINT' SYSIN: PNDR20

```
. Input file:
 - SYSOUT of the ENDEVOR list                : PAC7LT
   DSN=&&C1MSGS1A

. Output file:
 - 'PRINT' SYSIN of listed elements          : PAC7BS
   DSN=&&PAC7SB
 - 'PRINT' SYSIN of listed elements (M. opt) : PAC7BT
   DSN=&&PAC7TB
 - 'PRINT' SYSIN of elements descriptions    : PAC7XT
   DSN=&&PAC7XT
```

PRINTING OF SELECTED ENDEVOR ELEMENTS: NDVRC1B

```
. Input file:
 - 'PRINT' SYSIN of listed elements          : BSTIPT01
   DSN=&&PAC7SB

. Output files:
 - SYSOUT of the ENDEVOR list                : C1MSGS1
   DSN=&&C1MSGS1B
 - SYSOUT of ENDEVOR print                   : C1PRINT
   DSN=&&C1PRINTB
```

PRINTING OF SELECTED ENDEVOR ELEMENTS: NDVRC1C

With the 'MASTER' option

```
. Input file:
 - 'PRINT' SYSIN of listed elements          : BSTIPT01
   DSN=&&PAC7TB

. Output file:
 - SYSOUT of the ENDEVOR print (MASTER opt.) : C1PRINT
   DSN=&&C1PRINTC
```

CREATION OF 'INFOPAC' FROM THE COBOL SOURCE: PNDR30

```
. Input files:
 - SYSOUT of the ENDEVOR list                : PAC7LT
   DSN=&&C1MSGS1B
 - SYSOUT of ENDEVOR print                   : PAC7PT
   DSN=&&C1PRINTB
 - Interface of TYPES file                   : PAC7TY
   DSN=&INDEX..&ROOT.&FILE.TY
 - 'PRINT' SYSIN of elements descriptions    : PAC7XT
   DSN=&&PAC7XT

. Temporary work file                        : PAC7JD

. Output files:
 - execution JCL of the 'RIND' procedure     : PAC7JC
   DSN=&&PAC7JC
 - Sequential image of 'UQ' file             : PAC7UQ
   DSN=&&PAC7UQ
```

REDEFINITION OF 'UQ' JOURNAL: IDCAMS

```
. Input file:
 - Definition of journal file                 : SYSIN
   DSN=&INDEXP..&ROOT.&ROOT.SY(DF&ROOT.&FILE.UQ)
```

REINITIALIZATION OF 'UQ' JOURNAL: PNINUQ

```
. Output file:
 - Interface journal file                     : PAC7UQ
   DSN=&INDEX..&ROOT.&FILE.UQ
```

'UQ' LOADING FOR VA PAC UPDATE: PNDR35

```
. Input files:
 - SYSOUT of ENDEVOR print (MASTER option)    : PAC7PT
   DSN=&&C1PRINTC
 - Sequential image of 'UQ' file             : PAC7QU
   DSN=&&PAC7UQ

. Output file:
 - 'UQ' VSAM file ready for RPND procedure   : PAC7UQ
   (creation of elements in VA Pac)
   DSN=&INDEX..&ROOT.&FILE.UQ
```

```
//*********************************************************************
//* VA PAC-ENDEVOR 2.5   : RETRIEVAL OF ENDEVOR ELEMENTS              *
//*********************************************************************
//$RADP.RRND PROC FILE=$FILE,         NUMBER OF PHYSICAL DATABASE
//         ROOT=$ROOT,                ROOT OF THE SYSTEM
//         INDEX='$INDEX',            INDEX OF VSAM SYSTEM FILES
//         INDEXP='$INDEXP',          INDEX OF SYSEM FILES
//*:       VSAMCAT='$CATU',           USER VSAM CATALOG
//*:       SYSTCAT='$CATV',           SYSTEM VSAM CATALOG
//         STEPLIB='$MODB',           LIBRARY OF LOAD-MODULES
//         LOADLIB='$LDLIB',          ENDEVOR LOADLIB
//         CONLIB='$CONLIB',          ENDEVOR CONLIB
//         OUT=$OUT,                  OUTPUT CLASS
//         UWK=$UWK,                  WORK UNIT
//         RADP=$RADP,                PREFIX OF PROCEDURE NAMES
//         SPAMB='(TRK,(02,01),RLSE)', SPACE TRANSACTIONS FOR 'BSTIPT01
//         SPAMS='(TRK,(05,05),RLSE)', SPACE 'C1MSGS1 & C1PRINT' NDVRC1
//         SPAUR='(TRK,(02,01),RLSE)', SPACE TEMPORARY FILE
//         SPAJC='(TRK,(05,01),RLSE)'  SPACE GENERATED JCL
//*-------------------------------------------------------------------*
//*
//INPUT   EXEC PGM=PTU001
//STEPLIB  DD DSN=&STEPLIB,DISP=SHR
//CARTE    DD DDNAME=SYSIN,DCB=BLKSIZE=80
//PAC7MB   DD DSN=&&RRNDMB,DISP=(,PASS),
//            UNIT=&UWK,SPACE=(TRK,(1,1),RLSE),
//            DCB=(RECFM=FB,LRECL=80,BLKSIZE=80)
//*
//PNCI10  EXEC PGM=PNCI10,
//             PARM='*(''CONSTANTES-PAC'' OR ''PACBASE-CONSTANT'')'
//STEPLIB  DD DSN=$MODB,DISP=SHR
//PAC7TY   DD DSN=&INDEX..&ROOT.&FILE.TY,DISP=SHR
//PAC7IN   DD DSN=&&RRNDMB,DISP=(OLD,PASS)
//PAC7BS   DD DSN=&&PAC7BS,DISP=(,PASS),UNIT=&UWK,
//            DCB=(RECFM=FB,LRECL=80,BLKSIZE=6080),
//            SPACE=&SPAMB
//SYSOUT   DD SYSOUT=&OUT
//SYSOUX   DD SYSOUT=&OUT
//SYSUDUMP DD SYSOUT=&OUT
//*
//NDVRC1A EXEC PGM=NDVRC1,DYNAMNBR=1500,REGION=4096K,PARM='C1BM3000'
//STEPLIB  DD DSN=&LOADLIB,DISP=SHR
//CONLIB   DD DSN=&CONLIB,DISP=SHR
//SYSPRINT DD SYSOUT=&OUT
//SYSUDUMP DD SYSOUT=&OUT
//SORTWK01 DD UNIT=SYSDA,SPACE=(CYL,(2,1))
//SORTWK02 DD UNIT=SYSDA,SPACE=(CYL,(2,1))
//SORTWK03 DD UNIT=SYSDA,SPACE=(CYL,(2,1))
//SORTWK04 DD UNIT=SYSDA,SPACE=(CYL,(2,1))
//C1TPDD01 DD UNIT=SYSDA,SPACE=(CYL,5),
//            DCB=(RECFM=VB,LRECL=260,BLKSIZE=6160)
//C1TPDD02 DD UNIT=SYSDA,SPACE=(CYL,5),
//            DCB=(RECFM=VB,LRECL=260,BLKSIZE=6160)
//C1TPLSIN DD UNIT=SYSDA,SPACE=(CYL,5),
//            DCB=(RECFM=FB,LRECL=80,BLKSIZE=6160)
//C1TPLSOU DD UNIT=SYSDA,SPACE=(CYL,5)
//C1PLMSGS DD SYSOUT=&OUT
//*1MSGS1  DD SYSOUT=&OUT
//C1MSGS1  DD DSN=&&C1MSGS1A,DISP=(,PASS),UNIT=&UWK,
//            DCB=(RECFM=FBA,LRECL=133,BLKSIZE=26600),
//            SPACE=&SPAMS
//C1PRINT  DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,BLKSIZE=6171)
//SYSABEND DD SYSOUT=&OUT
//SYSOUT   DD SYSOUT=&OUT
```

```
//BSTIPT01 DD DSN=&&PAC7BS,DISP=(OLD,PASS)
//*
//PNDR20  EXEC PGM=PNDR20
//STEPLIB  DD DSN=&STEPLIB,DISP=SHR
//*:STEPCAT  DD DSN=&VSAMCAT,DISP=SHR
//PAC7LT   DD DSN=*.NDVRC1A.C1MSGS1,DISP=(OLD,PASS)
//PAC7BS   DD DSN=&&PAC7SB,DISP=(,PASS),UNIT=&UWK,
//            DCB=(RECFM=FB,LRECL=80,BLKSIZE=6080),
//            SPACE=&SPAMB
//PAC7BT   DD DSN=&&PAC7TB,DISP=(,PASS),UNIT=&UWK,
//            DCB=(RECFM=FB,LRECL=80,BLKSIZE=6080),
//            SPACE=&SPAMB
//PAC7XT   DD DSN=&&PAC7XT,DISP=(,PASS),UNIT=&UWK,
//            DCB=(RECFM=FB,LRECL=43,BLKSIZE=4300),
//            SPACE=&SPAMB
//SYSOUT   DD SYSOUT=&OUT
//SYSOUX   DD SYSOUT=&OUT
//SYSUDUMP DD SYSOUT=&OUT
//*
//NDVRC1B EXEC PGM=NDVRC1,DYNAMNBR=1500,REGION=4096K,PARM='C1BM3000'
//STEPLIB  DD DSN=&LOADLIB,DISP=SHR
//CONLIB   DD DSN=&CONLIB,DISP=SHR
//SYSPRINT DD SYSOUT=&OUT
//SYSUDUMP DD SYSOUT=&OUT
//SORTWK01 DD UNIT=SYSDA,SPACE=(CYL,(2,1))
//SORTWK02 DD UNIT=SYSDA,SPACE=(CYL,(2,1))
//SORTWK03 DD UNIT=SYSDA,SPACE=(CYL,(2,1))
//SORTWK04 DD UNIT=SYSDA,SPACE=(CYL,(2,1))
//C1TPDD01 DD UNIT=SYSDA,SPACE=(CYL,5),
//            DCB=(RECFM=VB,LRECL=260,BLKSIZE=6160)
//C1TPDD02 DD UNIT=SYSDA,SPACE=(CYL,5),
//            DCB=(RECFM=VB,LRECL=260,BLKSIZE=6160)
//C1TPLSIN DD UNIT=SYSDA,SPACE=(CYL,5),
//            DCB=(RECFM=FB,LRECL=80,BLKSIZE=6160)
//C1TPLSOU DD UNIT=SYSDA,SPACE=(CYL,5)
//C1PLMSGS DD SYSOUT=&OUT
//*1MSGS1  DD SYSOUT=&OUT
//C1MSGS1  DD DSN=&&C1MSGS1B,DISP=(,PASS),UNIT=&UWK,
//            DCB=(RECFM=FBA,LRECL=133,BLKSIZE=26600),
//            SPACE=&SPAMS
//*1PRINT  DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,BLKSIZE=6171)
//C1PRINT  DD DSN=&&C1PRINTB,DISP=(,PASS),UNIT=&UWK,
//            DCB=(RECFM=FBA,LRECL=133,BLKSIZE=26600),
//            SPACE=&SPAMS
//SYSABEND DD SYSOUT=&OUT
//SYSOUT   DD SYSOUT=&OUT
//BSTIPT01 DD DSN=&&PAC7SB,DISP=(OLD,PASS)
//*
//NDVRC1C EXEC PGM=NDVRC1,DYNAMNBR=1500,REGION=4096K,PARM='C1BM3000'
//STEPLIB  DD DSN=&LOADLIB,DISP=SHR
//CONLIB   DD DSN=&CONLIB,DISP=SHR
//SYSPRINT DD SYSOUT=&OUT
//SYSUDUMP DD SYSOUT=&OUT
//SORTWK01 DD UNIT=SYSDA,SPACE=(CYL,(2,1))
//SORTWK02 DD UNIT=SYSDA,SPACE=(CYL,(2,1))
//SORTWK03 DD UNIT=SYSDA,SPACE=(CYL,(2,1))
//SORTWK04 DD UNIT=SYSDA,SPACE=(CYL,(2,1))
//C1TPDD01 DD UNIT=SYSDA,SPACE=(CYL,5),
//            DCB=(RECFM=VB,LRECL=260,BLKSIZE=6160)
//C1TPDD02 DD UNIT=SYSDA,SPACE=(CYL,5),
//            DCB=(RECFM=VB,LRECL=260,BLKSIZE=6160)
//C1TPLSIN DD UNIT=SYSDA,SPACE=(CYL,5),
//            DCB=(RECFM=FB,LRECL=80,BLKSIZE=6160)
//C1TPLSOU DD UNIT=SYSDA,SPACE=(CYL,5)
```

```
//C1PLMSGS DD SYSOUT=&OUT
//C1MSGS1  DD SYSOUT=&OUT
//*1PRINT  DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,BLKSIZE=6171)
//C1PRINT  DD DSN=&&C1PRINTC,DISP=(,PASS),UNIT=&UWK,
//            DCB=(RECFM=FBA,LRECL=133,BLKSIZE=26600),
//            SPACE=&SPAMS
//SYSABEND DD SYSOUT=&OUT
//SYSOUT   DD SYSOUT=&OUT
//BSTIPT01 DD DSN=&&PAC7TB,DISP=(OLD,PASS)
//*
//PNDR30  EXEC PGM=PNDR30,PARM='&RADP'
//STEPLIB  DD DSN=&STEPLIB,DISP=SHR
//*:STEPCAT  DD DSN=&VSAMCAT,DISP=SHR
//PAC7TY   DD DSN=&INDEX..&ROOT.&FILE.TY,DISP=SHR
//PAC7LT   DD DSN=*.NDVRC1B.C1MSGS1,DISP=(OLD,PASS)
//PAC7PT   DD DSN=*.NDVRC1B.C1PRINT,DISP=(OLD,PASS)
//PAC7XT   DD DSN=&&PAC7XT,DISP=(OLD,PASS)
//PAC7JC   DD DSN=&&PAC7JC,DISP=(,PASS),UNIT=&UWK,
//            DCB=(RECFM=FB,LRECL=80,BLKSIZE=24000),
//            SPACE=&SPAJC
//PAC7UQ   DD DSN=&&PAC7UQ,DISP=(,PASS),UNIT=&UWK,
//            DCB=(RECFM=FB,LRECL=170,BLKSIZE=17000),
//            SPACE=&SPAUR
//PAC7JD   DD UNIT=&UWK,SPACE=&SPAUR,DCB=BLKSIZE=24000
//SYSOUT   DD SYSOUT=&OUT
//SYSOUX   DD SYSOUT=&OUT
//SYSUDUMP DD SYSOUT=&OUT
//*
//*
//DEFINUQ EXEC PGM=IDCAMS,COND=(00,NE,PNDR30)
//*:STEPCAT  DD DSN=&VSAMCAT,DISP=SHR
//SYSPRINT DD SYSOUT=&OUT
//SYSIN    DD DSN=&INDEXP..&ROOT.&ROOT.SY(DF&ROOT.&FILE.UQ),DISP=SHR
//*
//PNINUQ  EXEC PGM=PNINUQ,COND=(00,NE,PNDR30)
//STEPLIB  DD DSN=&STEPLIB,DISP=SHR
//PAC7UQ   DD DSN=&INDEX..&ROOT.&FILE.UQ,DISP=SHR
//SYSOUT   DD SYSOUT=*
//SYSUDUMP DD SYSOUT=*
//*
//PNDR35  EXEC PGM=PNDR35
//STEPLIB  DD DSN=&STEPLIB,DISP=SHR
//*:STEPCAT  DD DSN=&VSAMCAT,DISP=SHR
//PAC7UQ   DD DSN=&INDEX..&ROOT.&FILE.UQ,DISP=SHR
//PAC7PT   DD DSN=*.NDVRC1C.C1PRINT,DISP=(OLD,PASS)
//PAC7QU   DD DSN=&&PAC7UQ,DISP=(OLD,PASS)
//SYSOUT   DD SYSOUT=&OUT
//SYSOUX   DD SYSOUT=&OUT
//SYSUDUMP DD SYSOUT=&OUT
//*
```

7.14. RTND: RETRIEVAL 8.0.2 --> 2.5

RTND: RETRIEVAL 8.0.2 --> 2.5

The 'RRND' procedure retrieves the 'TS' TARGET LIBRARIES/ SESSIONS and 'TY' TYPEs tables of the VA Pac-ENDEVOR interface to switch from release 8.0.2 to release 2.5. It assigns the VA Pac database internal code to the 'TS' TARGET LIBRARIES/SESSIONS.

EXECUTION CONDITION

The Interface must not be in use.

USER INPUT

.One identification line of the VA Pac database internal code.

Pos.	Len.	Value	Meaning
1	4		VA Pac database internal code

PRINTED REPORT

The list of TYPES, TARGET LIBRARIES AND SESSIONS defined on the site.

RESULT

This procedure rebuilds the data of the TARGET LIBRARIES/SESSIONS and TYPEs tables for release 2.5.

DESCRIPTION OF STEPS

This procedure comprises the following steps:

. Take the input into account : PTU001

. 2.5 TS and TY tables definition : IDCAMS

. TS and TY tables retrival : PNRPTS

. Print the TYPEs and TARGET LIBRARIES and SESSIONS table after retrieval : PNDU30

```
TAKE THE INPUT INTO ACCOUNT: PTU001

. Input data                                      : CARTE
  DSN=SYSIN

. Output file                                     : PAC7MB
  DSN=&&RTNDMB

TS AND TY TABLES DEFINITION: DEFINE

. This step carries out a DELETE/DEFINE on the TS and TY
files.
 - Defined files:
   DSN=&INDEX..&ROOT.&FILE.TS
   DSN=&INDEX..&ROOT.&FILE.TY
```

TS AND TY TABLES RETRIEVAL: PNRPTS

```
. Input files:
 - 8.0.2 TARGET LIBRARIES/SESSIONS table        : PAC7TT
   DSN=&&OLDTS
 - 8.0.2 TYPES table                            : PAC7TZ
   DSN=&&OLDTY
 - Identification VA Pac logical database code  : PAC7MV
   DSN=&&RTNDMB
 - VA Pac error message file                    : PAC7AE
   DSN=&INDEXX..&ROOTX.&ROOTX.AE

. Output files:
 - 2.5 TARGET LIBRARIES/SESSIONS table          : PAC7TS
   DSN=&INDEX..&ROOT.&FILE.TS
 - 2.5 TYPES table                              : PAC7TY
   DSN=&INDEX..&ROOT.&FILE.TY
```

PRINTOUT OF TYPES, TARGET LIBRARIES AND SESSIONS TABLES
AFTER RETRIEVAL: PNDU30

```
. Input files:
 - TYPEs file                                   : PAC7TY
   DSN=&INDEX..&ROOT.&FILE.TY
 - LIBRARIES and SESSIONS file                  : PAC7TS
   DSN=&INDEX..&ROOT.&FILE.TS
 - VA Pac error message file                    : PAC7AE
   DSN=&INDEXX..&ROOTX.&ROOTX.AE

. Output file:
 - Printout of file contents                    : PAC7ET
   DSN=&OUT
```

```
//***************************************************************
//* VA PAC-ENDEVOR 2.5   : RETRIEVE 8.0.2 --> 2.5
//***************************************************************
//$RADP.RTND PROC FILE=$FILE,         NUMBER OF PHYSICAL DATABASE
//         ROOT=$ROOT,                ROOT OF THE SYSTEM
//         ROOTX=$ROOTX,              ROOT OF 'VA PAC' SYSTEM
//         INDEX='$INDEX',            INDEX OF VSAM SYSTEM FILES
//         INDEXX='$INDEXX',          INDEX OF VA PAC VSAM SYSTEMFILES
//         INDEXP='$INDEXP',          INDEX OF NON VSAM SYSTEM FILES
//*:       VSAMCAT='$CATU',           USER VSAM CATALOG
//*:       SYSTCAT='$CATV',           SYSTEM VSAM CATALOG
//         STEPLIB='$MODB',           LIBRARY OF LOAD-MODULES
//         OUT=$OUT,                  OUTPUT CLASS
//         UWK=$UWK,                  WORK UNIT
//         OLDTS=,                    OLD LIBRARY/SESSION FILE
//         OLDTY=,                    OLD ENDEVOR TYPE TABLE
//    PSBLIB='$PSBLIB',     PSB LIBRARY
//    DBDLIB='$DBDLIB',     DBD LIBRARY
//    RESLIB='$RESLIB',     RESLIB IMS
//    PROCLIB='$PRCLIB',    PROCLIB IMS
// BUF=40,SPIE=0,TEST=0,EXCPVR=0,RST=0,PRLD=,SRCH=0,
// CKPTID=,MON=N,LOGA=0,FMTO=T,DBRC=$DBRC,IRLM=$IRLM
//*--------------------------------------------------------------
//*
//INPUT   EXEC PGM=PTU001
//STEPLIB  DD DSN=&STEPLIB,DISP=SHR
//CARTE    DD DDNAME=SYSIN,DCB=BLKSIZE=80
//PAC7MB   DD DSN=&&RTNDMB,DISP=(,PASS),
//            UNIT=&UWK,SPACE=(TRK,(1,1),RLSE),
//            DCB=(RECFM=FB,LRECL=80,BLKSIZE=80)
//*
//DEFINE  EXEC PGM=IDCAMS
//*:STEPCAT DD DSN=&VSAMCAT,DISP=SHR
//SYSPRINT DD SYSOUT=&OUT
//SYSIN    DD DSN=&INDEXP..&ROOT.&ROOT.SY(DF&ROOT.&FILE.TS),DISP=SHR
//SYSIN    DD DSN=&INDEXP..&ROOT.&ROOT.SY(DF&ROOT.&FILE.TY),DISP=SHR
//*
//PNRPTS  EXEC PGM=DFSRRC00,REGION=$REGSIZ,
//        PARM=(DLI,PNRPTS,PNRPTS$SUG,&BUF,
//        &SPIE&TEST&EXCPVR&RST,&PRLD,
//        &SRCH,&CKPTID,&MON,&LOGA,&FMTO,,,&DBRC,&IRLM)
//STEPLIB  DD DSN=&RESLIB,DISP=SHR
//         DD DSN=&STEPLIB,DISP=SHR
//DFSRESLB DD DSN=&RESLIB,DISP=SHR
//IMS      DD DSN=&PSBLIB,DISP=SHR
//         DD DSN=&DBDLIB,DISP=SHR
//DDSNAP   DD SYSOUT=&OUT
//PROCLIB  DD DSN=&PROCLIB,DISP=SHR
//IEFRDER  DD DUMMY,
//            DCB=(RECFM=VB,BLKSIZE=1920,LRECL=1916,BUFNO=2)
//SYSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//            BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//            BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSMON   DD DUMMY
//DFSVSAMP DD DSN=&INDEXP..&ROOT.&ROOT.SY(DFSVSAM8),DISP=SHR
//*:STEPCAT  DD DSN=&VSAMCAT,DISP=SHR
//*:         DD DSN=&SYSTCAT,DISP=SHR
//PAC7AE$SUF DD DSN=&INDEXX..&ROOTX.&ROOTX.AE,DISP=SHR
//PAC7TS   DD DSN=&INDEX..&ROOT.&FILE.TS,DISP=SHR
//PAC7TY   DD DSN=&INDEX..&ROOT.&FILE.TY,DISP=SHR
//PAC7TT   DD DSN=&OLDTS,DISP=SHR
//PAC7TZ   DD DSN=&OLDTY,DISP=SHR
//PAC7MV   DD DSN=&&RTNDMB,DISP=(OLD,DELETE)
```

```
//PAC7ET   DD SYSOUT=&OUT
//SYSOUT   DD SYSOUT=&OUT
//SYSOUX   DD SYSOUT=&OUT
//SYSUDUMP DD SYSOUT=&OUT
//*
//PNDU30  EXEC PGM=DFSRRC00,REGION=$REGSIZ,
//        PARM=(DLI,PNDU30,PNDU30$SUG,&BUF,
//        &SPIE&TEST&EXCPVR&RST,&PRLD,
//        &SRCH,&CKPTID,&MON,&LOGA,&FMTO,,,&DBRC,&IRLM)
//STEPLIB  DD DSN=&RESLIB,DISP=SHR
//         DD DSN=&STEPLIB,DISP=SHR
//DFSRESLB DD DSN=&RESLIB,DISP=SHR
//IMS      DD DSN=&PSBLIB,DISP=SHR
//         DD DSN=&DBDLIB,DISP=SHR
//DDSNAP   DD SYSOUT=&OUT
//PROCLIB  DD DSN=&PROCLIB,DISP=SHR
//IEFRDER  DD DUMMY,
//            DCB=(RECFM=VB,BLKSIZE=1920,LRECL=1916,BUFNO=2)
//SYSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//            BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//            BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSMON   DD DUMMY
//DFSVSAMP DD DSN=&INDEXP..&ROOT.&ROOT.SY(DFSVSAM8),DISP=SHR
//*:STEPCAT  DD DSN=&VSAMCAT,DISP=SHR
//*:         DD DSN=&SYSTCAT,DISP=SHR
//PAC7AE$SUF DD DSN=&INDEXX..&ROOTX.&ROOTX.AE,DISP=SHR
//PAC7TS   DD DSN=&INDEX..&ROOT.&FILE.TS,DISP=SHR
//PAC7TY   DD DSN=&INDEX..&ROOT.&FILE.TY,DISP=SHR
//PAC7ET   DD SYSOUT=&OUT
//SYSOUT   DD SYSOUT=&OUT
//SYSOUX   DD SYSOUT=&OUT
//SYSPRINT DD SYSOUT=&OUT
//SYSUDUMP DD SYSOUT=&OUT
//*
```

7.15. TYND: TYPE UPDATING

TYND: TYPE UPDATING

The TYND procedure updates the TYPEs table of the elements managed by the Interface.

EXECUTION CONDITION

None.

USER INPUT

. One line per update request.

TYPEs UPDATE PARAMETER LINE

Pos.	Length	Value	Meaning
1	1		Transaction code
		'C'	Creation
		'M'	Modification
		'D'	Deletion
		'X'	Creation or Modification
2	8	pppppppp	VA Pac ELEMENT type
11	8	iiiiiiii	INFOPAC ELEMENT type
19	1		VA Pac ELEMENT nature
		'P'	Program
		'O'	Screen
		'B'	Database block
		'D'	Copy Data clause
20	36		Comments

PRINTED REPORT

An update report with the detected anomalies and a report with the TYPES, TARGET SESSIONS and LIBRARIES defined on the site.

```
 DESCRIPTION OF STEPS

This procedure comprises three steps:

 . Recognition of input                       : PTU001

 . Update of the TYPEs table                  : PNDU20

 . Printing of the TYPEs, TARGET LIBRARIES and
   SESSIONS table                             : PNDU30


 RECOGNITION OF INPUT: PTU001

. Input data                                  : CARTE
  DDNAME=SYSIN

. Output file                                 : PAC7MB
  DSN=&&TYNDMB


 UPDATE OF THE TYPEs TABLE: PNDU20

. Input/Output file:
 - TYPEs file:                                : PAC7TY
   DSN=&INDEX..&ROOT.&FILE.TY

. Input files:
 - VA Pac error messages file                : PAC7AE
   DSN=&INDEXX..&ROOTX.&ROOTX.AE
 - Update transactions                       : PAC7MV
   DSN=&INPUT

. Output report:
 - Update report                             : PAC7ET
   DSN=&OUT


PRINTING OF TYPEs, TARGET LIBRARIES/SESSIONS: PNDU30


. Input files:
 - TYPEs file:                               : PAC7TY
   DSN=&INDEX..&ROOT.&FILE.TY
 - LIBRARIES and SESSIONS file:              : PAC7TS
   DSN=&INDEX..&ROOT.&FILE.TS
 - VA Pac error messages file:               : PAC7AE
   DSN=&INDEXX..&ROOTX.&ROOTX.AE

. Output file:
 - Printing of files contents                : PAC7ET
   DSN=&OUT
```

```
//*****************************************************************
//* VA PAC-ENDEVOR 2.5   : UPDATE OF ENDEVOR TYPES TABLE              *
//*****************************************************************
//$RADP.TYND PROC FILE=$FILE,         NUMBER OF PHYSICAL DATABASE
//         ROOT=$ROOT,                ROOT OF THE SYSTEM
//         ROOTX=$ROOTX,              ROOT OF 'VA PAC' SYSTEM
//         INDEX='$INDEX',            INDEX OF VSAM SYSTEM FILES
//         INDEXP='$INDEXP',          INDEX OF NON VSAM SYSTEM FILES
//         INDEXX='$INDEXX',          INDEX OF VA PAC VSAM SYSTEM FILES
//*:       VSAMCAT='$CATU',           USER VSAM CATALOG
//*:       SYSTCAT='$CATV',           SYSTEM VSAM CATALOG
//         STEPLIB='$MODB',           LIBRARY OF LOAD-MODULES
//         OUT=$OUT,                  OUTPUT CLASS
//         UWK=$UWK,                  WORK UNIT
//    PSBLIB='$PSBLIB',     PSB LIBRARY
//    DBDLIB='$DBDLIB',     DBD LIBRARY
//    RESLIB='$RESLIB',     RESLIB IMS
//    PROCLIB='$PRCLIB',    PROCLIB IMS
// BUF=40,SPIE=0,TEST=0,EXCPVR=0,RST=0,PRLD=,SRCH=0,
// CKPTID=,MON=N,LOGA=0,FMTO=T,DBRC=$DBRC,IRLM=$IRLM
//*-------------------------------------------------------------
//*
//INPUT   EXEC PGM=PTU001
//STEPLIB  DD DSN=&STEPLIB,DISP=SHR
//CARTE    DD DDNAME=SYSIN,DCB=BLKSIZE=80
//PAC7MB   DD DSN=&&TYNDMB,DISP=(,PASS),
//            UNIT=&UWK,SPACE=(TRK,(1,1),RLSE),
//            DCB=(RECFM=FB,LRECL=80,BLKSIZE=80)
//*
//PNDU20  EXEC PGM=DFSRRC00,REGION=$REGSIZ,
//        PARM=(DLI,PNDU20,PNDU20$SUG,&BUF,
//        &SPIE&TEST&EXCPVR&RST,&PRLD,
//        &SRCH,&CKPTID,&MON,&LOGA,&FMTO,,,&DBRC,&IRLM)
//STEPLIB  DD DSN=&RESLIB,DISP=SHR
//         DD DSN=&STEPLIB,DISP=SHR
//DFSRESLB DD DSN=&RESLIB,DISP=SHR
//IMS      DD DSN=&PSBLIB,DISP=SHR
//         DD DSN=&DBDLIB,DISP=SHR
//DDSNAP   DD SYSOUT=&OUT
//PROCLIB  DD DSN=&PROCLIB,DISP=SHR
//IEFRDER  DD DUMMY,
//            DCB=(RECFM=VB,BLKSIZE=1920,LRECL=1916,BUFNO=2)
//SYSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//            BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//            BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSMON   DD DUMMY
//DFSVSAMP DD DSN=&INDEXP..&ROOT.&ROOT.SY(DFSVSAM8),DISP=SHR
//*:STEPCAT  DD DSN=&VSAMCAT,DISP=SHR
//*:         DD DSN=&SYSTCAT,DISP=SHR
//PAC7AE$SUF DD DSN=&INDEXX..&ROOTX.&ROOTX.AE,DISP=SHR
//PAC7TY   DD DSN=&INDEX..&ROOT.&FILE.TY,DISP=SHR
//PAC7MV   DD DSN=&&TYNDMB,DISP=(OLD,DELETE)
//PAC7ET   DD SYSOUT=&OUT
//SYSOUT   DD SYSOUT=&OUT
//SYSOUX   DD SYSOUT=&OUT
//SYSUDUMP DD SYSOUT=&OUT
//*
//PNDU30  EXEC PGM=DFSRRC00,REGION=$REGSIZ,
//        PARM=(DLI,PNDU30,PNDU30$SUG,&BUF,
//        &SPIE&TEST&EXCPVR&RST,&PRLD,
//        &SRCH,&CKPTID,&MON,&LOGA,&FMTO,,,&DBRC,&IRLM)
//STEPLIB  DD DSN=&RESLIB,DISP=SHR
//         DD DSN=&STEPLIB,DISP=SHR
```

```
//DFSRESLB DD DSN=&RESLIB,DISP=SHR
//IMS      DD DSN=&PSBLIB,DISP=SHR
//         DD DSN=&DBDLIB,DISP=SHR
//DDSNAP   DD SYSOUT=&OUT
//PROCLIB  DD DSN=&PROCLIB,DISP=SHR
//IEFRDER  DD DUMMY,
//            DCB=(RECFM=VB,BLKSIZE=1920,LRECL=1916,BUFNO=2)
//SYSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//            BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//            BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSMON   DD DUMMY
//DFSVSAMP DD DSN=&INDEXP..&ROOT.&ROOT.SY(DFSVSAM8),DISP=SHR
//*:STEPCAT  DD DSN=&VSAMCAT,DISP=SHR
//*:         DD DSN=&SYSTCAT,DISP=SHR
//PAC7AE$SUF DD DSN=&INDEXX..&ROOTX.&ROOTX.AE,DISP=SHR
//PAC7TS   DD DSN=&INDEX..&ROOT.&FILE.TS,DISP=SHR
//PAC7TY   DD DSN=&INDEX..&ROOT.&FILE.TY,DISP=SHR
//PAC7ET   DD SYSOUT=&OUT
//SYSOUT   DD SYSOUT=&OUT
//SYSOUX   DD SYSOUT=&OUT
//SYSPRINT DD SYSOUT=&OUT
//SYSUDUMP DD SYSOUT=&OUT
//*
```

7.16. UPND: VA PAC ELEMENTS IMPORT INTO ENDEVOR

UPND: IMPORT OF VA PAC ELEMENTS TO ENDEVOR

The UPND procedure imports generated VA Pac and INFOPAC ELEMENTs into ENDEVOR.

EXECUTION CONDITION

This procedure is prepared by VA Pac GPRT generation-print procedure and completed by the GPND procedure.

DESCRIPTION OF STEPS

This procedure comprises four steps:

```
 . Recognition of the content of the INFOPAC element
   in a temporary file                          : IEBGENER

 . Import of a VA Pac ELEMENT into Endevor       : NDVRC1

 . Import of an INFOPAC ELEMENT into Endevor     : NDVRC1

 . Deletion of VA Pac ELEMENT if error during the import
   of INFOPAC ELEMENT                           : NDVRC1
```

RECOGNITION OF INFOPAC CONTENTS: IEBGENER

```
. Input file:
- Contents of the INFOPAC ELEMENT

. Output file:
  DSN=&&INFOPAC
```

IMPORT OF VA PAC ELEMENT IN ENDEVOR: NDVRC1

```
. Input files:
 - TYPES files:                                   : PAC7TY
   DSN=&INDEX..&ROOT.&FILE.TY
 - Endevor ADD request of the VA Pac ELEMENT      : BSTIPT01

. Input/Output files:
 - Interface work file                            : PAC7PU
   DSN=&INDEX..&ROOT.&FILE.PU
 - Interface work file                            : PAC7UP
   DSN=&INDEX..&ROOT.&FILE.UP

. Output file:
 - Interface journal file                         : PAC7UQ
   DSN=&INDEX..&ROOT.&FILE.UQ
```

```
IMPORT OF THE INFOPAC ELEMENT INTO ENDEVOR: NDVRC1

. Input files:
 - TYPES file                                       : PAC7TY
   DSN=&INDEX..&ROOT.&FILE.TY
 - Endevor ADD request of the INFOPAC ELEMENT       : BSTIPT01

. Input/Output files:
 - Interface work file                              : PAC7PU
   DSN=&INDEX..&ROOT.&FILE.PU
 - Interface work file                              : PAC7UP
   DSN=&INDEX..&ROOT.&FILE.UP

. Output file:
 - Interface journal file                           : PAC7UQ
   DSN=&INDEX..&ROOT.&FILE.UQ


DELETION OF VA PAC ELEMENT IN CASE OF ERROR: NDVRC1

. Input files:
 - Interface TYPES file                             : PAC7TY
   DSN=&INDEX..&ROOT.&FILE.TY
 - Endevor DELETE request on VA Pac ELEMENT         : BSTIPT01

. Input/Output files:
 - Interface work file                              : PAC7PU
   DSN=&INDEX..&ROOT.&FILE.PU
 - Interface work file                              : PAC7UP
   DSN=&INDEX..&ROOT.&FILE.UP

. Output file:
 - Interface journal file                           : PAC7UQ
   DSN=&INDEX..&ROOT.&FILE.UQ
```

```
//*****************************************************************
//* VA PAC-ENDEVOR 2.5   : ENDEVOR UPDATE                          *
//*****************************************************************
//$RADP.UPND PROC FILE=$FILE,         NUMBER OF PHYSICAL DATABASE
//         ROOT=$ROOT,                ROOT OF THE SYSTEM
//         ROOTX=$ROOTX,              ROOT OF 'VA PAC' SYSTEM
//         INDEX='$INDEX',            INDEX OF VSAM SYSTEM FILES
//         INDEXP='$INDEXP',          INDEX OF NON VSAM SYSTEM F ILES
//         INDEXX='$INDEXX',          INDEX OF VA PAC VSAM SYSTEM FILES
//*:       VSAMCAT='$CATU',           USER VSAM CATALOG
//*:       SYSTCAT='$CATV',           SYSTEM VSAM CATALOG
//         STEPLIB='$MODB',           LIBRARY OF LOAD-MODULES
//         LOADLIB='$LDLIB',          ENDEVOR LOADLIB
//         CONLIB='$CONLIB',          ENDEVOR CONLIB
//         OUT=$OUT,                  OUTPUT CLASS
//    PSBLIB='$PSBLIB',     PSB LIBRARY
//    DBDLIB='$DBDLIB',     DBD LIBRARY
//    RESLIB='$RESLIB',     RESLIB IMS
//    PROCLIB='$PRCLIB',    PROCLIB IMS
// BUF=40,SPIE=0,TEST=0,EXCPVR=0,RST=0,PRLD=,SRCH=0,
// CKPTID=,MON=N,LOGA=0,FMTO=T,DBRC=$DBRC,IRLM=$IRLM
//***********************************************************
//**********************************************
//* CREATE 'INFOPAC' ELEMENT IN TEMPORARY FILE *
//*
//**********************************************
//*
//STEP01  EXEC PGM=IEBGENER
//SYSUT1   DD DUMMY
//SYSUT2   DD DSN=&&INFOPAC,UNIT=SYSDA,DISP=(NEW,PASS),
//            DCB=(RECFM=FB,LRECL=80,BLKSIZE=80),
//            SPACE=(TRK,(1,1),RLSE)
//SCLDEL   DD DSN=&&SCLDEL,UNIT=SYSDA,DISP=(NEW,PASS),
//            DCB=(RECFM=FB,LRECL=80,BLKSIZE=80),
//            SPACE=(TRK,(1,1),RLSE)
//SYSIN    DD DUMMY
//SYSPRINT DD SYSOUT=&OUT
//*
//**********************************************
//* ENDEVOR 'ADD' ACTION FOR 'VA PAC' ELEMENT  *
//**********************************************
//*
//STEP02  EXEC PGM=NDVRC1,DYNAMNBR=1500,REGION=4096K,PARM='C1BM3000'
//STEPLIB  DD DSN=&LOADLIB,DISP=SHR
//CONLIB   DD DSN=&CONLIB,DISP=SHR
//SYSPRINT DD SYSOUT=&OUT
//SYSUDUMP DD SYSOUT=&OUT
//SORTWK01 DD UNIT=SYSDA,SPACE=(CYL,(2,1))
//SORTWK02 DD UNIT=SYSDA,SPACE=(CYL,(2,1))
//SORTWK03 DD UNIT=SYSDA,SPACE=(CYL,(2,1))
//SORTWK04 DD UNIT=SYSDA,SPACE=(CYL,(2,1))
//C1TPDD01 DD UNIT=SYSDA,SPACE=(CYL,5),
//            DCB=(RECFM=VB,LRECL=260,BLKSIZE=6160)
//C1TPDD02 DD UNIT=SYSDA,SPACE=(CYL,5),
//            DCB=(RECFM=VB,LRECL=260,BLKSIZE=6160)
//C1TPLSIN DD UNIT=SYSDA,SPACE=(CYL,5),
//            DCB=(RECFM=FB,LRECL=80,BLKSIZE=6160)
//C1TPLSOU DD UNIT=SYSDA,SPACE=(CYL,5)
//PAC7TY   DD DSN=&INDEX..&ROOT.&FILE.TY,DISP=SHR
//PAC7PU   DD DSN=&INDEX..&ROOT.&FILE.UP,DISP=SHR
//PAC7UP   DD DSN=&INDEX..&ROOT.&FILE.UP,DISP=SHR
//PAC7UQ   DD DSN=&INDEX..&ROOT.&FILE.UQ,DISP=SHR
//C1PLMSGS DD SYSOUT=&OUT
//C1MSGS1  DD SYSOUT=&OUT
```

```
//C1PRINT  DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,BLKSIZE=6171)
//SYSABEND DD SYSOUT=&OUT
//SYSOUT   DD SYSOUT=&OUT
//PTRACE   DD SYSOUT=&OUT
//BSTIPT01 DD DUMMY
//*
//*********************************************
//*     BUILD 'DELETE' SCL VA PAC ELEMENT      *
//*              IF 'ADD' ERROR                *
//*********************************************
//*
//STEP03  EXEC PGM=DFSRRC00,REGION=$REGSIZ,
//        COND=(11,GE,STEP02),
//        PARM=(DLI,PNDV20,PNDV20$SUG,&BUF,
//        &SPIE&TEST&EXCPVR&RST,&PRLD,
//        &SRCH,&CKPTID,&MON,&LOGA,&FMTO,,,&DBRC,&IRLM)
//STEPLIB  DD DSN=&RESLIB,DISP=SHR
//         DD DSN=&STEPLIB,DISP=SHR
//DFSRESLB DD DSN=&RESLIB,DISP=SHR
//IMS      DD DSN=&PSBLIB,DISP=SHR
//         DD DSN=&DBDLIB,DISP=SHR
//DDSNAP   DD SYSOUT=&OUT
//PROCLIB  DD DSN=&PROCLIB,DISP=SHR
//IEFRDER  DD DUMMY,
//            DCB=(RECFM=VB,BLKSIZE=1920,LRECL=1916,BUFNO=2)
//SYSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//            BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//            BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSMON   DD DUMMY
//DFSVSAMP DD DSN=&INDEXP..&ROOT.&ROOT.SY(DFSVSAM8),DISP=SHR
//*:STEPCAT  DD DSN=&SYSTCAT,DISP=SHR
//*:         DD DSN=&VSAMCAT,DISP=SHR
//SYSIN    DD DUMMY
//SYSUT1   DD DUMMY
//SYSUT2   DD DSN=*.STEP01.SCLDEL,DISP=(MOD,PASS)
//PAC7AE$SUF DD DSN=&INDEXX..&ROOTX.&ROOTX.AE,DISP=SHR
//SYSOUT   DD SYSOUT=&OUT
//SYSPRINT DD SYSOUT=&OUT
//*
//*********************************************
//* ENDEVOR 'ADD' ACTION FOR 'INFOPAC' ELEMENT *
//*********************************************
//*
//STEP04  EXEC PGM=NDVRC1,DYNAMNBR=1500,REGION=4096K,PARM='C1BM3000',
//        COND=(11,LT,STEP02)
//STEPLIB  DD DSN=&LOADLIB,DISP=SHR
//CONLIB   DD DSN=&CONLIB,DISP=SHR
//SYSPRINT DD SYSOUT=&OUT
//SYSUDUMP DD SYSOUT=&OUT
//SORTWK01 DD UNIT=SYSDA,SPACE=(CYL,(2,1))
//SORTWK02 DD UNIT=SYSDA,SPACE=(CYL,(2,1))
//SORTWK03 DD UNIT=SYSDA,SPACE=(CYL,(2,1))
//SORTWK04 DD UNIT=SYSDA,SPACE=(CYL,(2,1))
//C1TPDD01 DD UNIT=SYSDA,SPACE=(CYL,5),
//            DCB=(RECFM=VB,LRECL=260,BLKSIZE=6160)
//C1TPDD02 DD UNIT=SYSDA,SPACE=(CYL,5),
//            DCB=(RECFM=VB,LRECL=260,BLKSIZE=6160)
//C1TPLSIN DD UNIT=SYSDA,SPACE=(CYL,5),
//            DCB=(RECFM=FB,LRECL=80,BLKSIZE=6160)
//C1TPLSOU DD UNIT=SYSDA,SPACE=(CYL,5)
//PAC7TY   DD DSN=&INDEX..&ROOT.&FILE.TY,DISP=SHR
//PAC7PU   DD DSN=&INDEX..&ROOT.&FILE.UP,DISP=SHR
//PAC7UP   DD DSN=&INDEX..&ROOT.&FILE.UP,DISP=SHR
```

```
//PAC7UQ   DD DSN=&INDEX..&ROOT.&FILE.UQ,DISP=SHR
//C1PLMSGS DD SYSOUT=&OUT
//C1MSGS1  DD SYSOUT=&OUT
//C1PRINT  DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,BLKSIZE=6171)
//SYSABEND DD SYSOUT=&OUT
//SYSOUT   DD SYSOUT=&OUT
//PTRACE   DD SYSOUT=&OUT
//INFOPAC  DD DSN=*.STEP01.SYSUT2,DISP=(OLD,DELETE)
//BSTIPT01 DD DUMMY
//*
//*********************************************
//*    BUILD 'DELETE' SCL FOR 'VA PAC' AND     *
//* 'INFOPAC' ELEMENTS IF ERROR(S) IN STEP03   *
//*********************************************
//*
//ISTEP05 IF (STEP02.RC < 12 AND STEP04.RC > 11) THEN
//STEP05  EXEC PGM=DFSRRC00,REGION=$REGSIZ,
//        PARM=(DLI,PNDV20,PNDV20$SUG,&BUF,
//        &SPIE&TEST&EXCPVR&RST,&PRLD,
//        &SRCH,&CKPTID,&MON,&LOGA,&FMTO,,,&DBRC,&IRLM)
//STEPLIB  DD DSN=&RESLIB,DISP=SHR
//         DD DSN=&STEPLIB,DISP=SHR
//DFSRESLB DD DSN=&RESLIB,DISP=SHR
//IMS      DD DSN=&PSBLIB,DISP=SHR
//         DD DSN=&DBDLIB,DISP=SHR
//DDSNAP   DD SYSOUT=&OUT
//PROCLIB  DD DSN=&PROCLIB,DISP=SHR
//IEFRDER  DD DUMMY,
//            DCB=(RECFM=VB,BLKSIZE=1920,LRECL=1916,BUFNO=2)
//SYSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//            BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSUDUMP DD SYSOUT=&OUT,DCB=(RECFM=FBA,LRECL=121,
//            BLKSIZE=605),SPACE=(605,(500,500),RLSE,,ROUND)
//IMSMON   DD DUMMY
//DFSVSAMP DD DSN=&INDEXP..&ROOT.&ROOT.SY(DFSVSAM8),DISP=SHR
//*:STEPCAT  DD DSN=&SYSTCAT,DISP=SHR
//*:         DD DSN=&VSAMCAT,DISP=SHR
//SYSIN    DD DUMMY
//SYSUT1   DD DUMMY
//SYSUT2   DD DSN=*.STEP01.SCLDEL,DISP=(MOD,PASS)
//PAC7AE$SUF DD DSN=&INDEXX..&ROOTX.&ROOTX.AE,DISP=SHR
//SYSOUT   DD SYSOUT=&OUT
//SYSPRINT DD SYSOUT=&OUT
//ESTEP05  ENDIF
//*
```

8. THE VA PAC-ENDEVOR COMPONENTS

8.1. INTRODUCTION

INTRODUCTION

The VA Pac-ENDEVOR interface manages permanent data in on-line (TSO FOREGROUND) or in batch mode.

Two types of resources are then necessary:

. Libraries to store the programs which make up the system and the parameters required for its operations:

A load module library
A parameter library

. Permanent files to materialize the data manipulated by the programs previously defined. These files are divided into two categories:

'System' files which are not modified by the use of VA Pac-ENDEVOR,

'Evolving' files which are manipulated by the user. Their size varies depending on the updates that are performed.

8.2. DSNAMES ASSIGNMENTS

ATTRIBUTION OF DSNAMES

The attribution of DSNAMEs is done according to the following principles:

. The first index level(s) is (are) represented by the &INDEX symbolic parameter for VSAM files, or by the &INDEXP and &INDEXQ symbolic parameters for non-VSAM files. These parameters can be assigned the same value. The &INDEXX parameter is reserved for VA Pac files.

. The last index level is the file name. This name is always 6 characters long; its constitution varies according to the nature of the file concerned:

```
- &ROOT.&ROOT.xx
   for 'system' files,

- &ROOT.&FILE.xx
   for evolving files,
```

&ROOT is a symbolic parameter on two characters identifying the VA Pac-ENDEVOR system, &FILE is a symbolic parameter on two numeric characters identifying the VA Pac-ENDEVOR database concerned and 'xx' is a logical name on two characters characteristic of the file.

Two parameters call the catalogs:

&SYSTCAT indicating the catalog of VSAM files belonging to the VA Pac-ENDEVOR system, &VSAMCAT the catalogue of VSAM files belonging to a VA Pac-ENDEVOR database. These two parameters may be assigned the same value.

8.3. LOAD-MODULES LIBRARY

LOAD MODULE LIBRARY

.Organization	: PDS
.DSN	: (defined during implementation)
.DCB	: (RECFM=U,BLKSIZE=6144)
.Size	: 40 tracks (3390 disk)

The load module library contains the following programs:

CODE	PROC	COMMENTS
C1UEXT02	(1)	EXIT 2
C1UEXT03	(1)	EXIT 3
PBBTST	CIND	Inter-Environment integrity validation
-	GPND	Parameter valorization
-	MEND	Formatting of VA Pac update trans.
PBBT98	CIND	
-	GPND	
-	MEND	
PNCI10	CEND	Intra-ENDEVOR integrity validation
-	RRND	Existing data retrieval
PNCI20	CEND	
PNDC10	CIND	Inter-Environment integrity validation
PNDC50	-	
PNDC60	-	
PNDC70	-	
PNDC90	-	
PNDJ10	JRND	Retrieval of archived trans. journal
PNDM05	JRND	
-	MEND	Formatting of VA Pac update trans.
-	RPND	'VA Pac' creation (existing retrieval)
PNDM10	JRND	
-	MEND	
-	RPND	
-	CIND	
PNDM50	MEND	
PNDM60	-	
PNDR20	RRND	
PNDR30	-	
PNDR35	-	
PNDR40	RPND	
PNDU10	LSND	Library and session table update
PNDU20	TYND	ENDEVOR TYPE table update
PNDU30	LSND	Table printing
-	RNTD	
-	TYND	

CODE	PROC	COMMENTS
PNDV10	GPND	Parameter value assignment
PNDV20	UPND	Import of VA Pac elements into Endevor
PNINUQ	JRND	
PNINUQ	MEND	
PNPR10	(2)	CONWIN + BSTIPT01 for 'INFOPAC' types
PNPR11	(2)	Copy of seq. update trans. on VSAM file
PNRJ10	JJND	Retrieval of arch. journal 1.6 --> 2.5
PNRPEU	RPEU	Retrieval of User Entities 1.6 --> 2.5
PNRPTS	RTND	Retrieval of 'TS' & 'TY' 8.0.2 --> 2.5
PNTRAN	(2)	Processing of INFOS for TRANSFER action
PNXDAT	INND	List of program dates
PRMSYS	CIND	
-	GPND	
-	MEND	
-	RPND	
PTU001	all	Reception of user input

(1): These programs (EXITS) are not called by any procedure but are used by ENDEVOR.
(see 'ENDEVOR/MVS' in the Computer Associates docu_ mentation).

(2): These programs are called by processors:
'GENERATE - VA Pac and Infopac', 'MOVE - VA Pac',
'DELETE - VA Pac'.

8.4. PARAMETERS LIBRARY

PARAMETER LIBRARY: SY

```
.Organization  : PDS
.DSN           : &INDEXP..&ROOT.&ROOT.SY
.DCB           : (RECFM=FB,LRECL=80,BLKSIZE=6080)
.Size          : 3 tracks (3390 disk)
```

The VA Pac-ENDEVOR parameter library contains the input of the various utilities used when the VA Pac-ENDEVOR system is operating.

It contains:

. The DEFINE of VSAM files:

The DFxxxxxx name (with xxxxxx: suffix of the specific file) identifies the DELETE / DEFINE of each VSAM file used in VA Pac-ENDEVOR.

Data concerning the catalog used, disks, space, etc. is initialized according to the initial parameterizing defined during installation and can be modified later - if necessary - by the VA Pac-ENDEVOR system administrator.

. Usual VSAM manipulations:

The VERIFYxx and REPROxx names identify the VERIFY and REPRO commands applied to VA Pac-ENDEVOR files.

. The BLDG of 'QU' generation file

The BLxxyyQU name identifies the building command of the 'QU' backup index of 'UQ' VA Pac update file.

. The files to be added to the TSO LOGON

The 'ISPLOGON' name identifies the VSAM files used by the VA Pac-ENDEVOR system. These files must be defined in the user LOGON of the TSO procedure.

. The allocation of the EXITS TRACE file to be added to the connection procedure to ISPF.

The 'ISPF' name identifies the allocation command of the EXITs TRACE file which must be defined in the user's ISPF connection procedure.

. TSO messages of the VA Pac-ENDEVOR system

The 'CIUU$msgsx' name (see the meaning of the '$MSGSX' parameter in the table of parameters) represents the TSO messages of the VA Pac-ENDEVOR system. These messages must be copied in the ENDEVOR 'ISPMLIB' library.

. The processors used by the system

- 'PRCSGENI': JCL lines of the 'GENERATE' type processor for INFOPAC type elements.

- 'PRCSGEPP': JCL lines of the 'GENERATE' type processor for VA Pac ELEMENT types (only non-compilable entities).

- 'PRCSGENP': JCL lines of the 'GENERATE' type processor for VA Pac ELEMENT types (only compilable entities).

- 'PRCSDELP': JCL lines of the 'DELETE' type processor for VA Pac ELEMENT types.

- 'PRCSMOVP': JCl lines of the 'MOVE' type processor for VA Pac ELEMENT types.

  NOTE: The implementation of these processors is explained in the 'INSTALLATION' chapter, PHASE 1.

. The source of PSBs

NOTE

Any modification of the file characteristics must be made in this parameter library.

9. INSTALLATION

9.1. INTRODUCTION

WARNING

The preparation, installation and operation of the system uses OS and VSAM standard functions. The user is entirely responsible for any adaptation other than those mentioned in PHASE 2 of the JCL generation process.

In particular, all modifications on JCL and executable modules' names etc., must be performed with caution. This kind of modifications often causes abnormal conditions which are sometimes difficult to detect.

In case of an incident during the execution of an installation phase, the phase must be started again from the beginning without any JCL modification of the specific JOBs.

INTRODUCTION

The VA Pac ENDEVOR system must be installed using the tape (or cartridge) provided by IBM. You must carefully follow the explanations given in this chapter.

The installation process breaks down into three steps:

. Phase 1: the preparation of the environment on site,

. Phase 2: the generation of an installation and test JCL adapted to the site,

. Phase 3: the installation proper.

9.2. INSTALLATION TAPE

INSTALLATION TAPE

The installation tape (6250 BPI, standard labels) contains the following files:

Line	Label	Record	Block	Contents
01	INST.JCL	80	3440	Initial preparation JCL
02	INST.MOD		6144	MM1JCL load-module, JCL preparation utility
03	NDVR.JCL	80	3440	Skeleton JCL
04	NDVR.LOAD		6144	Load-modules
05	NDVR.PACVINS	117	9360	VA Pac update transactions for User Entities
06	NDVR.PSB		6144	PSBs library

9.3. PHASE 1: ENVIRONMENT PREPARATION

PHASE 1: ENVIRONMENT PREPARATION

This phase prepares the environment for the installation of the VA Pac-ENDEVOR system. Referring to the system technical characteristics as described in the first chapters of this manual, in this phase:

. Choose the various prefixes, suffixes and roots to be allocated to the VA Pac-ENDEVOR codes and files,

. Choose the VSAM catalogs and reserve the necessary disk space.

Allocation of processors to the PROCESSOR GROUP

Reminder: The VA Pac-ENDEVOR system is constituted of two types of elements:

. 'VA Pac' type: All VA Pac entities (compilable or not) are stored in ENDEVOR TYPES called 'VA Pac' types. Several VA Pac types can be created.

. 'INFOPAC' type: Each VA Pac type element is associated with an INFOPAC type, which is in its turn associated with this VA Pac type. The INFOPAC type is transparent to the user. It cannot be modified, except on some specified processors.

Three PROCESSOR GROUPs must be defined for the VA Pac-ENDEVOR system. Two of these PROCESSOR GROUPs are allocated to each VA Pac type and one is allocated to each INFOPAC type.

These PROCESSOR GROUPs must be defined in the following way:

1. Non-compilable VA Pac entities (VA Pac type)

```
PROCESSOR GROUP:        (user-defined name)

GENERATE PROCESSOR:     (5)
DELETE PROCESSOR:       (2)
MOVE PROCESSOR:         (3)
```

2. Compilable VA Pac entities (VA Pac type)

```
PROCESSOR GROUP:        (user-defined name)

GENERATE PROCESSOR:     (1)
DELETE PROCESSOR:       (2)
MOVE PROCESSOR:         (3)
```

3. Compilable or non-compilable VA Pac entities (INFOPAC type)

```
PROCESSOR GROUP:        (user-defined name)

GENERATE PROCESSOR:     (4)
DELETE PROCESSOR:       *NOPROC*
MOVE PROCESSOR:         *NOPROC*
```

With:

(1) 'PRCSGENP' member of the 'SY' parameters PDS.

This processor compiles and link-edits the generated VA Pac entity, and creates its FOOTPRINT.

NOTE: This JCL must be adapted to the site standards before it is installed in the ENDEVOR system. The PDS name of SYSLIN and that of SYSLMOD (the same) in the 'IEWL' step must correspond to the user load-module library.

(2) 'PRCSDELP' member of the 'SY' parameters PDS.

This processor executes the DELETE action on the INFOPAC type associated with the VA Pac type (See the 'TYND' batch procedure). It also communicates to EXIT3 the VA Pac data of the element to be deleted. This data is contained in its homonym, but its type is INFOPAC. EXIT3 then relates this data to the data of the ENDEVOR context and prepares the VA Pac database update transactions.

(3) 'PRCSMOVP' member of the 'SY' parameters PDS.

This processor executes the MOVE action on the INFOPAC type associated with the VA Pac type: it transfers it from STAGE 1 to STAGE 2 (See 'TYND' batch procedure). It also communicates to EXIT3 the VA Pac data of the element to be transferred. This data is contained in its homonym, but its type is INFOPAC. EXIT3 then relates this data to the data of the ENDEVOR context and prepares VA Pac database update transactions.

(4) 'PRCSGENI' member of the 'SY' parameters PDS.

This processor executes the ADD action on the INFOPAC type associated with the VA Pac type. IT also modifies the compiled load module FOOTPRINT (when the entity is compilable) in order to allocate the FOOTPRINT of the current INFOPAC to it. It finally communicates to EXIT3 the VA Pac data of the element to be transferred. This data is contained in its homonym, but its type is INFOPAC. EXIT3 then relates this data to the data of the ENDEVOR context and prepares VA Pac database update transactions.

NOTE: The load module library name mentioned on line DD "//IN" of the 'GENOA' step must be adapted (compilable entities).

(5) 'PRCSGEPP' member of the 'SY' parameters PDS.

This 'GENERATE' type processor is only used for the 'TRANSFER' action of a non-compilable VA Pac element. It generates and executes the 'TRANSFER' action of the 'INFOPAC' element.

IMPOR TANT : All these processors can be used without modification. They can also be customized to meet site's requirements. Proceed with caution and carefully follow the recommendations given in this manual (see Chapter DESCRIPTION OF THE INTERFACE ELEMENTS, Subchapter INTERFACE PROCESSORS).

9.4. PHASE 2: GENERATION OF THE INSTALLATION JCL

PHASE 2: GENERATION OF THE INSTALLATION JCL

This phase generates the JCLs for the system installation procedure. The first file on the installation tape (DSM=INST.JCL,RECFM=FB,LRECL=80) must be read and loaded onto the site editor. This file contains the JCLs of the JOBs to be executed:

1. PACBASE0: Duplication of the installation tape onto a tape provided by the user site (backup of the system).

2. PACBASE1: Copy of the MM1JCL utility into a library of executable modules.

3. PACBASE2: Execution of the MM1JCL utility which will transform the JCL skeleton from the installation tape into a JCL adapted to the site according to the given input values.

INSTALLATION OF THE COMPLETE JCL

The MM1JCL module reads the skeleton JCL file and outputs an adapted JCL. It uses commands which allow the user to:

. Select portions of the skeleton JCL called 'JCL modules',

. Parameterize the skeleton in order to obtain a JCL which may be operational with a minimum of modifications,

. Select installation variants to be able to generate the JCL needed for specific processing, depending on the site or the installation conditions,

. Add before and/or after lines in order to separate the JCL modules.

This step can be executed as many times as necessary to produce the correct JCL.

MM1JCL provides a list for each JCL module created, with the parameters and the requested variants.

NOTE ON THE PRINTED OUTPUT:
The skeleton JCL parameters having the $xxxx format, when MM1JCL encounters, during the execution, a $ character which does not correspond to a defined parameter, it indicates it with the error message: 'UNKNOWN SYMBOLIC PARAMETER'. This message does not stop the execution and should be ignored; it applies to '$' characters in the JCL stream processed by MM1JCL which are not parameters.

Once the JCL is created, it can be retrieved into a text editor. At this point, the actual installation procedure can begin.

It is recommended that as many members as there are JCL modules be created in the text editor; each module constitutes a logical step in the installation procedure.

```
CODING OF MM1JCL COMMANDS

===SELM mmmm1 mmmm2 ...    .Selection of JCL modules
                            mmmm1=name of JCL module
                            mmmm2=name of JCL module
                            etc.

===PRM PPPP=pppp           .Parameter
                            pppp=name of parameter
                            pppp=value of parameter

NOTE: On ===PRM lines, a comment can be entered and must
      be preceded by a period.

===BEGMOD    Insertion of lines at the beginning of module
DDDD1
.....        (Lines to be inserted before each module)
DDDDn
===ENDMOD    Insertion of lines at the end of module
FFFF1
.....        (Lines to be inserted after each module)
FFFFn
```

INSTALLATION DEFAULT OPTIONS

. VARIANTS (===SELV):

All available variants are selected.
IMPORTANT: The user must delete the lines which correspond to the variants not installed on the site.

. PARAMETERS (===PRM):

The values indicated here are examples. So,the user must set them according to the site's specific needs.

. MODULES (===SELM):

No selection of modules is provided. All modules (corresponding to the variants) are selected.

. JCL MODULE FIRST LINE (===BEGMOD):

A line: ./ ADD NAME=<JCL module>

This makes it possible to split the JCL into a PDS by IEBUPDTE utility.

Because of this default option, the './' characters found in the JCL modules have been replaced by ':/' so that they are not taken into account in this IEBUPDTE.

Once the JCL is split up, the ':/' characters must be changed back to './' before executing jobs which contain IEBUPDTE.

. VSAM catalogs are entered as comments in the installation JCL:

```
- in the DELETE/DEFINE, as   /*:CATALOG ($VCAT) */
                        or   /*:CATALOG ($SCAT) */

- in the JCL STEPCATs,  as   //*:STEPCAT DD
                     or/and  //*:        DD

- in the procedure parameters as //*:      VSAMCAT='$CATU'
                              or //*:      SYSTCAT='$CATV'
```

If these parameters are not required on the site, the JCL does not need to be changed.

If these parameters are required, the comment lines must be transformed into active lines by:

- Substituting all '//*:' by '//'
- Substituting all '/*:' and '*/' with blanks.

These transformations should be done before splitting the JCL into different members to avoid repetitions.

JCL MODULES TABLE: ===SELM mmmm1 mmmm2 ... mmmmn

mmmm	Content	Nature
EXIT	Link-Edit EXIT2 and EXIT3 in authorized library.	JCL
INSL	Compilation dates of installed modules.	JCL
LOAD	Down-loading of load-modules.	JCL
MSGS	Chargement des messages utilisateur Down-loading of TSO/VA Pac-ENDEVOR user messages.	JCL JCL
PREP	Initial allocations and downloading	JCL
PROC	Downloading batch procedures	JCL
PSB	Downloading PSBLIB	JCL
SY	Downloading parameters' PDS	JCL
VINS	VA Pac updating of VA Pac-ENDEVOR system user entities.	JCL

(1) This JCL module includes retrieval 1.6 -> 2.5 (Year2000)

PARAMETERS TABLE

```
===PRM PPPP=pppp               .Comments
```

Code PPPP	Meaning	Default pppp
	On the JOB lines	
PRFJ	Prefix of job names (max. 4 characters)	NDVR
CCPT	Job accounting code	<>
CLASSJ	Jobs execution class	1
MSGCL	JCL output class	A
	Coding of files DSN	
	The names of VA Pac-ENDEVOR permanent files (except the load-modules libraries) are formatted as:	
	INDEXQ.XXNNSS : Non VSAM user file	
	INDEX.XXXXSS : VSAM system file	
	INDEXP.XXXXSS : Non VSAM system file	
	IND-- Index of file names:	
INDEX	VSAM system	SYS.NDV150
INDEXP	Non VSAM system (SAM, PDS)	SYS.NDV150
INDEXQ	Non VSAM user (SAM)	USR.NDV150
	The permanent VA Pac files have a similar coding.	
INDEXX	VSAM system	SYS.NDV150
	XX=ROOT, NN=FILE, SS=file code	
ROOT	Root of VA Pac-ENDEVOR system (2 characters different from 'XX')	C1
FILE	Nb of VA Pac-ENDEVOR user database	00

PARAMETERS TABLE: (Cont'd)

Code	Meaning	Default
PPPP		pppp
ROOTX	Root of VA Pac system (1) (2 characters different from 'XX')	C1
FILEX	Nb of VA Pac user database (1)	00
	On DD lines	
OUT	SYSOUTs printing cards,	X
UTAPE	UNIT of the installation tape, copy of the tape supplied.	TAPE
TAPEI	Name of the installation tape, copy of the tape supplied.	<>
UWK	UNIT of work files used.	SYSDA
UNITO	UNIT of non VSAM user files.	3390
UNITP	UNIT of non VSAM system files.	3390
UNITU	UNIT of VSAM user files.	3390
UNITV	UNIT of VSAM system files.	3390
VOLP	Name of volume of non VSAM system files.	<>
VOLV	Name of volume of VSAM system files.	<>
VOLO	Name of volume of non VSAM user files.	<>
VOLU	Name of VSAM user files.	<>

PARAMETERS TABLE: (Cont'd)

Code	Meaning	Default
PPPP		pppp
	Various parameters	
RADP	Prefix of procedures names to be installed (max. 4 characters)	N160
CATU	Catalogue of VSAM user VA Pac-ENDEVOR files	<>
CATV	Catalogue of VSAM system VA Pac-ENDEVOR files	<>
MODB	DSNAME of VA Pac-ENDEVOR load-module librarie	LOAD.NDV150
BIBP	DSNAME of procedures lib. in which the VA Pac-ENDEVOR procedures are catalogued	SYS1.PROCLIB
BIBT	DSANME of sort library used	SYS1.SORTLIB
BCOB	DSName of Cobol/VS routine library	SYS1.VSCLLIB
CONLIB	DSName of ENDEVOR CONLIB library	NDVR.CONLIB
LDLIB	DSName of ENDEVOR LOADLIB library (authorized library)	NDVR.LOADLIB
MSGLIB	DSName of ENDEVOR ISPMLIB library	NDVR.ISPMLIB
MSGSX	VA Pac-ENDEVOR message suffix in the ENDEVOR ISPMLIB lib.	50
MSGSXH	(same as 'MSGSX' parameter but in hexadecimal)	F5F0
PSBLIB	VA Pac ENDEVOR PSBs LIBRARY	PACBP.PSB
DBDLIB	VA Pac ENDEVOR DBD LIBRARY	PACBP.DBD
RESLIB	IMS RESLIB	RESLIB
PRCLIB	IMS PROCLIB	IMS.PROCLIB
REGSIZ	REGION SIZE FOR BATCH PROCEDURES	1536K
IMSID	BMP PROCEDURES IMSID PARAMETER	IM16
DBRC	BATCH PROCEDURE DBRC USE (N=NO)	N
IRLM	BATCH PROCEDURE IRLM USE (N=NO)	N
SUG	BATCH PSBs SUFF. (2 CHARACTERS)	12
SUF	BATCH DBDs SUFF. (2 CHARACTERS)	12

(1) : The 'ROOTX' and 'FILEX' parameters are specific to the VA Pac system. They can be the same as or different from VA Pac-ENDEVOR 'ROOT' and 'FILE'. However, it is recommended to assign them, as far as possible, the same values. They must have the same values as those defined during the installation of the VA Pac system which is independent of that of VA Pac-ENDEVOR.

Notes

'<>' signals a parameter that MUST be coded.

The parameter values containing special characters must be delimited by simple quotes.

The comments on the ===PRM lines must not exceed column 72. They must be preceded by a '.'.

LINES BEFORE AND AFTER JCL MODULES

```
===BEGMOD
DDDD1     )
.....     ) Lines to be inserted before each JCL module
DDDDn     )

===ENDMOD
FFFF1     )
.....     ) Lines to be inserted after each JCL module
FFFFn     )
```

The purpose of these lines is to split the JCL file created by the utility into as many members as there are JCL modules.

These lines may be inserted as input in MM1JCL if the default option is not appropriate (see paragraph 'INSTALLATION DEFAULT OPTIONS' in this subchapter).

The utility adds DDDD1 to DDDDn lines before each JCL module and FFFF1 to FFFFn lines after each JCL module.

```
$CO***********  VA PAC-ENDEVOR 2.5   IMS  $VV  $DATE **************
//PACBASE0 JOB (---),'TAPE',CLASS=D,MSGCLASS=A
//ALLOC   EXEC PGM=IEHINITT
//TCGI     DD DISP=SHR,UNIT=(TAPE,,DEFER),VOL=(,RETAIN,SER=$BDECGI)
//TINST    DD DISP=SHR,UNIT=(TAPE,,DEFER),VOL=(,RETAIN,SER=úúúúúú),
//            DCB=DEN=3
//SYSPRINT DD SYSOUT=A
//SYSIN    DD *
TINST INITT SER=úúúúúú,OWNER='úúúúúú',DISP=REWIND
/*
//PACCOP  PROC INDEX='$INPRO',NAME=XXX,LAB=N
//GENER   EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=A
//SYSIN    DD DUMMY
//SYSUT1   DD DSN=&INDEX..&NAME,DISP=SHR,
//            VOL=(,RETAIN,REF=*.ALLOC.TCGI),LABEL=&LAB
//SYSUT2   DD DSN=&INDEX..&NAME,DISP=(,KEEP),
//            VOL=(,RETAIN,REF=*.ALLOC.TINST),LABEL=&LAB,
//            DCB=*.SYSUT1
//        PEND
//*
//STEP01  EXEC PACCOP,LAB=01,NAME=JCL,INDEX=INST
//STEP02  EXEC PACCOP,LAB=02,NAME=MOD,INDEX=INST
//STEP03  EXEC PACCOP,LAB=03,NAME=JCL,INDEX=NDVR
//STEP04  EXEC PACCOP,LAB=04,NAME=LOAD,INDEX=NDVR
//STEP05  EXEC PACCOP,LAB=05,NAME=PACVINS,INDEX=NDVR
//STEP06  EXEC PACCOP,LAB=06,NAME=PSB,INDEX=NDVR
//*
//
```

```
//PACBASE1 JOB (---),'UTI',CLASS=D,MSGCLASS=A
//*
//COPIE   EXEC  PGM=IEBCOPY
//SYSPRINT DD SYSOUT=A
//SYSUT3   DD UNIT=SYSDA,SPACE=(TRK,10)
//SYSUT4   DD UNIT=SYSDA,SPACE=(TRK,10)
//IM       DD DISP=OLD,UNIT=TAPE,VOL=(,RETAIN,SER=úúúúúú),
//            DSN=INST.MOD,LABEL=02
//OM       DD DISP=SHR,DSN=úúú.úúú.úúú
//SYSIN    DD *
 C  I=((IM,R)),O=OM
/*
//*
//
```

```
//PACBASE2 JOB (---),'JCL INSTALLATION',CLASS=D,MSGCLASS=A
//*
//ET010   EXEC PGM=IEFBR14
//DD1      DD DSN=úúúúúúúúúúúúúúúúú,        <-- PDS NAME OF THE JOBS
//            UNIT=%%%%,VOL=SER=??????,     <-- UNIT AND VOLUME
//            DISP=(,CATLG),
//            SPACE=(TRK,(10,10,10)),
//            DCB=(RECFM=FB,LRECL=80,BLKSIZE=6080)
//*
//MM1JCL  EXEC PGM=MM1JCL
//STEPLIB  DD DISP=SHR,DSN=úúú.úúú.úúú
//SYSOUT   DD SYSOUT=A
//SYSUT1   DD DSN=NDVR.JCL,DISP=OLD,
//            UNIT=TAPE,VOL=(,RETAIN,SER=úúúúúú),LABEL=03
//SYSUT3   DD UNIT=SYSDA,SPACE=(CYL,(5,2)),DCB=BLKSIZE=4160
//SYSUT4   DD UNIT=SYSDA,SPACE=(CYL,(5,2)),DCB=BLKSIZE=4160
//SYSUT8   DD DUMMY,DCB=BLKSIZE=1370
//SYSUT9   DD DUMMY,DCB=BLKSIZE=1370
//*******************************************************************
//*  --- CREATION OF INSTALLATION JCL THROUGH PROGRAM :  'MM1JCL' ---
//*******************************************************************
//*
//*      -MODIFY THE LIST OF THE SUPPLIED COMMANDS BY ENTERING THE
//*       INSTALLATION VARIANTS (IF  ANY), BY ASKING, IF NECESSARY, A
//*       SELECTION OF PARTS OF INSTALLATION JCL (JCL MODULES), BY
//*       GIVING THE APPROPRIATE VALUES TO THE INSTALLATION PARAMETERS,
//*       AND BY SPECIFYING (IF NECESSARY) THE LINES TO BE ADDED AT
//*       THE BEGINNING OR AT THE END OF EACH JCL MODULE.
//*******************************************************************
//SYSPRM   DD *
 *                                                              A
//SYSUT2   DD DSN=&&PACBASE2,DISP=(,PASS,DELETE),UNIT=úúúúú,
//            SPACE=(TRK,(10,10),RLSE),
//            DCB=(RECFM=FB,LRECL=80,BLKSIZE=6080)
//SYSIN DD *
===PRM PRFJ=NDVR              .JOB NAMES PREFIXES (4 CHARACTERS)
===PRM CCPT=<>                .JOB ACCOUNTING CODES (JOB LINES)
===PRM CLASSJ=1               .JOB EXECUTION CLASS (JOB LINES)
===PRM MSGCL=A                .JCL OUTPUT CLASS (MSGCLASS)
===PRM UTAPE=TAPE             .TAPE UNIT
===PRM TAPEI=<>               .NAME OF THE INSTALLATION TAPE
===PRM OUT=A                  .JOB SYSOUT CLASS
===PRM INDEX='SYS.NDV250'     .SYSTEM VSAM FILES INDEXES
===PRM INDEXX='SYS.NDV250'    .VA PAC SYSTEM VSAM FILES INDEXES
===PRM INDEXP='SYS.NDV250'    .SYSTEM NON VSAM FILES INDEXES
===PRM INDEXQ='USR.NDV250'    .USER NON VSAM FILES INDEXES
===PRM ROOT='P2'              .ROOT OF SYSTEM (2 CHARACTERS)
===PRM ROOTX='P2'             .ROOT OF VA PAC SYSTEM (2 CHARACTE RS
===PRM FILE='00'              .NUMBER OF USER'S DATABASE (2 DIGITS)
===PRM FILEX='00'             .NUMBER OF VA PAC USER'S DATABASE (DIGITS
===PRM RADP='N250'            .PREFIX OF CATALOGUED PROCEDURE NAMES
===PRM VOLV=<>                .VOLUME OF SYSTEM VSAM FILES
===PRM VOLU=<>                .VOLUME OF USER VSAM FILES
===PRM CATV=<>                .CATALOG OF SYSTEM VSAM FILES
===PRM CATU=<>                .CATALOG OF USER VSAM FILES
===PRM UWK=SYSDA              .WORK UNIT
===PRM UNITP=SYSDA            .NON VSAM SYSTEM FILES UNIT
===PRM UNITO=SYSDA            .NON USER VSAM FILES UNIT
===PRM UNITV=SYSDA            .VSAM SYSTEM FILES UNIT
===PRM UNITU=SYSDA            .VSAM USER FILES UNIT
===PRM VOLP=<>                .NON VSAM SYSTEM FILES VOLUME
===PRM VOLO=<>                .NON VSAM USER FILES VOLUME
===PRM MODB='LOAD.NDV250'     .LOAD-MODULES LIBRARY
===PRM CONLIB='NDVR.CONLIB'   .ENDEVOR CONLIB
```

```
===PRM LDLIB='NDVR.LOADLIB'   .ENDEVOR LOADLIB (AUTHORIZED LIBRAR
===PRM MSGLIB='NDVR.ISPMLIB'  .ENDEVOR ISPMLIB
===PRM MSGSX='50'             .ENDEVOR ISPMLIB MSGS SUFFIX (00 ->99)
===PRM MSGSXH='F5F0'          .SAME AS ===PRM MSGSX BUT IN HEXA
===PRM PSBLIB='PACBP.PSB'     .VA PAC-ENDEVOR PSB LIBRARY
===PRM DBDLIB='PACBP.DBD'     .VA PAC-ENDEVOR DBD LIBRARY
===PRM RESLIB='RESLIB'        .RESLIB IMS
===PRM PRCLIB='IMS.PROCLIB'   .PROCLIB IMS
===PRM REGSIZ='1536K'         .REGION SIZE FOR BATCH PROCEDURE
===PRM IMSID='IM25'           .'IMSID' PARAMETER BMP PROCEDURE
===PRM DBRC=N                 .BATCH PROC DBRC USE         N=NO
===PRM IRLM=N                 .BATCH PROC IRLM USE         N=NO
===PRM SUG='22'               .PSB BATCH SUFF (2 CHARACTERS)
===PRM SUF='22'               .DBD BATCH SUFF (2 CHARACTERS)
===PRM BIBP='SYS1.PROCLIB'    .PROCEDURE LIBRARY
===PRM BIBT='SYS1.SORTLIB'    .SORT LIBRARY
===PRM BCOB='SYS1.VSCLLIB'    .COBOL/VS ROUTINE LIBRARY
===BEGMOD
./ ADD NAME=$MODULE
/*
//*
//ET020   EXEC PGM=IEBUPDTE,PARM=NEW
//SYSPRINT DD SYSOUT=$S
//SYSUT1   DD DSN=*.ET010.DD1,DISP=SHR
//SYSUT2   DD DSN=*.ET010.DD1,DISP=SHR
//SYSIN    DD DSN=&&PACBASE2,DISP=(OLD,DELETE)
//
aa
//*********************************************
//* END OF FILE   1 : END OF INITIAL JCL      *
//*********************************************
//*
```

9.5. PHASE 3: INSTALLATION

PHASE 3: INSTALLATION PROCESS

The JCL obtained during phase 2 breaks down into 8 JOBs (each corresponding to a JCL module) which make up the installation of the VA Pac-ENDEVOR system.

1. Loading the load-modules.
2. Loading the system parameters PDS.
3. Loading the operational batch procedures.
4. Loading the PSBLIB.
5. Initial preparation of the files.
6. Loading the EXITs in an authorized library.
7. Loading the TSO/VA Pac-ENDEVOR messages in the site's ENDEVOR library.
8. Creating the VA Pac-ENDEVOR system user entities in VA Pac.
9. List of installed programs.

1. LOADING THE LOADS-MODULES:

This loading is made up of a '$PRFJ.LOAD' job which comprises the following steps:

ET010 : IDCAMS

DELETE of VA Pac-ENDEVOR load-modules library.

ET020 : IEFBR14

VA Pac-ENDEVOR load-modules libraries allocation.

ET030 : IEBCOPY

Loading of the load-modules.

ET040 : IMASPZAP

ZAP of EXIT2. It concerns the member's name of the VA Pac-ENDEVOR 'ISPMLIB' messages library.
Only to be executed if the value chosen for the $MSGSXH parameter is not '50'.

2. LOADING THE SYSTEM PARAMETERS PDS

This loading is made up of a '$PRFJ.SY' job which comprises the following steps:

ET010 : IDCAMS

DELETE of SY parameters PDS.

ET020 : IEFBR14

Allocation of parameters PDS.

ET030 : IEBUPDTE

- The input for the SORT utilities.

- The definitions (DELETE/DEFINE), the verifications (VERIFY) and the REPRO of the VSAM files.

- The BLxxyyQU member containing the index building command of the 'QU' backup of the VA Pac 'UQ' update file.

- The 'ISPLOGON' member containing the three VSAM files used by the VA Pac-ENDEVOR system and to be defined in the user LOGON TSO procedure.

- The 'CIUU$msgsx'member (see the meaning of the '$MSGSX' parameter in the parameters table). It is made up of the TSO messages of the VA Pac-ENDEVOR system and is copied in the ENDEVOR's 'ISPMLIB' library.

- The 'PRCSGENI' member: JCL of the 'ADD'-type processor for the 'INFOPAC'-type elements (cf. PHASE 1).

- The 'PRCSGENP' member: JCL of the 'ADD'-type processor for the 'VA Pac'-type elements (cf. PHASE 1).

- The 'PRCSDELP' member: JCL of the 'DELETE'-type processor for the 'VA Pac'-type elements (cf. PHASE 1).

- The 'PRCSMOVP' member: JCL of the 'MOVE'-type processor for the 'VA Pac'-type elements (cf. PHASE 1).

3. LOADING THE OPERATIONS PROCEDURES

This loading comprises a '$PRFJ.PROC' JOB which carries out, via IEBUPDTE, the cataloging in the PROCLIB of all the operations batch procedures.

Each procedure is a '$radp.NNNN' coded member, where '$radp' is the root chosen when generating the JCL and NNNN the standard procedure name. The procedures are described in other chapters of this manual.

4. LOADING THE PSBLIB

This loading comprises a '$PRFJ.PSB' JOB which carries out, via IEBCOPY, the loading of the PSBs in the PSBLIB.

The installation tape contains the PSBs in the form of objects and sources.

This loading is to be executed only if the chosen value in the generation of the JCL for:

.the DBDs suffix (SUF parameter) is: '22'

.the system root (ROOT parameter) is: 'P2'

.the batch PSBs suffix (SUG parameter) is: '22'

If not, the sources must be recompiled. These sources were cataloged in the 'SY' parameters file, under the name XXXXXX$SUG where XXXXXX is equal to: PNCI20 PNDC10 PNDC90 PNDJ10 PNDM05 PNDM10 PNDM50 PNDU10 PNDU20 PNDU30 PNDV10 PNDV20 PNRPTS PNXDAT PNRPEU

5. INITIAL PREPARATION OF THE FILES

This preparation, which must only be executed during the initial installation includes a '$PRFJ.PRE' JOB which comprises the following steps:

ET010 : IDCAMS

Table initialization:

- libraries/sessions........'TS'
- ENDEVOR types of VA Pac

  and 'INFOPAC' elements....'TY'

Work files initialization:

- VA Pac context of elements in

  ENDEVOR (update preparation)..'UP'
- VA Pac & ENDEVOR context of elements

  in ENDEVOR (update prep.)..'UQ'

ET020 : PNINUQ

Loading the RRDS 'UQ' file.

ET030 : IDCAMS

DELETE the model DSCB.

ET040 : IDCAMS

BLDG data-group index and initialization
of 'QU' file (backup of VA Pac update file: 'UQ')

ET050 : IEBGENER

Loading the 'QU' file

6. LINK-EDIT OF EXITS IN AN AUTHORIZED LIBRARY:

The VA Pac-ENDEVOR system is made up of 2 EXITS (EXIT2 and EXIT3) which must be linked with the 'EPC1EUXT' ENDEVOR DRIVER.

This is carried out via the '$PRFJ.EXT' JOB which includes the following steps:

ET010 : IEWL : EXIT2 link.

ET020 : IEWL : EXIT3 link.

Important : After the execution of this job do not forget to define the EXITS in ENDEVOR by adding them via the aC1UEXIT macro-command lines of the ENDEVOR 'BC1JXITS' JCL.

7. LOADING THE TSO/VA PAC-ENDEVOR MESSAGES:

The '$PRFJ.MSG' job loads the TSO user messages of the VA Pac-ENDEVOR system in the ENDEVOR 'ISPMLIB' message library from the 'CIUU$MSGSX' member present in the 'SY' parameters PDS. (see the meaning and possible values of the '$MSGSX' parameter in the parameters table).

This job comprises the following step:

ET010 : IEBCOPY: Loading of the 'CIUU$MSGSX' member in the ENDEVOR 'ISPMLIB' library.

Note : The member name respects the standards imposed by LEGENT company.

8. UPDATING THE VA PAC DATABASE:

This JOB creates in the VA Pac database, the user entities used by the VA Pac entities called in an ENDEVOR context.

It also retrieves the archived journal and the User Entity Occurrences already existing in VA Pac for users switching from release 1.6 to 2.5.

ET010 : ????VINS

update of ENDEVOR user entities

.NDVLM User Entity and .NDVRL Relationship

The .NDVLM user entity records information of the ENDEVOR contexts where the VA Pac generated objects are managed.

The .NDVRL relationship establishes the link between the User Entity Occurrence and the VA Pac entity itself.

They are described as batch update transactions that can be used by the VA Pac UPDT procedure.

They must be found in the libraries where these ENDEVOR contexts are managed. It is recommended that the .NDVLM User Entity and the .NDVRL Relation should be created in the highest library of the VA Pac Database or in the central library by the interface Administrator. These entities are thus locked under the Interface Administrator's code, and their occurrences cannot be updated by other users.

If the Administrator wants to manage these contexts in frozen sessions, he must repeat the operation as many times as there are used frozen sessions.

Steps to be executed only for users switching from release 1.6 to 2.5.

ET020 : $RADP.JJND (Retrieval of archived journal 1.6)

ET020 : ????SAVE (Backup of VA Pac Database)

ET020 : $RADP.REND (Retrieval of VA Pac/ENDEVOR UEOs)

9. LIST OF INSTALLED PROGRAMS:

===MOD NDVRINSL

This list is obtained by the $PRFJ.INSL job which executes the '$RADP.INND'procedure.

This step is not compulsory but recommended. It includes:

. the list of the VA Pac-ENDEVOR system load-modules with their compilation date.

This list must be kept and sent to the VA Pac Technical Support in case malfunctions are detected in the VA Pac System.

```
//$PRFJ.LOAD JOB ($CCPT),'LOAD-MODULES',CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
//****************************************************
//*         --- COPY LOAD-MODULES IN PDS ---         *
//****************************************************
//*
//*--------------------------------------------------*
//*               I M P O R T A N T                  *
//*                  -----------                     *
//*  IT IS NOT NECESSARY TO EXECUTE THE 'ET040' STEP *
//*     IF THE SUFFIX OF THE VA PAC USER MESSAGES    *
//*     MEMBER OF ENDEVOR 'ISPMLIB' LIBRARY CONTAINS *
//*     THE VALUE '50' (CHECK 'MSGSX' PARAMETER).    *
//*  IF THE SUFFIX IS MODIFIED, THIS STEP MUST BE    *
//*     EXECUTED. IN ORDER TO DO SO, THE 'MSGSHX'    *
//*     MUST BE CODED WITH THE CORRESPONDING HEXA-   *
//*     DECIMAL VALUE OF THE NEW SUFFIX.             *
//*     FOR INST: IF THE NEW SUFFIX IS: MSGSX='89',  *
//*               THE 'MSGSXH' PARAMETER MUST BE     *
//*               CODED: MSGSXH='F8F9'.              *
//*--------------------------------------------------*
//ET010   EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=$OUT
//SYSIN    DD *
  DELETE ($MODB)
//*
//ET020   EXEC PGM=IEFBR14
//DDA      DD DSN=$MODB,DISP=(,CATLG,DELETE),UNIT=$UNITP,
//            VOL=SER=$VOLP,
//            SPACE=(TRK,(40,10,10)),
//            DCB=(RECFM=U,BLKSIZE=6144)
//*
//ET030   EXEC PGM=IEBCOPY
//SYSPRINT DD SYSOUT=$OUT
//SYSUT3   DD UNIT=$UWK,SPACE=(TRK,20)
//SYSUT4   DD UNIT=$UWK,SPACE=(TRK,20)
//INB      DD DSN=NDVR.LOAD,DISP=OLD,UNIT=$UTAPE,LABEL=(04,SL),
//            VOL=(,RETAIN,SER=$TAPEI)
//OUTB     DD DSN=$MODB,DISP=SHR
//SYSIN    DD *
  COPY INDD=((INB,R)),OUTDD=OUTB
  S M=C1UEXT02
  S M=C1UEXT03
  S M=PACMODG3
  S M=PBTPST
  S M=PNCI10
  S M=PNCI20
  S M=PNDC10
  S M=PNDC50
  S M=PNDC60
  S M=PNDC70
  S M=PNDC90
  S M=PNDJ10
  S M=PNDM05
  S M=PNDM10
  S M=PNDM50
  S M=PNDM60
  S M=PNDR20
  S M=PNDR30
  S M=PNDR35
  S M=PNDR40
  S M=PNDU10
  S M=PNDU20
  S M=PNDU30
  S M=PNDV10
  S M=PNDV20
  S M=PNINUQ
  S M=PNPR10
  S M=PNPR11
  S M=PNRJ10
  S M=PNRPTS
  S M=PNRPEU
  S M=PNTRAN
  S M=PNXDAT
  S M=PTU001
/*
//ET040   EXEC PGM=IMASPZAP
//SYSPRINT DD SYSOUT=$OUT
//SYSLIB   DD DSN=$MODB,DISP=SHR
 NAME C1UEXT02 C1UEXT02
 VER 0303 F5F0
 REP 0303 $MSGSXH
//*
//
```

```
//$PRFJ.SY JOB ($CCPT),'LOAD PARAM.',CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
//*
//**************************************************************
//*       LOADING PACBASE-ENDEVOR PARAMETERS IN 'SY' PDS       *
//**************************************************************
//*
//*    *--------------------------------------------------*
//*    *              I M P O R T A N T                   *
//*    *                 -----------                      *
//*    *      BEFORE EXECUTING THIS JOB, REPLACE ALL      *
//*    *          ':/' BY './' VIA THE EDITOR.            *
//*    *--------------------------------------------------*
//*
//*
//ET010   EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=$OUT
//SYSIN    DD *
  DELETE ($INDEXP..$ROOT.$ROOT.SY)
//*
//ET020   EXEC PGM=IEFBR14
//DDA      DD DSN=$INDEXP..$ROOT.$ROOT.SY,DISP=(,CATLG,DELETE),
//            UNIT=$UNITP,
//            VOL=SER=$VOLP,
//            DCB=(RECFM=FB,LRECL=80,BLKSIZE=6080),
//            SPACE=(TRK,(03,02,04))
//*
//ET030   EXEC PGM=IEBUPDTE,PARM=NEW
//SYSPRINT DD SYSOUT=$OUT
//SYSUT1   DD DSN=$INDEXP..$ROOT.$ROOT.SY,DISP=SHR
//SYSUT2   DD DSN=$INDEXP..$ROOT.$ROOT.SY,DISP=SHR
//SYSIN    DD DATA,DLM='$%'
:/        ADD NAME=DF$ROOT.$FILE.TS
 DELETE ($INDEX..$ROOT.$FILE.TS) CLUSTER
 DEFINE CLUSTER ( NAME ($INDEX..$ROOT.$FILE.TS)     -
                  SHR (2 3)       KEYS (14 1)       -
                  REC  (100)        -
                  VOL ($VOLU)       -
                  RECSZ (080,080) RUS )             -
        INDEX   ( NAME ($INDEX..$ROOT.$FILE.TS.I)   -
                  CISZ (0512) )                     -
        DATA    ( NAME ($INDEX..$ROOT.$FILE.TS.D)   -
                  FSPC (10,5) SPEED                 -
                  CISZ (4096) )  /*: CATALOG ($CATU) */
:/        ADD NAME=DF$ROOT.$FILE.TY
 DELETE ($INDEX..$ROOT.$FILE.TY) CLUSTER
 DEFINE CLUSTER ( NAME ($INDEX..$ROOT.$FILE.TY)     -
                  SHR (2,3)       KEYS (8,1)        -
                  REC  (150)        -
                  VOL ($VOLU)       -
                  RECSZ (080,080) RUS )             -
        INDEX   ( NAME ($INDEX..$ROOT.$FILE.TY.I)   -
                  CISZ (4096) )                     -
        DATA    ( NAME ($INDEX..$ROOT.$FILE.TY.D)   -
                  FSPC (10,5)                       -
                  CISZ (4096) )  /*: CATALOG ($CATU) */
:/        ADD NAME=DF$ROOT.$FILE.UP
 DELETE ($INDEX..$ROOT.$FILE.UP) CLUSTER
 DEFINE CLUSTER ( NAME ($INDEX..$ROOT.$FILE.UP)     -
                  SHR (2,3)       KEYS (43,0)       -
                  REC  (10000)      -
                  VOL ($VOLU)       -
                  RECSZ (112,112) RUS )             -
        INDEX   ( NAME ($INDEX..$ROOT.$FILE.UP.I)   -
                  CISZ (4096) )                     -
        DATA    ( NAME ($INDEX..$ROOT.$FILE.UP.D)   -
                  FSPC (10,5)                       -
                  CISZ (4096) )  /*: CATALOG ($CATU) */
:/        ADD NAME=DF$ROOT.$FILE.UQ
 DELETE ($INDEX..$ROOT.$FILE.UQ) CLUSTER
 DEFINE CLUSTER ( NAME ($INDEX..$ROOT.$FILE.UQ)     -
                  SHR (2,3)       NUMBERED          -
                  REC  (10000)      -
                  VOL ($VOLU)       -
                  RECSZ (170,170) RUS )             -
        DATA    ( NAME ($INDEX..$ROOT.$FILE.UQ.D)   -
                  CISZ (4096) )  /*: CATALOG ($CATU) */
:/        ADD NAME=DF$ROOT.$FILE.UU
 DELETE ($INDEX..$ROOT.$FILE.UU) CLUSTER
 DEFINE CLUSTER ( NAME ($INDEX..$ROOT.$FILE.UU)     -
                  SHR (2,3)       NUMBERED          -
                  REC  (10000)      -
                  VOL ($VOLU)       -
                  RECSZ (170,170) RUS )             -
        DATA    ( NAME ($INDEX..$ROOT.$FILE.UU.D)   -
                  CISZ (4096) )  /*: CATALOG ($CATU) */
:/ ADD NAME=DFSVSAM8
4096,8
:/ ADD NAME=DL$ROOT.$FILE.UU
 DELETE ($INDEX..$ROOT.$FILE.UU) CLUSTER
:/ ADD NAME=MAXKEY
99999999999999999999999999999999999999999999999999999999999999999
:/ ADD NAME=REPRO999
 REPRO INFILE(MAXKEY) OUTFILE(SYSPAF)
:/ ADD NAME=VERIFUP
 VERIFY FILE (DDUP)
:/ ADD NAME=VERIFUQ
 VERIFY FILE (DDUQ)
:/ ADD NAME=VERIFAE
 VERIFY FILE (DDAE)
:/ ADD NAME=REPROTS
 REPRO INFILE (INTS) OUTFILE (OUTTS)
:/ ADD NAME=BL$ROOT.$FILE.QU
   DEFINE GENERATIONDATAGROUP -
          (NAME ($INDEXQ..$ROOT.$FILE.QU) LIMIT (3) SCR)
:/ ADD NAME=ISPLOGON
//*
//* THE FOLLOWING THREE FILES MUST BE DECLARED
//* INTO THE USER TSO LOGON PROCEDURE.
//*
//IKJ     EXEC PGM=IKJEFT01, ...
  .
  .
//PAC7TY   DD DSN=$INDEX..$ROOT.$FILE.TY,DISP=SHR
//PAC7UP   DD DSN=$INDEX..$ROOT.$FILE.UP,DISP=SHR
//PAC7UQ   DD DSN=$INDEX..$ROOT.$FILE.UQ,DISP=SHR
  .
  .
//
:/ ADD NAME=ISPF
//*
//* EXITS TRACE FILE ALLOCATION MUST BE DECLARED
//* INTO THE USER ISPF CONNECTION PROCEDURE.
//*
CONTROL MAIN NOMSG
SET &PTRACE = &STR('$INDEXP..&SYSUID..PTRACE')
FREE  FI(PTRACE)
ALLOC FI(PTRACE) SHR REU DA(&PTRACE)
```

```
IF &LASTCC > 0 THEN DO
   FREE ATTRLIST(DCBTRACE)
   ATTR DCBTRACE  RECFM(F B) LRECL(120) BLKSIZE(12000) DSORG(PS)
   ALLOC FI(PTRACE) DA(&PTRACE) SPACE(20,10) TRACKS USING(DCBTRACE)
   FREE  FI(PTRACE)
   ALLOC FI(PTRACE) SHR REU DA(&PTRACE)
END
:/ ADD NAME=CIUU$MSGSX
CIUU$MSGSX.1E     'INVALID TYPE (INFOPAC)  ' .ALARM = YES .HELP = *
'''INFOPAC'' TYPE ELEMENTS ARE NOT ALLOWED FOR UPDATING
CIUU$MSGSX.2E     'INVALID ACTION (PACBASE)' .ALARM = YES .HELP = *
'ACTIONS ''ADD & UPDATE'' ARE NOT ALLOWED FOR PACBASE OUT OF PROCESSOR
CIUU$MSGSX.3E     'INVALID ACTION (PACBASE)' .ALARM = YES .HELP = *
'ONLY ACTIONS ''ADD & UPDATE'' ARE ALLOWED FOR PACBASE THROUGH PROCESSO
:/ ADD NAME=PRCSGENI
//*
//*-------------------------------------------------------------------*
//*  G E N E R A T E    P R O C E S S O R           (INFOPAC TYPE)  *
//*-------------------------------------------------------------------*
//*
//GEN00   EXEC PGM=BC1PDSIN,MAXRC=0,
//        EXECIF=(&C1COMMENT(36,5),NE,'*RND*')
//GEN02A   DD DSN=&&SYSOUT1,DISP=(,PASS,DELETE),UNIT=$UWK,
//            DCB=(RECFM=FBA,LRECL=121,BLKSIZE=6171),
//            SPACE=(TRK,(5,5),RLSE)
//GEN02B   DD DSN=&&SYSDMP1,DISP=(,PASS,DELETE),UNIT=$UWK,
//            DCB=(RECFM=FBA,LRECL=121,BLKSIZE=6171),
//            SPACE=(TRK,(5,5),RLSE)
//GEN03A   DD DSN=&&SYSOUT2,DISP=(,PASS,DELETE),UNIT=$UWK,
//            DCB=(RECFM=FBA,LRECL=121,BLKSIZE=6171),
//            SPACE=(TRK,(5,5),RLSE)
//GEN04A   DD DSN=&&SYSOUT3,DISP=(,PASS,DELETE),UNIT=$UWK,
//            DCB=(RECFM=FBA,LRECL=121,BLKSIZE=6171),
//            SPACE=(TRK,(5,5),RLSE)
//GEN04B   DD DSN=&&SYSDMP3,DISP=(,PASS,DELETE),UNIT=$UWK,
//            DCB=(RECFM=FBA,LRECL=121,BLKSIZE=6171),
//            SPACE=(TRK,(5,5),RLSE)
//*
//GEN01   EXEC PGM=PNTRAN,
//        EXECIF=(&C1COMMENT(36,5),NE,'*RND*'),
//             PARM=(&C1ACTION(1,8,_),&C1ENVMNT(1,8,_),
//             &C1SYSTEM(1,8,_)&C1SUBSYS(1,8,_)&C1ELTYPE(1,8,_),
//             &C1ELEMENT(1,10,_)&C1STGNUM(1,1,_)&C1CCID(1,12,_)I)
//STEPLIB  DD DSN=$MODB,DISP=SHR
//PAC7PU   DD DSN=$INDEX..$ROOT.$FILE.UP,DISP=SHR
//PAC7TR   DD DSN=&&PAC7TR,DISP=(,PASS),UNIT=$UWK,
//            DCB=(RECFM=FB,LRECL=58,BLKSIZE=58),
//            SPACE=(TRK,(01,01),RLSE)
//*
//GEN02   EXEC PGM=PNPR10,MAXRC=0,
//        EXECIF=(&C1COMMENT(36,5),NE,'*RND*'),
//             PARM=(&C1ACTION(1,8,_),&C1ENVMNT(1,8,_),
//             &C1SYSTEM(1,8,_)&C1SUBSYS(1,8,_)&C1ELTYPE(1,8,_),
//             &C1ELEMENT(1,10,_)&C1STGNUM(1,1,_)&C1CCID(1,12,_))
//STEPLIB  DD DSN=$MODB,DISP=SHR
//SYSIN    DD DSN=&&PAC7TR,DISP=(OLD,DELETE)
//PAC7TZ   DD DSN=$INDEX..$ROOT.$FILE.TY,DISP=SHR
//PAC7BS   DD DUMMY,DCB=(RECFM=FB,LRECL=80,BLKSIZE=6080)
//PAC7CW   DD DSN=&&CONWIN,DISP=(,PASS),UNIT=$UWK,
//            DCB=(RECFM=FB,LRECL=80,BLKSIZE=6080),
//            SPACE=(TRK,(01,01),RLSE)
//PAC7EV   DD DSN=&&PAC7EV,DISP=(,PASS),UNIT=$UWK,
//            DCB=(RECFM=FB,LRECL=126,BLKSIZE=12600),
//            SPACE=(TRK,(01,01),RLSE)
```

```
//SYSOUT   DD DSN=&&SYSOUT1,DISP=(MOD,PASS)
//SYSUDUMP DD DSN=&&SYSDMP1,DISP=(MOD,PASS)
//*
//*---------------------------------------------------*
//*  PRINT 'INFOPAC' MEMBER OF THE ORIGINAL STAGE     *
//*  INTO SEQUENTIAL FILE 'PAC7IP'                    *
//*---------------------------------------------------*
//*
//GEN03   EXEC PGM=CONWRITE,MAXRC=0,
//        EXECIF=(&C1COMMENT(36,5),NE,'*RND*')
//CONWIN   DD DSN=&&CONWIN,DISP=(OLD,DELETE)
//PAC7IP   DD DSN=&&PAC7IP,DISP=(,PASS),UNIT=$UWK,
//            DCB=(RECFM=FB,LRECL=80,BLKSIZE=6080),
//            SPACE=(TRK,(02,01),RLSE)
//SYSOUT   DD DSN=&&SYSOUT2,DISP=(MOD,PASS)
//*
//*---------------------------------------------------*
//*  COPY SEQUENTIAL 'PAC7IP' IN THE VSAM 'PAC7UP'    *
//*  DEFINED IN TSO WITH A NEW DDNAME FOR EXIT3       *
//*---------------------------------------------------*
//*
//GEN04   EXEC PGM=PNPR11,COND=(00,NE,GEN02),
//        EXECIF=(&C1COMMENT(36,5),NE,'*RND*')
//STEPLIB  DD DSN=$MODB,DISP=SHR
//PAC7EV   DD DSN=&&PAC7EV,DISP=(OLD,PASS)
//PAC7IP   DD DSN=&&PAC7IP,DISP=(OLD,PASS)
//PAC7PU   DD DSN=$INDEX..$ROOT.$FILE.UP,DISP=SHR
//SYSOUT   DD DSN=&&SYSOUT3,DISP=(MOD,PASS)
//SYSUDUMP DD DSN=&&SYSDMP3,DISP=(MOD,PASS)
//*
//*---------------------------------------------------*
//*           PRINT SYSOUTS, SYSUDUMP, ...            *
//*---------------------------------------------------*
//*
//GEN05   EXEC PGM=CONLIST,PARM=PRINT,COND=EVEN,MAXRC=0,
//        EXECIF=((&C1COMMENT(36,5),NE,'*CGI*'),
//               (&C1COMMENT(36,5),NE,'*RND*'))
//C1BANNER DD DSN=&&BANNER,DISP=(,PASS,DELETE),UNIT=$UWK,
//            SPACE=(TRK,(1,1),RLSE)
//C1PRINT  DD SYSOUT=*,DCB=(RECFM=FBA,LRECL=121,BLKSIZE=6171,DSORG=PS)
//LIST01   DD DSN=&&SYSOUT1,DISP=(OLD,DELETE)
//LIST02   DD DSN=&&SYSDMP1,DISP=(OLD,DELETE)
//LIST03   DD DSN=&&SYSOUT2,DISP=(OLD,DELETE)
//LIST04   DD DSN=&&SYSOUT3,DISP=(OLD,DELETE)
//LIST05   DD DSN=&&SYSDMP3,DISP=(OLD,DELETE)
//*
:/ ADD NAME=PRCSGEPP
//*
//*-------------------------------------------------------------------*
//*  G E N E R A T E    P R O C E S S O R             (VA PAC  TYPE)  *
//*-------------------------------------------------------------------*
//*
//*
//TRS01   EXEC PGM=BC1PDSIN,MAXRC=0,
//        EXECIF=((&C1ACTION(1,3),NE,'ADD'),
//               (&C1ACTION(1,4),NE,'MOVE'),
//               (&C1ACTION(1,6),NE,'UPDATE'))
//TRS03A   DD DSN=&&SYSOUT1,DISP=(,PASS,DELETE),UNIT=$UWK,
//            DCB=(RECFM=FBA,LRECL=121,BLKSIZE=6171),
//            SPACE=(TRK,(5,5),RLSE)
//TRS03B   DD DSN=&&SYSDMP1,DISP=(,PASS,DELETE),UNIT=$UWK,
//            DCB=(RECFM=FBA,LRECL=121,BLKSIZE=6171),
//            SPACE=(TRK,(5,5),RLSE)
//TRS04A   DD DSN=&&SYSOUT2,DISP=(,PASS,DELETE),UNIT=$UWK,
//            DCB=(RECFM=FBA,LRECL=121,BLKSIZE=6171),
//            SPACE=(TRK,(5,5),RLSE)
//TRS05A   DD DSN=&&SYSOUT3,DISP=(,PASS,DELETE),UNIT=$UWK,
//            DCB=(RECFM=FBA,LRECL=121,BLKSIZE=6171),
//            SPACE=(TRK,(5,5),RLSE)
//TRS05B   DD DSN=&&SYSDMP3,DISP=(,PASS,DELETE),UNIT=$UWK,
//            DCB=(RECFM=FBA,LRECL=121,BLKSIZE=6171),
//            SPACE=(TRK,(5,5),RLSE)
//*
//TRS02   EXEC PGM=PNTRAN,
//        EXECIF=((&C1ACTION(1,3),NE,'ADD'),
//               (&C1ACTION(1,4),NE,'MOVE'),
//               (&C1ACTION(1,6),NE,'UPDATE')),
//             PARM=(&C1ACTION(1,8,_),&C1ENVMNT(1,8,_),
//             &C1SYSTEM(1,8,_)&C1SUBSYS(1,8,_)&C1ELTYPE(1,8,_),
//             &C1ELEMENT(1,10,_)&C1STGNUM(1,1,_)&C1CCID(1,12,_)G)
//STEPLIB  DD DSN=$MODB,DISP=SHR
//PAC7PU   DD DSN=$INDEX..$ROOT.$FILE.UP,DISP=SHR
//PAC7TR   DD DSN=&&PAC7TR,DISP=(,PASS),UNIT=$UWK,
//            DCB=(RECFM=FB,LRECL=58,BLKSIZE=58),
//            SPACE=(TRK,(01,01),RLSE)
//*
//TRS03   EXEC PGM=PNPR10,
//        EXECIF=((&C1ACTION(1,3),NE,'ADD'),
//               (&C1ACTION(1,4),NE,'MOVE'),
//               (&C1ACTION(1,6),NE,'UPDATE')),
//             PARM=(&C1ACTION(1,8,_),&C1ENVMNT(1,8,_),
//             &C1SYSTEM(1,8,_)&C1SUBSYS(1,8,_)&C1ELTYPE(1,8,_),
//             &C1ELEMENT(1,10,_)&C1STGNUM(1,1,_)&C1CCID(1,12,_))
//STEPLIB  DD DSN=$MODB,DISP=SHR
//SYSIN    DD DSN=&&PAC7TR,DISP=(OLD,DELETE)
//PAC7TZ   DD DSN=$INDEX..$ROOT.$FILE.TY,DISP=SHR
//PAC7BS   DD DSN=&&BSTIPT01,DISP=(,PASS),UNIT=$UWK,
//            DCB=(RECFM=FB,LRECL=80,BLKSIZE=6080),
//            SPACE=(TRK,(01,01),RLSE)
//PAC7CW   DD DSN=&&CONWIN,DISP=(,PASS),UNIT=$UWK,
//            DCB=(RECFM=FB,LRECL=80,BLKSIZE=6080),
//            SPACE=(TRK,(01,01),RLSE)
//PAC7EV   DD DSN=&&PAC7EV,DISP=(,PASS),UNIT=$UWK,
//            DCB=(RECFM=FB,LRECL=126,BLKSIZE=12600),
//            SPACE=(TRK,(01,01),RLSE)
//SYSOUT   DD DSN=&&SYSOUT1,DISP=(MOD,PASS)
//SYSUDUMP DD DSN=&&SYSDMP1,DISP=(MOD,PASS)
//*
//*-------------------------------------------------*
//*  PRINT 'INFOPAC' MEMBER OF THE ORIGINAL STAGE     *
//*  INTO SEQUENTIAL FILE 'PAC7IP'                    *
//*-------------------------------------------------*
//*
//TRS04   EXEC PGM=CONWRITE,MAXRC=0,
//        EXECIF=(&C1ACTION(1,8),EQ,'GENERATE')
//CONWIN   DD DSN=&&CONWIN,DISP=(OLD,DELETE)
//PAC7IP   DD DSN=&&PAC7IP,DISP=(,PASS),UNIT=$UWK,
//            DCB=(RECFM=FB,LRECL=80,BLKSIZE=6080),
//            SPACE=(TRK,(02,01),RLSE)
//SYSOUT   DD DSN=&&SYSOUT2,DISP=(MOD,PASS)
//*
//*-------------------------------------------------*
//*  COPY SEQUENTIAL 'PAC7IP' IN THE VSAM 'PAC7UP'    *
//*  DEFINED IN TSO WITH A NEW DDNAME FOR  EXIT3      *
//*-------------------------------------------------*
//*
//TRS05   EXEC PGM=PNPR11,COND=(00,NE,TRS03),
//        EXECIF=(&C1ACTION(1,8),EQ,'GENERATE')
```

```
//STEPLIB  DD DSN=$MODB,DISP=SHR
//PAC7EV   DD DSN=&&PAC7EV,DISP=(OLD,PASS)
//PAC7IP   DD DSN=&&PAC7IP,DISP=(OLD,PASS)
//PAC7PU   DD DSN=$INDEX..$ROOT.$FILE.UP,DISP=SHR
//SYSOUT   DD DSN=&&SYSOUT3,DISP=(MOD,PASS)
//SYSUDUMP DD DSN=&&SYSDMP3,DISP=(MOD,PASS)
//*
//*--------------------------------------------------*
//*          TRANSFER 'INFOPAC' ELEMENT                *
//*--------------------------------------------------*
//*
//TRS06   EXEC PGM=C1BM3000,PARM=(PAC7BS,CXMSGSX),
//        EXECIF=(&C1ACTION(1,8),EQ,'TRANSFER')
//PAC7BS   DD DSN=&&BSTIPT01,DISP=(OLD,PASS)
//CXMSGSX  DD SYSOUT=*
//*
//*--------------------------------------------------*
//*            PRINT SYSOUTS, SYSUDUMP, ...            *
//*--------------------------------------------------*
//*
//TRS07   EXEC PGM=CONLIST,PARM=PRINT,COND=EVEN,MAXRC=0,
//        EXECIF=((&C1ACTION(1,3),NE,'ADD'),
//               (&C1ACTION(1,4),NE,'MOVE'),
//               (&C1ACTION(1,6),NE,'UPDATE'))
//C1BANNER DD DSN=&&BANNER,DISP=(,PASS,DELETE),UNIT=$UWK,
//            SPACE=(TRK,(1,1),RLSE)
//C1PRINT  DD SYSOUT=*,DCB=(RECFM=FBA,LRECL=121,BLKSIZE=6171,DSORG=PS)
//LIST01   DD DSN=&&SYSOUT1,DISP=(OLD,DELETE)
//LIST02   DD DSN=&&SYSDMP1,DISP=(OLD,DELETE)
//LIST03   DD DSN=&&SYSOUT2,DISP=(OLD,DELETE)
//LIST04   DD DSN=&&SYSOUT3,DISP=(OLD,DELETE)
//LIST05   DD DSN=&&SYSDMP3,DISP=(OLD,DELETE)
//*
//
:/ ADD NAME=PRCSGENP
//*
//*----------------------------------------------------------------*
//*  G E N E R A T E  (LOAD)  P R O C E S S O R      (VA PAC  TYPE)  *
//*----------------------------------------------------------------*
//*
//*
//CNWRITE EXEC PGM=CONWRITE
//ELMSRC   DD  DSN=&&ELMSRC,UNIT=SYSDA,DISP=(,PASS),
//             DCB=(RECFM=FB,LRECL=80,BLKSIZE=400),
//             SPACE=(TRK,(15,15),RLSE)
//CONWIN   DD *
  WRITE ELEMENT          &C1ELEMENT
        FROM ENVIRONMENT &C1ENVMNT
             SYSTEM      &C1SYSTEM
             SUBSYSTEM   &C1SUBSYS
             TYPE        &C1ELTYPE
             STAGE       &C1STGID
        TO   DDN         ELMSRC.
//*
//COB     EXEC PGM=IKFCBL00,MAXRC=04,
//             PARM='SIZE=500K,BUF=90K,LANGLVL(1),CLI,NONAME,LIB,OPT'
//SYSUT1   DD  UNIT=$UWK,SPACE=(460,(950,200)),DISP=(,DELETE)
//SYSUT2   DD  UNIT=$UWK,SPACE=(460,(950,200)),DISP=(,DELETE)
//SYSUT3   DD  UNIT=$UWK,SPACE=(460,(950,200)),DISP=(,DELETE)
//SYSUT4   DD  UNIT=$UWK,SPACE=(460,(950,200)),DISP=(,DELETE)
//SYSIN    DD  DSN=&&ELMSRC,DISP=(OLD,DELETE)
//SYSLIN   DD  DSN=&&LOADSET,UNIT=$UWK,DISP=(MOD,PASS),
//             SPACE=(400,(500,200)),
//             DCB=(RECFM=FB,LRECL=80,BLKSIZE=400)
```

```
//SYSPRINT DD  SYSOUT=$OUT
//*
//LKED    EXEC PGM=IEWL,PARM=(LIST,LET,XREF),MAXRC=04
//**      COND=(4,LT,COB)
//DDSET    DD UNIT=$UWK,SPACE=(CYL,1)
//SYSLIB   DD DSN=SYS1.COB2LIB,DISP=SHR
//         DD DSN=???,DISP=SHR                   <-- LOAD-MODULES
//SYSLIN   DD DSN=&&LOADSET,DISP=(OLD,DELETE)
//         DD *,DCB=BLKSIZE=80
 NAME &C1ELEMENT(R)
//SYSLMOD  DD DSN=???,DISP=SHR,FOOTPRNT=CREATE   <-- LOAD-MODULES
//SYSUT1   DD UNIT=$UWK,SPACE=(1024,(50,20))
//SYSPRINT DD SYSOUT=$OUT
//*
//TRS00   EXEC PGM=BC1PDSIN,MAXRC=0,
//        EXECIF=((&C1ACTION(1,3),NE,'ADD'),
//               (&C1ACTION(1,4),NE,'MOVE'),
//               (&C1ACTION(1,6),NE,'UPDATE'))
//TRS02A   DD DSN=&&SYSOUT1,DISP=(,PASS,DELETE),UNIT=$UWK,
//            DCB=(RECFM=FBA,LRECL=121,BLKSIZE=6171),
//            SPACE=(TRK,(05,05),RLSE)
//TRS02B   DD DSN=&&SYSDMP1,DISP=(,PASS,DELETE),UNIT=$UWK,
//            DCB=(RECFM=FBA,LRECL=121,BLKSIZE=6171),
//            SPACE=(TRK,(05,05),RLSE)
//TRS03B   DD DSN=&&SYSOUT2,DISP=(,PASS,DELETE),UNIT=$UWK,
//            DCB=(RECFM=FBA,LRECL=121,BLKSIZE=6171),
//            SPACE=(TRK,(05,05),RLSE)
//TRS04A   DD DSN=&&SYSOUT3,DISP=(,PASS,DELETE),UNIT=$UWK,
//            DCB=(RECFM=FBA,LRECL=121,BLKSIZE=6171),
//            SPACE=(TRK,(05,05),RLSE)
//TRS04B   DD DSN=&&SYSDMP3,DISP=(,PASS,DELETE),UNIT=$UWK,
//            DCB=(RECFM=FBA,LRECL=121,BLKSIZE=6171),
//            SPACE=(TRK,(05,05),RLSE)
//*
//TRS01   EXEC PGM=PNTRAN,
//        EXECIF=((&C1ACTION(1,3),NE,'ADD'),
//               (&C1ACTION(1,4),NE,'MOVE'),
//               (&C1ACTION(1,6),NE,'UPDATE')),
//             PARM=(&C1ACTION(1,8,_),&C1ENVMNT(1,8,_),
//             &C1SYSTEM(1,8,_)&C1SUBSYS(1,8,_)&C1ELTYPE(1,8,_),
//             &C1ELEMENT(1,10,_)&C1STGNUM(1,1,_)&C1CCID(1,12,_)G)
//STEPLIB  DD DSN=$MODB,DISP=SHR
//PAC7PU   DD DSN=$INDEX..$ROOT.$FILE.UP,DISP=SHR
//PAC7TR   DD DSN=&&PAC7TR,DISP=(,PASS),UNIT=$UWK,
//            DCB=(RECFM=FB,LRECL=58,BLKSIZE=58),
//            SPACE=(TRK,(01,01),RLSE)
//*
//TRS02   EXEC PGM=PNPR10,
//        EXECIF=((&C1ACTION(1,3),NE,'ADD'),
//               (&C1ACTION(1,4),NE,'MOVE'),
//               (&C1ACTION(1,6),NE,'UPDATE')),
//             PARM=(&C1ACTION(1,8,_),&C1ENVMNT(1,8,_),
//             &C1SYSTEM(1,8,_)&C1SUBSYS(1,8,_)&C1ELTYPE(1,8,_),
//             &C1ELEMENT(1,10,_)&C1STGNUM(1,1,_)&C1CCID(1,12,_))
//STEPLIB  DD DSN=$MODB,DISP=SHR
//SYSIN    DD DSN=&&PAC7TR,DISP=(OLD,DELETE)
//PAC7TZ   DD DSN=$INDEX..$ROOT.$FILE.TY,DISP=SHR
//PAC7BS   DD DSN=&&BSTIPT01,DISP=(,PASS),UNIT=$UWK,
//            DCB=(RECFM=FB,LRECL=80,BLKSIZE=6080),
//            SPACE=(TRK,(01,01),RLSE)
//PAC7CW   DD DSN=&&CONWIN,DISP=(,PASS),UNIT=$UWK,
//            DCB=(RECFM=FB,LRECL=80,BLKSIZE=6080),
//            SPACE=(TRK,(01,01),RLSE)
//PAC7EV   DD DSN=&&PAC7EV,DISP=(,PASS),UNIT=$UWK,
//             DCB=(RECFM=FB,LRECL=126,BLKSIZE=12600),
//             SPACE=(TRK,(01,01),RLSE)
//SYSOUT   DD DSN=&&SYSOUT1,DISP=(MOD,PASS)
//SYSUDUMP DD DSN=&&SYSDMP1,DISP=(MOD,PASS)
//*--------------------------------------------------*
//*  PRINT 'INFOPAC MEMBER OF THE ORIGINAL STAGE      *
//*  INTO SEQUENTIAL FILE 'PAC7IP'                    *
//*--------------------------------------------------*
//*
//TRS03   EXEC PGM=CONWRITE,MAXRC=0,
//        EXECIF=(&C1ACTION(1,8),EQ,'GENERATE')
//CONWIN   DD DSN=&&CONWIN,DISP=(OLD,DELETE)
//PAC7IP   DD DSN=&&PAC7IP,DISP=(,PASS),UNIT=$UWK,
//             DCB=(RECFM=FB,LRECL=80,BLKSIZE=6080),
//             SPACE=(TRK,(02,01),RLSE)
//*
//*--------------------------------------------------*
//*  COPY SEQUENTIAL 'PAC7IP' IN THE VSAM 'PAC7UP'    *
//*  DEFINED IN TSO WITH A NEW DDNAME FOR EXIT3       *
//*--------------------------------------------------*
//*
//TRS04   EXEC PGM=PNPR11,COND=(00,NE,TRS03),
//        EXECIF=(&C1ACTION(1,8),EQ,'GENERATE')
//STEPLIB  DD DSN=$MODB,DISP=SHR
//PAC7EV   DD DSN=&&PAC7EV,DISP=(OLD,PASS)
//PAC7IP   DD DSN=&&PAC7IP,DISP=(OLD,PASS)
//PAC7PU   DD DSN=$INDEX..$ROOT.$FILE.UP,DISP=SHR
//SYSOUT   DD DSN=&&SYSOUT3,DISP=(MOD,PASS)
//SYSUDUMP DD DSN=&&SYSDMP3,DISP=(MOD,PASS)
//*
//*--------------------------------------------------*
//*           TRANSFER 'INFOPAC' ELEMENT              *
//*--------------------------------------------------*
//*
//TRS05   EXEC PGM=C1BM3000,PARM=(PAC7BS,CXMSGSX),
//        EXECIF=(&C1ACTION(1,8),EQ,'TRANSFER'),
//PAC7BS   DD DSN=&&BSTIPT01,DISP=(OLD,PASS)
//CXMSGSX  DD SYSOUT=$OUT
//*
//*--------------------------------------------------*
//*           PRINT SYSOUTS, SYSUDUMP, ...            *
//*--------------------------------------------------*
//*
//TRS06   EXEC PGM=CONLIST,PARM=PRINT,COND=EVEN,MAXRC=0,
//        EXECIF=((&C1ACTION(1,3),NE,'ADD'),
//               (&C1ACTION(1,4),NE,'MOVE'),
//               (&C1ACTION(1,6),NE,'UPDATE'))
//C1BANNER DD DSN=&&BANNER,DISP=(,PASS,DELETE),UNIT=$UWK,
//             SPACE=(TRK,(01,01),RLSE)
//C1PRINT  DD SYSOUT=$OUT,
//             DCB=(RECFM=FBA,LRECL=121,BLKSIZE=6171,DSORG=PS)
//LIST01   DD DSN=&&SYSOUT1,DISP=(OLD,DELETE)
//LIST02   DD DSN=&&SYSDMP1,DISP=(OLD,DELETE)
//LIST03   DD DSN=&&SYSOUT2,DISP=(OLD,DELETE)
//LIST04   DD DSN=&&SYSOUT3,DISP=(OLD,DELETE)
//LIST05   DD DSN=&&SYSDMP3,DISP=(OLD,DELETE)
//*
//
:/ ADD NAME=PRCSDELP
//*
//*-----------------------------------------------------------------*
//*  D E L E T E    P R O C E S S O R                 (VA PAC  TYPE)  *
//*-----------------------------------------------------------------*
//*
```

```
//DEL00   EXEC PGM=BC1PDSIN,MAXRC=0
//DEL01A   DD DSN=&&SYSOUT1,DISP=(,PASS,DELETE),UNIT=SYSDA,
//            DCB=(RECFM=FBA,LRECL=121,BLKSIZE=6171),
//            SPACE=(TRK,(5,5),RLSE)
//DEL01B   DD DSN=&&SYSDMP1,DISP=(,PASS,DELETE),UNIT=SYSDA,
//            DCB=(RECFM=FBA,LRECL=121,BLKSIZE=6171),
//            SPACE=(TRK,(5,5),RLSE)
//DEL03A   DD DSN=&&SYSOUT2,DISP=(,PASS,DELETE),UNIT=SYSDA,
//            DCB=(RECFM=FBA,LRECL=121,BLKSIZE=6171),
//            SPACE=(TRK,(5,5),RLSE)
//DEL03B   DD DSN=&&SYSDMP2,DISP=(,PASS,DELETE),UNIT=SYSDA,
//            DCB=(RECFM=FBA,LRECL=121,BLKSIZE=6171),
//            SPACE=(TRK,(5,5),RLSE)
//*
//DEL01   EXEC PGM=PNPR10,
//        EXECIF=(&C1COMMENT(36,5),NE,'*IBM*'),
//             PARM=(DELETE__,&C1ENVMNT(1,8,_),
//             &C1SYSTEM(1,8,_)&C1SUBSYS(1,8,_)&C1ELTYPE(1,8,_),
//             &C1ELEMENT(1,10,_)&C1STGNUM(1,1,_)&C1CCID(1,12,_))
//STEPLIB  DD DSN=$MODB,DISP=SHR
//SYSIN    DD DUMMY
//PAC7TZ   DD DSN=$INDEX..$ROOT.$FILE.TY,DISP=SHR
//PAC7BS   DD DSN=&&BSTIPT01,DISP=(,PASS),UNIT=$UWK,
//            DCB=(RECFM=FB,LRECL=80,BLKSIZE=6080),
//            SPACE=(TRK,(01,01),RLSE)
//PAC7CW   DD DSN=&&CONWIN,DISP=(,PASS),UNIT=$UWK,
//            DCB=(RECFM=FB,LRECL=80,BLKSIZE=6080),
//            SPACE=(TRK,(01,01),RLSE)
//PAC7EV   DD DSN=&&PAC7EV,DISP=(,PASS),UNIT=$UWK,
//            DCB=(RECFM=FB,LRECL=126,BLKSIZE=12600),
//            SPACE=(TRK,(01,01),RLSE)
//SYSOUT   DD DSN=&&SYSOUT1,DISP=(MOD,PASS)
//SYSUDUMP DD DSN=&&SYSDMP1,DISP=(MOD,PASS)
//*
//*---------------------------------------------------*
//*  PRINT 'INFOPAC' MEMBER INTO SEQUENTIAL 'PAC7IP'  *
//*---------------------------------------------------*
//*
//DEL02   EXEC PGM=CONWRITE,MAXRC=12,
//        EXECIF=(&C1COMMENT(36,5),NE,'*IBM*')
//CONWIN   DD DSN=&&CONWIN,DISP=(OLD,DELETE)
//PAC7IP   DD DSN=&&PAC7IP,DISP=(,PASS),UNIT=$UWK,
//            DCB=(RECFM=FB,LRECL=80,BLKSIZE=6080),
//            SPACE=(TRK,(02,01),RLSE)
//*
//*---------------------------------------------------*
//*  COPY SEQUENTIAL 'PAC7IP' IN VSAM 'PAC7UP'        *
//*  DEFINED UNDER TSO WITH ANOTHER DDNAME FOR EXIT3  *
//*---------------------------------------------------*
//*
//DEL03   EXEC PGM=PNPR11,COND=(00,NE,DEL02),
//        EXECIF=(&C1COMMENT(36,5),NE,'*IBM*')
//STEPLIB  DD DSN=$MODB,DISP=SHR
//PAC7EV   DD DSN=&&PAC7EV,DISP=(OLD,PASS)
//PAC7IP   DD DSN=&&PAC7IP,DISP=(OLD,PASS)
//PAC7PU   DD DSN=$INDEX..$ROOT.$FILE.UP,DISP=SHR
//SYSOUT   DD DSN=&&SYSOUT2,DISP=(MOD,PASS)
//SYSUDUMP DD DSN=&&SYSDMP2,DISP=(MOD,PASS)
//*
//*---------------------------------------------------*
//*            DELETE 'INFOPAC' ELEMENT               *
//*---------------------------------------------------*
//*
//DEL04   EXEC PGM=C1BM3000,PARM=(PAC7BS,CXMSGSX),COND=(00,NE,DEL02),
//        EXECIF=((&C1ACTION,NE,GENERATE),
//                (&C1COMMENT(36,5),NE,'*IBM*'))
//PAC7BS   DD DSN=&&BSTIPT01,DISP=(OLD,PASS)
//CXMSGSX  DD SYSOUT=$OUT
//*
//*---------------------------------------------------*
//*           PRINT SYSOUTS, SYSUDUMP, ...            *
//*---------------------------------------------------*
//*
//DEL05   EXEC PGM=CONLIST,PARM=PRINT,COND=EVEN,MAXRC=0,
//        EXECIF=(&C1COMMENT(36,5),NE,'*IBM*')
//C1BANNER DD DSN=&&BANNER,DISP=(,PASS,DELETE),UNIT=$UWK,
//            SPACE=(TRK,(01,01),RLSE)
//C1PRINT  DD SYSOUT=$OUT,
//            DCB=(RECFM=FBA,LRECL=121,BLKSIZE=6171,DSORG=PS)
//LIST01   DD DSN=&&SYSOUT1,DISP=(OLD,DELETE)
//LIST02   DD DSN=&&SYSDMP1,DISP=(OLD,DELETE)
//LIST03   DD DSN=&&SYSOUT2,DISP=(OLD,DELETE)
//LIST04   DD DSN=&&SYSDMP2,DISP=(OLD,DELETE)
//*
//
:/ ADD NAME=PRCSMOVP
//*
//*-------------------------------------------------------------------*
//*  M O V E    P R O C E S S O R                      (VA PAC  TYPE)  *
//*-------------------------------------------------------------------*
//*
//MOV00   EXEC PGM=BC1PDSIN,MAXRC=0
//MOV02A   DD DSN=&&SYSOUT1,DISP=(,PASS,DELETE),UNIT=SYSDA,
//            DCB=(RECFM=FBA,LRECL=121,BLKSIZE=6171),
//            SPACE=(TRK,(5,5),RLSE)
//MOV03A   DD DSN=&&SYSOUT2,DISP=(,PASS,DELETE),UNIT=SYSDA,
//            DCB=(RECFM=FBA,LRECL=121,BLKSIZE=6171),
//            SPACE=(TRK,(5,5),RLSE)
//MOV03B   DD DSN=&&SYSDMP2,DISP=(,PASS,DELETE),UNIT=SYSDA,
//            DCB=(RECFM=FBA,LRECL=121,BLKSIZE=6171),
//            SPACE=(TRK,(5,5),RLSE)
//MOV05A   DD DSN=&&SYSOUT3,DISP=(,PASS,DELETE),UNIT=SYSDA,
//            DCB=(RECFM=FBA,LRECL=121,BLKSIZE=6171),
//            SPACE=(TRK,(5,5),RLSE)
//MOV05B   DD DSN=&&SYSDMP3,DISP=(,PASS,DELETE),UNIT=SYSDA,
//            DCB=(RECFM=FBA,LRECL=121,BLKSIZE=6171),
//            SPACE=(TRK,(5,5),RLSE)
//*
//MOV01   EXEC PGM=PNTRAN,
//        EXECIF=(&C1ACTION,EQ,TRANSFER),
//             PARM=(&C1ACTION(1,8,_),&C1ENVMNT(1,8,_),
//             &C1SYSTEM(1,8,_)&C1SUBSYS(1,8,_)&C1ELTYPE(1,8,_),
//             &C1ELEMENT(1,10,_)&C1STGNUM(1,1,_)&C1CCID(1,12,_)M)
//STEPLIB  DD DSN=$MODB,DISP=SHR
//PAC7PU   DD DSN=$INDEX..$ROOT.$FILE.UP,DISP=SHR
//PAC7TR   DD DSN=&&&&C1ACTION,DISP=(,PASS),UNIT=$UWK,
//            DCB=(RECFM=FB,LRECL=58,BLKSIZE=58),
//            SPACE=(TRK,(01,01),RLSE)
//*
//MOV02   EXEC PGM=IEBGENER,EXECIF=(&C1ACTION,EQ,MOVE)
//SYSIN    DD *
  GENERATE MAXFLDS=1
  RECORD FIELD=(058,1,,1)
//SYSUT1   DD *
,&C1SENVMNT(1,8,_),&C1SSYSTEM(1,8,_)&C1SSUBSYS(1,8,_)
&C1SELTYPE(1,8,_),&C1SELEMENT(1,10,_)&C1SSTGNUM)
/*
//SYSUT2   DD DSN=&&&&C1ACTION,DISP=(,PASS),UNIT=$UWK,
//            DCB=(RECFM=FB,LRECL=58,BLKSIZE=58),
//            SPACE=(TRK,(01,01),RLSE)
//SYSOUT   DD DSN=&&SYSOUT1,DISP=(MOD,PASS)
//SYSPRINT DD SYSOUT=*
//*
//MOV03   EXEC PGM=PNPR10,MAXRC=0,
//             PARM=(&C1ACTION(1,8,_),&C1ENVMNT(1,8,_),
//             &C1SYSTEM(1,8,_)&C1SUBSYS(1,8,_)&C1ELTYPE(1,8,_),
//             &C1ELEMENT(1,10,_)&C1STGNUM(1,1,_)&C1CCID(1,12,_))
//STEPLIB  DD DSN=$MODB,DISP=SHR
//SYSIN    DD DSN=&&&&C1ACTION,DISP=(OLD,DELETE)
//PAC7TZ   DD DSN=$INDEX..$ROOT.$FILE.TY,DISP=SHR
//PAC7BS   DD DSN=&&BSTIPT01,DISP=(,PASS),UNIT=$UWK,
//            DCB=(RECFM=FB,LRECL=80,BLKSIZE=6080),
//            SPACE=(TRK,(01,01),RLSE)
//PAC7CW   DD DSN=&&CONWIN,DISP=(,PASS),UNIT=$UWK,
//            DCB=(RECFM=FB,LRECL=80,BLKSIZE=6080),
//            SPACE=(TRK,(01,01),RLSE)
//PAC7EV   DD DSN=&&PAC7EV,DISP=(,PASS),UNIT=$UWK,
//            DCB=(RECFM=FB,LRECL=126,BLKSIZE=12600),
//            SPACE=(TRK,(01,01),RLSE)
//SYSOUT   DD DSN=&&SYSOUT2,DISP=(MOD,PASS)
//SYSUDUMP DD DSN=&&SYSDMP2,DISP=(MOD,PASS)
//*
//*---------------------------------------------------*
//*  PRINT 'INFOPAC' MEMBER FROM ORIGIN STAGE INTO    *
//*  SEQUENTIAL 'PAC7IP'                              *
//*---------------------------------------------------*
//*
//MOV04   EXEC PGM=CONWRITE,MAXRC=0
//CONWIN   DD DSN=&&CONWIN,DISP=(OLD,DELETE)
//PAC7IP   DD DSN=&&PAC7IP,DISP=(,PASS),UNIT=$UWK,
//            DCB=(RECFM=FB,LRECL=80,BLKSIZE=6080),
//            SPACE=(TRK,(02,01),RLSE)
//*
//*---------------------------------------------------*
//*  COPY SEQUENTIAL 'PAC7IP' IN VSAM 'PAC7UP'        *
//*  DEFINED UNDER TSO WITH ANOTHER DDNAME FOR EXIT3  *
//*---------------------------------------------------*
//*
//MOV05   EXEC PGM=PNPR11,MAXRC=0,COND=(00,NE,MOV04)
//STEPLIB  DD DSN=$MODB,DISP=SHR
//PAC7EV   DD DSN=&&PAC7EV,DISP=(OLD,PASS)
//PAC7IP   DD DSN=&&PAC7IP,DISP=(OLD,PASS)
//PAC7PU   DD DSN=$INDEX..$ROOT.$FILE.UP,DISP=SHR
//SYSOUT   DD DSN=&&SYSOUT3,DISP=(MOD,PASS)
//SYSUDUMP DD DSN=&&SYSDMP3,DISP=(MOD,PASS)
//*
//*---------------------------------------------------*
//*             MOVE 'INFOPAC' ELEMENT                *
//*---------------------------------------------------*
//*
//MOV06   EXEC PGM=C1BM3000,PARM=(PAC7BS,CXMSGSX),MAXRC=8,
//             COND=(00,NE,MOV04)
//PAC7BS   DD DSN=&&BSTIPT01,DISP=(OLD,PASS)
//CXMSGSX  DD SYSOUT=$OUT
//*
//*---------------------------------------------------*
//*           PRINT SYSOUTS, SYSUDUMP, ...            *
//*---------------------------------------------------*
//*
//MOV07   EXEC PGM=CONLIST,PARM=PRINT,COND=EVEN
//C1BANNER DD DSN=&&BANNER,DISP=(,PASS,DELETE),UNIT=$UWK,
//            SPACE=(TRK,(01,01),RLSE)
```

```
//C1PRINT  DD SYSOUT=$OUT,
//            DCB=(RECFM=FBA,LRECL=121,BLKSIZE=6171,DSORG=PS)
//LIST01   DD DSN=&&SYSOUT1,DISP=(OLD,DELETE)
//LIST02   DD DSN=&&SYSOUT2,DISP=(OLD,DELETE)
//LIST03   DD DSN=&&SYSDMP2,DISP=(OLD,DELETE)
//LIST04   DD DSN=&&SYSOUT3,DISP=(OLD,DELETE)
//LIST05   DD DSN=&&SYSDMP3,DISP=(OLD,DELETE)
//*
//
:/ ADD NAME=PNCI20$SUG
       PCB     TYPE=DB,DBDNAME=PACDAE$SUF,PROCOPT=GOT,KEYLEN=12
       SENSEG  NAME=PAC7AE
       PSBGEN  PSBNAME=PNCI20$SUG,LANG=COBOL,CMPAT=YES
       END
:/ ADD NAME=PNDC10$SUG
       PCB     TYPE=DB,DBDNAME=PACDAN$SUF,PROCOPT=GOT,KEYLEN=43
       SENSEG  NAME=PAC7AN
       PCB     TYPE=DB,DBDNAME=PACDAR$SUF,PROCOPT=GOT,KEYLEN=07
       SENSEG  NAME=PAC7AR
       PCB     TYPE=DB,DBDNAME=PACDAE$SUF,PROCOPT=GOT,KEYLEN=12
       SENSEG  NAME=PAC7AE
       PCB     TYPE=DB,DBDNAME=PACDPA$SUF,PROCOPT=A,KEYLEN=37
       SENSEG  NAME=PAC7PA
       PSBGEN  PSBNAME=PNDC10$SUG,LANG=COBOL,CMPAT=YES
       END
:/ ADD NAME=PNDJ10$SUG
       PCB     TYPE=DB,DBDNAME=PACDAE$SUF,PROCOPT=GOT,KEYLEN=12
       SENSEG  NAME=PAC7AE
       PSBGEN  PSBNAME=PNDJ10$SUG,LANG=COBOL,CMPAT=YES
       END
:/ ADD NAME=PNDM05$SUG
       PCB     TYPE=DB,DBDNAME=PACDAR$SUF,PROCOPT=GOT,KEYLEN=07
       SENSEG  NAME=PAC7AR
       PSBGEN  PSBNAME=PNDM05$SUG,LANG=COBOL,CMPAT=YES
       END
:/ ADD NAME=PNDM10$SUG
       PCB     TYPE=DB,DBDNAME=PACDAN$SUF,PROCOPT=GOT,KEYLEN=43
       SENSEG  NAME=PAC7AN
       PCB     TYPE=DB,DBDNAME=PACDAR$SUF,PROCOPT=GOT,KEYLEN=07
       SENSEG  NAME=PAC7AR
       PCB     TYPE=DB,DBDNAME=PACDAE$SUF,PROCOPT=GOT,KEYLEN=12
       SENSEG  NAME=PAC7AE
       PCB     TYPE=DB,DBDNAME=PACDPA$SUF,PROCOPT=A,KEYLEN=37
       SENSEG  NAME=PAC7PA
       PSBGEN  PSBNAME=PNDM10$SUG,LANG=COBOL,CMPAT=YES
       END
:/ ADD NAME=PNDM50$SUG
       PCB     TYPE=DB,DBDNAME=PACDAR$SUF,PROCOPT=GOT,KEYLEN=07
       SENSEG  NAME=PAC7AR
       PSBGEN  PSBNAME=PNDM50$SUG,LANG=COBOL,CMPAT=YES
       END
:/ ADD NAME=PNDV10$SUG
       PCB     TYPE=DB,DBDNAME=PACDAN$SUF,PROCOPT=GOT,KEYLEN=43
       SENSEG  NAME=PAC7AN
       PCB     TYPE=DB,DBDNAME=PACDAR$SUF,PROCOPT=GOT,KEYLEN=07
       SENSEG  NAME=PAC7AR
       PCB     TYPE=DB,DBDNAME=PACDAE$SUF,PROCOPT=GOT,KEYLEN=12
       SENSEG  NAME=PAC7AE
       PCB     TYPE=DB,DBDNAME=PACDPA$SUF,PROCOPT=A,KEYLEN=37
       SENSEG  NAME=PAC7PA
       PSBGEN  PSBNAME=PNDV10$SUG,LANG=COBOL,CMPAT=YES
       END
:/ ADD NAME=PNDV20$SUG
       PCB     TYPE=DB,DBDNAME=PACDAE$SUF,PROCOPT=GOT,KEYLEN=12
       SENSEG  NAME=PAC7AE
       PSBGEN  PSBNAME=PNDV20$SUG,LANG=COBOL,CMPAT=YES
       END
:/ ADD NAME=PNDC90$SUG
       PCB     TYPE=DB,DBDNAME=PACDAR$SUF,PROCOPT=GOT,KEYLEN=07
       SENSEG  NAME=PAC7AR
       PCB     TYPE=DB,DBDNAME=PACDAE$SUF,PROCOPT=GOT,KEYLEN=12
       SENSEG  NAME=PAC7AE
       PSBGEN  PSBNAME=PNDC90$SUG,LANG=COBOL,CMPAT=YES
       END
:/ ADD NAME=PNDU10$SUG
       PCB     TYPE=DB,DBDNAME=PACDAE$SUF,PROCOPT=GOT,KEYLEN=12
       SENSEG  NAME=PAC7AE
       PSBGEN  PSBNAME=PNDU10$SUG,LANG=COBOL,CMPAT=YES
       END
:/ ADD NAME=PNDU20$SUG
       PCB     TYPE=DB,DBDNAME=PACDAE$SUF,PROCOPT=GOT,KEYLEN=12
       SENSEG  NAME=PAC7AE
       PSBGEN  PSBNAME=PNDU20$SUG,LANG=COBOL,CMPAT=YES
       END
:/ ADD NAME=PNDU30$SUG
       PCB     TYPE=DB,DBDNAME=PACDAE$SUF,PROCOPT=GOT,KEYLEN=12
       SENSEG  NAME=PAC7AE
       PSBGEN  PSBNAME=PNDU30$SUG,LANG=COBOL,CMPAT=YES
       END
:/ ADD NAME=PNRPTS$SUG
       PCB     TYPE=DB,DBDNAME=PACDAE$SUF,PROCOPT=GOT,KEYLEN=12
       SENSEG  NAME=PAC7AE
       PSBGEN  PSBNAME=PNRPTS$SUG,LANG=COBOL,CMPAT=YES
       END
:/ ADD NAME=PNXDAT$SUG
       PCB     TYPE=DB,DBDNAME=PACDAE$SUF,PROCOPT=GOT,KEYLEN=12
       SENSEG  NAME=PAC7AE
       PSBGEN  PSBNAME=PNXDAT$SUG,LANG=COBOL,CMPAT=YES
       END
:/ ADD NAME=PNRPEU$SUG
       PCB     TYPE=DB,DBDNAME=PACDAR$SUF,PROCOPT=A,KEYLEN=07
       SENSEG  NAME=PAC7AR
       PSBGEN  PSBNAME=PNRPEU$SUG,LANG=COBOL,CMPAT=YES
       END
//*
//
$%
//*
//ET040   EXEC PGM=IEBGENER
//SYSUT1   DD DSN=NDVR.PACVINS,DISP=OLD,UNIT=$UTAPE,LABEL=(05,SL),
//            VOL=(,RETAIN,SER=$TAPEI)
//SYSUT2   DD DSN=$INDEXP..$ROOT.$ROOT.SY(PACVINS),DISP=SHR
//SYSIN    DD DUMMY
//SYSOUT   DD SYSOUT=$OUT
//SYSPRINT DD SYSOUT=$OUT
//*
//
```

```
//$PRFJ.PROC JOB ($CCPT),'LOADING PROCEDURES',CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
//ET010 EXEC PGM=IEBUPDTE,PARM=NEW
//*************************************************************
//*         CATALOGING THE VA PAC -ENDEVOR PROCEDURES          *
//*************************************************************
//*
//*    *---------------------------------------------------*
//*    *              I M P O R T A N T                    *
//*    *                 -----------                       *
//*    *      BEFORE EXECUTING THIS JOB, REPLACE ALL       *
//*    *         ':/' BY './' UNDER THE EDITOR.            *
//*    *---------------------------------------------------*
//*
//SYSPRINT DD SYSOUT=$OUT
//SYSUT2   DD DSN=$BIBP,DISP=SHR
//SYSIN    DD DATA,DLM='F+'
```

```
1:/ ADD NAME=$RADP.CEND
 //***************************************************************
 //* VA PAC-ENDEVOR 2.5   : INTRA-ENDEVOR INTEGRITY CONTROL
 //***************************************************************
 :/ ADD NAME=$RADP.CIND
 //***************************************************************
 //* VA PAC-ENDEVOR 2.5   : INTER-ENVIRONMENT INTEGRITY CONTROL
 //***************************************************************
 :/ ADD NAME=$RADP.GPND
 //***************************************************************
 //* VA PAC-ENDEVOR 2.5   : PARAMETERS VALORIZATION
 //***************************************************************
 :/ ADD NAME=$RADP.JJND
 //***************************************************************
 //* VA PAC-ENDEVOR 2.5   : CONVERSION OF ARCHIVED JOURNAL
 //***************************************************************
 :/ ADD NAME=$RADP.JRND
 //***************************************************************
 //* VA PAC-ENDEVOR 2.5   : RETRIEVAL  OF ARCHIVED JOURNAL
 //***************************************************************
 :/ ADD NAME=$RADP.LSND
 //***************************************************************
 //* VA PAC-ENDEVOR 2.5   : UPDATING LIBRARIES/SESSIONS TABLE
 //***************************************************************
 :/ ADD NAME=$RADP.MEND
 //***************************************************************
 //* VA PAC-ENDEVOR 2.5   : FORMATTING TRANSACTIONS FOR UPDATE
 //***************************************************************
 :/ ADD NAME=$RADP.RIND
 //***************************************************************
 //* VA PAC-ENDEVOR 2.5   : CREATE 'INFOPAC'     (RETRIEVAL ELEME
 //***************************************************************
 :/ ADD NAME=$RADP.RPND
 //***************************************************************
 //* VA PAC-ENDEVOR 2.5   : CREATE 'VA PAC'     (RETRIEVAL ELEME
 //***************************************************************
 :/ ADD NAME=$RADP.RRND
 //***************************************************************
 //* VA PAC-ENDEVOR 2.5   : RETRIEVAL OF ENDEVOR ELEMENTS
 //***************************************************************
 :/ ADD NAME=$RADP.RTND
 //***************************************************************
 //* VA PAC-ENDEVOR 2.5   : RETRIEVAL 8.0.2 --> 1.6
 //***************************************************************
 :/ ADD NAME=$RADP.REND
 //***************************************************************
 //* VA PAC-ENDEVOR 2.5   : RETRIEVAL 1.6 ----> 2.5
 //***************************************************************
 :/ ADD NAME=$RADP.TYND
 //***************************************************************
 //* VA PAC-ENDEVOR 2.5   : UPDATING ENDEVOR TYPES
 //***************************************************************
 :/ ADD NAME=$RADP.UPND
 //***************************************************************
 //* VA PAC-ENDEVOR 2.5   : ENDEVOR UPDATE
 //***************************************************************
 :/ ADD NAME=$RADP.INND
 //***************************************************************
 //* VA PAC-ENDEVOR 2.5   : DATES OF LOAD MODULES
 //***************************************************************
```

```
//$PRFJ.PSB JOB ($CCPT),'LOADING OF PSBLIB',CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
//ET010   EXEC PGM=IEBCOPY
//*****************************************************************
//*   LOADING OF PSBLIB                                           *
//*****************************************************************
//*  ONLY TO BE EXECUTED IF:                                      *
//*   - YOUR CHOSEN SYSTEM ROOT.....(ROOT PARAMETER) IS :  'P2' *
//*   - THE SUFFIX OF THE DBDS.......(SUF PARAMETER) IS :  '22' *
//*   - THE SUFFIX OF THE BATCH PSBS (SUG PARAMETER) IS :  '22' *
//*  IF ONLY ONE OF THESE CONDITIONS IS NOT MET,                  *
//*  YOU MUST COMPILE THE SOURCES PRESENT IN THE PARAMETERS      *
//*  PDS FILE.                                                    *
//*****************************************************************
//SYSPRINT DD SYSOUT=$OUT
//SYSUT3   DD UNIT=$UWK,SPACE=(TRK,20)
//SYSUT4   DD UNIT=$UWK,SPACE=(TRK,20)
//IND1     DD DSN=NDVR.PSB,DISP=OLD,UNIT=$UTAPE,
//         VOL=(,RETAIN,SER=$TAPEI),LABEL=(06,SL)
//OUT1     DD DSN=$PSBLIB,DISP=SHR
    COPY INDD=((IND1,R)),OUTDD=OUT1
 S M=PNCI2022
 S M=PNDC1022
 S M=PNDC9022
 S M=PNDJ1022
 S M=PNDM0522
 S M=PNDM1022
 S M=PNDM5022
 S M=PNDU1022
 S M=PNDU2022
 S M=PNDU3022
 S M=PNDV1022
 S M=PNDV2022
 S M=PNRPTS22
 S M=PNRPEU22
 S M=PNXDAT22
/*
//
```

```
//$PRFJ.PRE JOB ($CCPT),'PREPARATION',CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
//*
//*--------------------------------------------------------------
//*          JOB TO BE EXECUTED ONLY ON THE FIRST INSTALLATION
//*--------------------------------------------------------------
//*
//***************************************************************
//* INITIALIZATION OF TABLES AND WORK VSAM FILES
//***************************************************************
//*
//ET010   EXEC PGM=IDCAMS
//*:STEPCAT  DD DSN=$CATU,DISP=SHR
//SYSPRINT DD SYSOUT=$OUT
//SYSIN    DD DSN=$INDEXP..$ROOT.$ROOT.SY(DF$ROOT.$FILE.TS),DISP=SHR
//         DD DSN=$INDEXP..$ROOT.$ROOT.SY(DF$ROOT.$FILE.TY),DISP=SHR
//         DD DSN=$INDEXP..$ROOT.$ROOT.SY(DF$ROOT.$FILE.UP),DISP=SHR
//         DD DSN=$INDEXP..$ROOT.$ROOT.SY(DF$ROOT.$FILE.UQ),DISP=SHR
//*
//***************************************************************
//* LOADING OF UQ
//***************************************************************
//*
//ET020   EXEC PGM=PNINUQ
//STEPLIB  DD DSN=$MODB,DISP=SHR
//PAC7UQ   DD DSN=$INDEX..$ROOT.$FILE.UQ,DISP=SHR
//SYSOUT   DD SYSOUT=$OUT
//SYSUDUMP DD SYSOUT=$OUT
//*
//****************************************************
//*       DELETE DSCB                                *
//****************************************************
//ET030   EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=$OUT
//SYSIN    DD *
  DELETE ($INDEXQ..DSCB.$ROOT.$FILE.QU)
//*
//ET035   EXEC PGM=IEFBR14
//DSCBQU   DD DSN=$INDEXQ..DSCB.$ROOT.$FILE.QU,DISP=(,CATLG,DELETE),
//            SPACE=(TRK,(0)),VOL=SER=$VOLP,UNIT=$UNITP,
//            DCB=(RECFM=FB,LRECL=185,BLKSIZE=18500)
//***************************************************************
//*   BUILDING   INDEX OF GENERATION FILES
//***************************************************************
//*
//ET040   EXEC PGM=IDCAMS
//*:STEPCAT   DD DSN=$CATU,DISP=SHR
//*GDGMOD   DD DSN=$INDEXQ..DSCB.$ROOT.$FILE.QU,DISP=(,CATLG,DELETE),
//*         SPACE=(TRK,0),UNIT=$UNITP,
//*         VOL=SER=$VOLP,
//*         DCB=(RECFM=FB,LRECL=187,BLKSIZE=18700)
//SYSIN    DD DSN=$INDEXP..$ROOT.$ROOT.SY(BL$ROOT.$FILE.QU),DISP=SHR
//SYSPRINT DD SYSOUT=$OUT
//*
//***************************************************************
//* LOADING OF 'QU'
//***************************************************************
//*
//ET050   EXEC PGM=IEBGENER
//SYSIN    DD DUMMY
//SYSPRINT DD SYSOUT=$OUT
//SYSUT1   DD DUMMY,DCB=(RECFM=FB,LRECL=187,BLKSIZE=18700)
//SYSUT2   DD DSN=$INDEXQ..$ROOT.$FILE.QU(+1),DISP=(,CATLG,DELETE),
//            VOL=SER=$VOLO,
//             SPACE=(TRK,(1,1),RLSE),UNIT=$UNITO,
//             DCB=$INDEXQ..DSCB.$ROOT.$FILE.QU
//*
//****************************************************************
//* INITILIZATION OF UP
//****************************************************************
//ET060   EXEC PGM=IDCAMS
//*:STEPCAT  DD DSN=$CATU,DISP=SHR
//SYSPRINT DD SYSOUT=$OUT
//SYSPAF   DD DSN=$INDEX..$ROOT.$FILE.UP,DISP=SHR
//MAXKEY   DD DSN=$INDEXP..$ROOT.$ROOT.SY(MAXKEY),DISP=SHR
//SYSIN    DD DSN=$INDEXP..$ROOT.$ROOT.SY(REPRO999),DISP=SHR
//****************************************************************
//* INITILIZATION OF TS
//****************************************************************
//ET070   EXEC PGM=IDCAMS
//*:STEPCAT  DD DSN=$CATU,DISP=SHR
//SYSPRINT DD SYSOUT=$OUT
//SYSPAF   DD DSN=$INDEX..$ROOT.$FILE.TS,DISP=SHR
//MAXKEY   DD DSN=$INDEXP..$ROOT.$ROOT.SY(MAXKEY),DISP=SHR
//SYSIN    DD DSN=$INDEXP..$ROOT.$ROOT.SY(REPRO999),DISP=SHR
//****************************************************************
//* INITILIZATION OF TY
//****************************************************************
//ET080   EXEC PGM=IDCAMS
//*:STEPCAT  DD DSN=$CATU,DISP=SHR
//SYSPRINT DD SYSOUT=$OUT
//SYSPAF   DD DSN=$INDEX..$ROOT.$FILE.TY,DISP=SHR
//MAXKEY   DD DSN=$INDEXP..$ROOT.$ROOT.SY(MAXKEY),DISP=SHR
//SYSIN    DD DSN=$INDEXP..$ROOT.$ROOT.SY(REPRO999),DISP=SHR
```

```
//$PRFJ.EXT JOB ($CCPT),'EXIT2-EXIT3',CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
//*******************************************************************
//*                                                                 *
//*        LINK-EDIT EXITS IN AN AUTHORIZED LINKLIST LIBRARY        *
//*                                                                 *
//*                         ----------------                        *
//*                                                                 *
//* IMPORTANT: YOU MUST DEFINED EXIT PROGRAMS TO ENDEVOR BY ADDING  *
//*            THEM TO THE aC1UEXIT MACRO IN THE ENDEVOR JCL        *
//*            'BC1JXITS'.                                          *
//*                                                                 *
//*******************************************************************
//ET010   EXEC PGM=IEWL,PARM='LIST,XREF,LET'
//SYSUT1   DD  UNIT=SYSDA,SPACE=(1024,(300,100))
//SYSLIB   DD  DSN=$MODB,DISP=SHR
//         DD  DSN=$LDLIB,DISP=SHR
//SYSLMOD  DD  DSN=$LDLIB,DISP=SHR
//SYSPRINT DD  SYSOUT=*
//SYSLIN   DD  *
 INCLUDE SYSLIB(EPC1UEXT)
 INCLUDE SYSLIB(C1UEXT02)
 ENTRY EPC1UEXT
 NAME C1UEXT02(R)
/*
//*
//ET020   EXEC PGM=IEWL,PARM='LIST,XREF,LET'
//SYSUT1   DD  UNIT=SYSDA,SPACE=(1024,(300,100))
//SYSLIB   DD  DSN=$MODB,DISP=SHR
//         DD  DSN=$LDLIB,DISP=SHR
//SYSLMOD  DD  DSN=$LDLIB,DISP=SHR
//SYSPRINT DD  SYSOUT=*
//SYSLIN   DD  *
 INCLUDE SYSLIB(EPC1UEXT)
 INCLUDE SYSLIB(C1UEXT03)
 ENTRY EPC1UEXT
 NAME C1UEXT03(R)
/*
//*
//
```

```
//$PRFJ.MSG JOB ($CCPT),'LOADING USER MESSAGES',CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
//*
//*********************************************************************
//*        LOADING OF VA PAC MESSAGES IN THE ENDEVOR 'ISPMLIB'       *
//*********************************************************************
//*
//ET010   EXEC PGM=IEBCOPY
//SYSPRINT DD SYSOUT=$OUT
//SYSUT3   DD UNIT=$UWK,SPACE=(TRK,20)
//SYSUT4   DD UNIT=$UWK,SPACE=(TRK,20)
//IN       DD DSN=$INDEXP..$ROOT.$ROOT.SY,DISP=SHR
//OUT      DD DSN=$MSGLIB,DISP=SHR
//SYSIN    DD *
 C I=IN,O=OUT
 S M=CIUU$MSGSX
/*
//*
//
```

```
//$PRFJ.VINS JOB ($CCPT),'UPDATING PACBASE',CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
//*
//*******************************************************************
//*         CREATING USER'S ENTITIES FOR THE VA PAC DATABASE         *
//*******************************************************************
//*
//ET010   EXEC ????VINS       <-- PACBASE "VINS" PROCEDURE
//COPY.CARTE DD *
 *USER    PASSWORD        F
//PACINS.PAC7MV DD DSN=NDVR.PACVINS,DISP=OLD,UNIT=$UNITP,
//       LABEL=(05,SL),VOL=(,RETAIN,SER=$VOLP)
//**************************************************************
//*                       !!!! CAUTION   !!!!
//*         THE FOLLOWING STEPS MUST BE EXECUTED ONLY FOR A
//*                      RETRIEVAL 1.6 -> 2.5
//**************************************************************
//*
//*T020   EXEC $RADP.JJND,
//*       JNARCH='????'       <-- ARCHIVED JOURNAL 1.6
//*
//*T030   EXEC ????SAVE       <-- PACBASE "SAVE" PROCEDURE RECOMM
//*
//*T040   EXEC $RADP.REND
//*
//
```

```
//$PRFJ.INND JOB ($CCPT),'INSTALLATION LIST',CLASS=$CLASSJ,
// MSGCLASS=$MSGCL
// JCLLIB ORDER=($BIBP)
//*****************************************************************
//*    THE FOLLOWING JOB PROVIDES THE LIST OF INSTALLED PROGRAMS   *
//*****************************************************************
//*
//STEP01  EXEC $RADP.INND
//*
//
```

10. RE-INSTALLATION

10.1. RE-INSTALLATION PROCEDURE

RE-INSTALLATION OF THE SYSTEM

The VA Pac-ENDEVOR system must be re-installed whenever a new sub-version of the system comes out.

This sub-version, identified by a number (for example 2.5 V02), is generally delivered in the form of:

. A complete installation tape of the product,

. A list of corrected bugs,

. Possibly, an operations document which completes the instructions described in the present sub-chapter.

Generally, only the program library is affected by this new sub-version.

In most of the cases, the re-installation consists in executing the following JOBS:

1. PACBASE0, which ensures the copy of the installation tape (JOB contained in the initial JCL file).

2. PACBASE2, which ensures the creation of the re-installation JCL.

PACBASE2 executes the MM1JCL utility with the parameters given on installation, plus the following selection line of the JCL module:

```
        ===SELM NDVRLOAD
```

Follow the installation procedure described in Chapter INSTALLATION, Subchapter PHASE 3: INSTALLATION PROCESS, paragraph "Loading of Load Modules".

If the operations document contains corrections in one of the EXITs, add the following selection line of the JCL module:

```
        ===SELM NDVREXIT
```

Follow the installation procedure described in Chapter INSTALLATION, Subchapter PHASE 3: INSTALLATION PROCESS, paragraph "Link-Edit of Exits in an Authorized Library".

If the operations document contains corrections in the parameter PDS, add the following selection line of the JCL module:

```
===SELM NDVRSY
```

Follow the installation procedure described in Chapter INSTALLATION, Subchapter PHASE 3: INSTALLATION PROCESS, paragraph "Loading of the System Parameters PDS".

NOTE: If at least one PROCESS has been modified, it must be re-installed taking into account the information specified in Chapter INSTALLATION, Subchapter PHASE 1: ENVIRONMENT PREPARATION, paragraph "Allocation of Processors to the PROCESSOR GROUP".

If the operations document contains corrections in at least one batch procedure, add the following selection line of the JCL module:

```
===SELM NDVRPROC
```

Follow the installation procedure described in Chapter INSTALLATION, Subchapter PHASE 3: INSTALLATION PROCESS, paragraph "Loading of Operations Procedures".

If the operations document contains corrections in the VA Pac update transaction file of the user entities, add the following selection line of the JCL module:

```
===SELM NDVRVINS
```

Follow the installation procedure described in Chapter INSTALLATION, Subchapter PHASE 3: INSTALLATION PROCESS, paragraph ".NDVLM User Entity and .NDVRL Relation".

SWITCHING FROM VERSIONS OLDER THAN 8.0.2 V02I

From the sub-version 8.0.2 V02I, the length of the UQ journal file and of the QU archived journal has changed.

When switching from older versions to 8.0.2 V02I, the following operations must be carried out :

- Execute the MEND procedure to upload, in VA Pac, the last transactions linked with Endevor actions on the VA Pac elements,

- Back up the QU last archived journal file (generation 0), 135 characters long,

- Install the 8.0.2 V02I sub-version of the interface,

- Execute the JJND procedure, using in input the backup file of the archived journal previously created This procedure produces a new archived journal file (185 characters long) valid for the 8.0.2 V02I sub-version.