VisualAge Pacbase 2.5

AS/400 DBD
REFERENCE MANUAL

DDO38000021A

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.software.ibm.com/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 (December 1993)

This edition applies to the following licensed programs:

- VisualAge Pacbase Version 2.0
- 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.software.ibm.com/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. PACBASE FUNCTIONS

THE VisualAge Pacbase Application Development Solution

VisualAge Pacbase is an Application Development tool operating on mainframe, OS/2, UNIX or Windows NT. It has been designed to ensure the complete management of various information systems.

Consistency is ensured by all the data being stored in one Specification database and managed in a unique way by the System.

VISUALAGE PACBASE PRODUCTS

VisualAge Pacbase is a modular AD solution which is composed of two main products - Pacdesign for application design, Pacbench for application development.

Pacdesign and Pacbench are used to populate the Specifications Database and to ensure the maintenance of existing applications. Each product includes several functions.

Basic Functions

Dictionary
Structured Code
Personalized Documentation Manager (PDM-PDM+)

Generators

On-Line Systems Development
Client/Server Facility
Batch Systems Development
COB / Generator

Database Description

DBD
DBD-SQL

Application Revamping

Pacbench Automatic Windowing (PAW) (releases older than VisualAge Pacbase 2.0)

Pacbase Web Connection

Quality Control

Pacbench Quality Control (PQC)
Quality Control Extensibility

Table Management

Pactables

Production Turnover and Follow-up

Production Environment (PEI)
PacTransfer
Development Support Management System (DSMS)
PC function: revamped DSMS (in releases older than VisualAge Pacbase 2.0)

Additionnal services
Pac/Impact
Dictionary Extensibility
Pacbase Access Facility (PAF-PAF+)
DSMS Access Facility (DAF)
Methodology (Merise, YSM, etc.)
Sub-networks comparison utilities
Rename/move entity utility (RMEN)
Journal Statistics utility (ACTI)
RACF / TOPSECRET Security Interface
ENDEVOR
VisualAge Smalltalk-VisualAge Pacbase bridge
Team Connection-VisualAge Pacbase bridge

1.2. INTRODUCTION TO THE DATABASE DESCRIPTION FUNCTION

INTRODUCTION TO THE D.B.D. FUNCTION

The Database Description function automatically generates database descriptions adapted to the database management system in use. This is done by using segment and relationship descriptions defined during the application analysis phase.

The DBD function can generate the description of the following DBMS's:

. Relational databases,

. Network databases (CODASYL),

. Hierarchical databases (DL/1),

. Physical File - AS/400 databases and TANDEM DDL,

. TurboImage databases,

. DMSII databases.

Each one of these DBMS's is documented in a specific Reference Manual.

1.3. PRINCIPLES OF DESCRIPTION

DESCRIPTION PRINCIPLES

In this manual, the entities and screens managed by VisualAge Pacbase are described in two parts:

. An introductory comment explaining the purpose and the general characteristics of the entity or screen,

. A detailed description of each screen, including the input fields for both on-line (screens) and batch (forms) data entry into the Database.

Since input screens and batch forms usually contain the same fields, their descriptions are often identical.

All on-line fields described in this manual are assigned an order number. These numbers are printed in bold italics on the screen examples which appear before the input field descriptions and allow for easy identification of a given field. The numbers are circled on the batch forms.

For certain descriptions, there may be slight differences between the screen and the corresponding batch form. This can be explained by the fact that batch mode is less flexible than on-line mode and often needs additional input fields for some indicators which already exist on the screen.

In addition, the user may find that the field sequence on a screen is different from the field sequence on the corresponding batch form. If that occurs, the numbers referencing the fields may not appear in ascending sequence on either the screen example or the batch form.

>>>> If you use the VisualAge Pacbase WorkStation, the graphical interface of the corresponding windows is described in the VisualAge Pacbase WorkStation Reference Manual.

NOTES: Each type of Database Block has a specific description. However, several Database Block types may use the same Batch Form.

As a result, fields on the Batch Form may have different meanings or may not be used, depending on the type of Database Block.

2. USE OF THE FUNCTION

2.1. INTRODUCTION

INTRODUCTION

This database description reference manual is not a training manual for the IBM AS400 system.

A prior knowledge of the IBM AS400 system and the Dictionary function is required.

It is particularly important to understand the following entities:

. Data Elements,

. Segments,

. Parameterized Input Aids.

Using an example, this manual guides the user through the description and generation of a physical file.

This function is compatible with the IBM System 38.

2.2. USE OF PACBASE ENTITIES

USE OF PACBASE ENTITIES

ROLE OF THE DICTIONARY

The Dictionary manages a logical description of various external views to be transmitted to the programs. It uses the following entities:

. Data Elements,

. Segments,

. Database Blocks and the associated General Documentation lines,

. Parameterized Input Aids.

TERMINOLOGY CORRESPONDENCE

A Physical File consists of Records, which are composed of Fields (or Elements).

A Database Block consists of Segments, which call in the Data Elements.

The correspondence table below is based on this analogy:

AS400 OBJECT	PACBASE Equivalent
Physical File	Database Block
Record	Segment
Field	Data Element

All the required elements for a physical file data description exist in the System. Refer to the SPECIFICATIONS DICTIONARY Reference Manual, which describes the Data Elements and Segments.

COMMENTS ON THE DATA ELEMENTS

When data elements are called into a segment, a control break or numeric key between '0' and '9' is entered. This number identifies the keys and their sequence.

The group data elements are ignored unless they are repeated. In this case, elementary and group fields belonging to this data element are ignored at generation. The generated length is equal to the length of the group data element multiplied by the number of repetitions.

The same formula is used for repeated data elements. In this case, the numeric data elements with decimals are generated without the position of the decimal indicated in the DDS.

3. THE PHYSICAL FILES

3.1. DEFINITION OF A DATABASE BLOCK

DEFINITION OF A DATABASE BLOCK

A physical file is defined using a Database Block, which is accessed via the following CHOICE:

CH: B.....

This physical file is defined using a code, a clear name and a specific TYPE whose value is 'PF'.

The TYPE can also have the 'LF'(Logical File) value. In this case, it is only documentary and the generation is identical to that obtained with type 'PF'.

The System generates a DDS source file corresponding to the physical file characteristics, from the Database Block.

The clear name is generated in the TEXT function of the RECORD clause.

The data element clear name is generated in the TEXT function of each field.

If a column heading appears in the data element description screen, it is generated in the COLHDG function associated with the field.

```
--------------------------------------------------------------------------------
!                   IBM SYSTEM 38                              *PDLC.NDOC.A38.28!
!                                                                              !
! BLOCK DEFINITION.............:    BLOCPF 1                                   !
!                                                                              !
! NAME.................: DDS ORDERS FILE     2                                 !
! TYPE.................: PF S38 PHYSIC.FILE  3                                 !
! VERSION..............:  4                                                   !
!                                                                              !
!                                                                              !
! EXTERNAL NAME.........:  5                                                   !
!                                                                              !
!                                                                              !
! CONTROL CARDS....... FRONT: 6       BACK: 7                                  !
!                                                                              !
!                                                                              !
! EXPLICIT KEYWORDS..: 8                                                       !
!                                                                              !
!                                                                              !
! SESSION NUMBER.....: 0026        LIBRARY......: A38    LOCK....:             !
!                                                                              !
!                                                                              !
!                                                                              !
!                                                                              !
! O: C1 CH: Bblocpf                    ACTION:                                 !
--------------------------------------------------------------------------------
```

NUM	LEN	CLASS VALUE	DESCRIPTION OF FIELDS AND FILLING MODE
1	6		BLOCK CODE (REQUIRED) One to six alphanumeric characters.
2	36		NAME OF THE BLOCK (REQ. IN CREATION) This clear name should be as explicit as possible. Words used here become implicit keywords (subject to limitations specified in Subchapter "HOW TO BUILD THE THESAURUS", Chapter "KEYWORDS" in the SPECIFICATIONS DICTIONARY Reference Manual).
3	2		TYPE OF BLOCK (REQ. IN CREATION) For hierarchical or network databases, it is not necessary, when creating a database block, to enter the definitive block type. The selection of a network or hierarchical structure is sufficient at this point. A specific "physical" type must be entered when generating the Data Description Language (DDL).
		TR	Tree-like structure (hierarchical block).
		SE	Group of sets (network block).
			HIERARCHICAL DATABASES - IMS/DL1 ----------
		DP	Physical Database Description.
		DR	Physical Database Description (same as 'DP', but only the data elements referenced as access keys in the segment description are generated in the 'FIELD.....' statements).
		DL	Logical Database Description.
		PC	PCB.
		IP	Primary Index.
		IS	Secondary Index.
		PS	PSB (Assigned at creation. Cannot be modified at a later stage).
			RELATIONAL DATABASES ----------
		Q2	DB2 SQL
		Q3	SQL SERVER
		Q4	DB2/400
		QA	ALLBASE/SQL
		QB	DB2/2 and DB2/6000
		QC	DATACOM/DB
		QG	INGRES/SQL
		QI	INFORMIX-ESQL
		QN	NONSTOP SQL

NUM	LEN	CLASS VALUE	DESCRIPTION OF FIELDS AND FILLING MODE
		QO	ORACLE (releases earlier than V6)
		QP	ORACLE (from release V6 on)
		QR	RDMS
		QS	SQL/DS
		QT	INTEREL RDBC
		QU	INTEREL RFM
		QV	VAX SQL
		QY	SYBASE
		DB	DB2 (It is recommended to use the Q2 type)
			NETWORK DATABASES

			.CODASYL-DM4 (BULL 66 or DPS8):

		M1	DDL schema, only elementary fields are generated,
		M4	DDL schema, only group fields are generated,
		M2	DMCL schema,
		M3	Sub-schema.
			.CODASYL-IDS2 (BULL 64 or DPS7):

		I1	DDL schema,
		I2	DMCL schema,
		I3	SDDL sub-schema.
			.CODASYL-IDMS:

		D0	DDL schema (Release 10.0),
		D1	DDL schema,
		D2	DMCL schema,
		D3	Sub-schema,
		D4	Sub-schema (Release 5.7).
			.CODASYL-DMS (UNISYS 1100):

		S1	DDL Schema,
		S3	Sub-schema.
			DDL TANDEM

		TD	TANDEM
			AS/400 PHYSICAL FILE

NUM	LEN	CLASS VALUE	DESCRIPTION OF FIELDS AND FILLING MODE
		PF	AS/400 Physical file (IBM SYS. 38)
		LF	AS/400 Logical file (IBM SYS. 38).
			DDL TURBOIMAGE ----------
		TI	TurboImage Database.
			DMSII DATABASE ----------
		20	DMSII Database (DASDL)
4	4		VERSION This field is not used.
5	8		DATABASE BLOCK EXTERNAL NAME Necessary at generation time. This is the physical name of the System-generated DDL (Data Description Language) module. To obtain a list of blocks sorted by this external name, enter 'LEB' in the CHOICE field. For TurboImage, only the first six characters are processed.
6	1		CONTROL CARDS IN FRONT OF BLOCK Necessary at generation time. Enter the one-character code that identifies the job control card to be inserted before the generated block.
7	1		CONTROL CARDS IN BACK OF BLOCK Necessary at generation time. Enter the one-character code that identifies the job control card to be inserted after the generated block.
8	55		EXPLICIT KEYWORDS This field allows the user to enter additional (explicit) keywords. By default, keywords are generated from an occurrence's clear name (implicit keywords). This field only exists on-line. In batch mode, keywords are entered on Batch Form 'G'. Keywords must be separated by at least one space. Keywords have a maximum length of 13 characters which

NUM	LEN	CLASS VALUE	DESCRIPTION OF FIELDS AND FILLING MODE
			must be alphanumeric. However, '=' and '*' are reserved for special usage, and are therefore not permitted in keywords. Keywords are not case-sensitive: upper-case and lower-case letters are equivalent. NOTE: Characters bearing an accent and special characters can be declared as equivalent to an internal value in order to facilitate occurrence search by keywords. Refer to the Operations Manual - Part II "Administrator's Guide", Chapter "Database Management Utilities", Subchapter "PARM: Update of User Parameters". A maximum of ten explicit keywords can be assigned to one entity. For more details, refer to Chapter "KEYWORDS" Subchapter "BUILDING THE THESAURUS" in the SPECIFICATIONS DICTIONARY Reference Manual.

3.2. DESCRIPTION OF A DATABASE BLOCK

DESCRIPTION OF A DATABASE BLOCK

A physical file, which is similar to a hierarchical database, is described on the Database Block Description screen which is accessed via the following CHOICE:

CH: B.....DH

The Description consists of records which make up the physical file.

On the '-DH' screen, the user calls in the relevant Segment(s) by entering the Segment code(s). There is a COMMENT field available for additional information on each called Segment.

The record name generated is the value of the Data Structure of the corresponding Segment code, which is coded as follows: 'NNNNNN ' (six characters maximum), or if this is absent, it is the Segment name.

COMPLEMENTARY DESCRIPTION

The Database Block is defined and described using two screens: 'B.....' and 'B.....DH'. As indicated above, the user fills in the Segment codes corresponding to the Records to be generated, for which the Specifications Dictionary builds its cross-references.

For each Database Block Description line (one per Record), there is a General Documentation (-G) screen accessed via the following input in the CHOICE field:

B.....DH...G

This screen has a number of functions:

. Displaying descriptions generated by default,

. Replacing descriptions generated by default,

. Completing descriptions generated by default,

. Adding other descriptions to be generated,

. Entering comments.

VIRTUAL LINES

The purpose of these lines is to display the DDS which is generated "virtually", to give the user the option of replacing or completing the DDS generated by default, by inserting lines within the virtual lines.

As long as these lines are not modified by the user, they will be generated as they appear. By definition, the virtual lines displayed vary according to whether the General Documentation is linked to a Definition screen or to a Description screen line, depending on the type of line.

On the General Documentation screen of the associated Description, the virtual lines will be marked with an '*' in the ACTION CODE field and '*VIRT' in the COMMENT field.

PARAMETERIZED INPUT AIDS

The Parameterized Input Aid (P.I.A.) is an entity managed in the Specifications Dictionary. It allows the user to define his/her own standards in terms of the Data Description Language according to his/her needs. In contrast to the virtual lines, the Parameterized Input Aids are only displayed if called in by the user.

A particulary interesting feature of the P.I.A. is that when describing a P.I.A., an index name can be linked with a description line so that a cross-reference can be established between the index name and the values of the Parameterized Input Aid. These are then user-created cross-references.

For each record in a PF-type database Block, there are the following description lines:

```
100 G RECORD LEVEL
200 G ---> FIELD  BEGIN ('FUNCTIONS:')
400 G ---> FIELD  END
600 G ---> KEY    BEGIN ('FUNCTIONS:')
800 G ---> KEY    END
850 G ---> ADD  KEY  BEGIN
900 G ---> ADD  KEY  END
```

A line entered outside the 200-400, 600-800 and 850-950 line number ranges is generated according to the following rule:
The inputted line is generated as entered, and begins in column 7 of the generated line.
Since the information entered on these lines refers to a given field, this field must first be specified on a line between the "<" and ">" signs and the information to be generated must be entered on the line that immediately follows.

This operation entails a complete replacement of the line generated by default, or a data element deletion if no line follows the '<DELCO >' line. (There must be exactly six characters between '<' and '>').

If the user wants to specify keyword(s) other than those of the standard generated code, he/she should enter them using the reserved word 'FUNCTIONS:'.

The 850-950 range can be used to enter additional keys if the 10 automatically-generated keys are not sufficient.

EXAMPLE:

```
050 G * Comments                       <- added comments
100 G RECORD LEVEL

200 G ---> FIELD  BEGIN
300 G <DELCO1>
305 G FUNCTIONS:COLHDG('Order' 'Date') <- added keyword
380 G <DELCO3>                         <- DELCO3 deletion
400 G ---> FIELD  END

600 G ---> KEY    BEGIN
750 G <DELCO2>
755 G FUNCTIONS:DESCEND
800 G ---> KEY    END
```

```
!                    IBM SYSTEM 38                              *PDLC.NDOC.A38.28!
! DDL BLOCK PHYSICAL FILES          1 BLOCPF DDS ORDERS FILE                      !
! 2 3     4    5    6     7  8   9     10                                        !
! A LIN : SEGM      MODEL    DOC       COMMENT                          LIBR.!
!   100 : PF05                 *                                       0026 !
!       :                                                                      !
!       :                                                                      !
!       :                                                                      !
!       :                                                                      !
!       :                                                                      !
!       :                                                                      !
!       :                                                                      !
!       :                                                                      !
!       :                                                                      !
!       :                                                                      !
!       :                                                                      !
!       :                                                                      !
!       :                                                                      !
!       :                                                                      !
!       :                                                                      !
! *** END ***                                                                  !
! O: C1 CH: -DH                                                                !
```

NUM	LEN	CLASS VALUE	DESCRIPTION OF FIELDS AND FILLING MODE
1	6		BLOCK CODE (REQUIRED) One to six alphanumeric characters.
2	1		ACTION CODE
3	3		LINE NUMBER PURE NUMERIC FIELD It is advisable to begin with line number '100' and then number in intervals of 20. This facilitates subsequent line insertions, as necessary.
4	4		SEGMENT CODE (REQ. IN CREATION) This field is entered with the PACBASE Segment Code.
5	4		PARENT SEGMENT CODE This is the code of the segment upon which the given segment is hierarchically dependent. FOR INDEX-type DBD's: This field is not used for 'IP'- or 'IS'-type Data-base Blocks.
6	6		MODEL ENTITY RELATIONSHIP CODE OPTIONAL INPUT FIELD: Code of the Model Relationship corresponding to the DL/1 Relationship. The System automatically creates the cross-references of the Model Relationship to DL/1 Relationships. NOTE: Model Relationships are described through the PACMODEL function.
7	1		KEY INDICATOR Used for a symbolic reference of the key data element of a given segment in a given DBD. The character indi-cated in this field must also appear on the Segment Call of Elements (-CE) screen in the KEY INDICATOR FOR ACCESS OR SORT field, on the key data element line.
		U	References a unique key.
		M	References a multiple key.
		1 to 9	DL/1 Secondary index.
		$	In a PCB or a physical or logical DBD (Block type PC, DB, or DL), generates a non-qualified SSA (used in OLSD).
			All other values designate a search field.

NUM	LEN	CLASS VALUE	DESCRIPTION OF FIELDS AND FILLING MODE
			NOTE: Sort keys are not permitted on data elements redefining other data elements (see the Segment Call of Elements (-CE)).
8	1		DOCUMENTATION INDICATOR This field is a display field used on-line only. It does not accept input.
		*	General documentation exists for the element on this line. Access to line nnn: -CEnnn Access to the documentation of line nnn: -CEnnnG For more details, see the "GENERAL DOCUMENTATION" chapter in the SPECIFICATIONS DICTIONARY Reference Manual.
9	5		EST. NUMBER OF CHILD/PARENT LINKS This is the average number of occurrences of a child segment linked to one occurrence of its parent segment.
10	36		COMMENT / RELATIONSHIP / KEY LENGTH When generating "PS"-type Database Blocks, i.e. a PSB, the DBD function automatically calculates the the length of the longest concatenated key. This is done for: . Each DBD called in a PSB, . Each PCB called in a PSB, . Each INDEX Database called as an independent data-base in the PSB. This length may be overridden by entering the follow-ing input on the first line: CC=n (with n = 9 to 9999). On each segment call line, the user may enter: . Comments, or . PR=nnnn, used to generate the parameter PROCOPT=nnnn at the SENSEG Statement level when generating the PSB containing this DBD, PCB, or INDEX Database. NOTE: This calculation is done only for a primary Segment. In the case of a secondary index, the CC= parameter is required.

3.3. ON-LINE ACCESS COMMANDS

DATABASE BLOCKS: ON-LINE ACCESS

LISTS

CHOICE	SCREEN	UPD
LCBaaaaaa	List of database blocks by code (starting with block 'aaaaaa').	NO
LTBaabbbbbb	List of database blocks by type (starting with type 'aa' and block 'bbbbbb').	NO
LEBaaaaaaaa	List of database blocks by external name (starting with name 'aaaaaaaa').	NO

DESCRIPTION OF BLOCK 'aaaaaa'

CHOICE	SCREEN	UPD
Baaaaaa	Definition of database block 'aaaaaa'	YES
BaaaaaaGbbb	General documentation for block 'aaaaaa' (starting with line 'bbb').	YES
BaaaaaaATbbbbbb	Text assigned to block 'aaaaaa' (starting with text 'bbbbbb').	NO
BaaaaaaX	X-references of block 'aaaaaa'.	NO
BaaaaaaXBbbbbbb	X-references of block 'aaaaaa' to PSB's (starting with PSB 'bbbbbb').	NO
BaaaaaaXObbbbbb	X-references of block 'aaaaaa' to screens (starting with screen 'bbbbbb').	NO
BaaaaaaXObbbbbbCScdddd	X-references of block 'aaaaaa' to the Call of Segments of screen 'bbbbbb' (starting with category 'c' and with segment 'dddd'). Note: 'c' is equal to & for the screen-top category.	NO
BaaaaaaXObbbbbbWccddd	X-references of block 'aaaaaa' to the Work Areas of screen 'bbbbbb' (starting with work area 'cc', line number'ddd').	NO
BaaaaaaXQbbbbbb	List of entities linked to block 'aaaaaa' through user-defined relation-ship 'bbbbbb'.	NO
BaaaaaaXVvvvvvv	X-references of block 'aaaaaa' to volumes (starting with volume 'vvvvvv').	NO
BaaaaaaXPbbbbbb	X-references of block 'aaaaaa' to programs (starting with program 'bbbbbb').	NO
BaaaaaaXPbbbbbbWccddd	X-references of block 'aaaaaa' to Work Areas of program 'bbbbbb' (starting with work area 'cc', line number 'ddd').	NO

BaaaaaaDHbbb	Description of hierarchical block 'aaaaaa' (starting with line 'bbb')	YES
BaaaaaaDHbbbGccc	General documentation of hierarchical block 'aaaaaa' description line 'bbb' (starting with general documentation line 'ccc').	YES

NOTES: After the first choice of type 'Baaaaaa', 'Baaaaaa' can be replaced with '-'.

All notations between parentheses are optional.

3.4. BATCH ACCESS COMMANDS

DATABASE BLOCKS: BATCH ACCESS

DEFINITION

Batch Form 'L1' is used to define a Database Block.

ACTION CODES

```
C     = Creation of a line in the library.

M     = Modification of a line.

Blank = Creation or modification of a line,  depending
        on its presence or absence in the library.

X     = Creation or modification with possible use of
        ampersands (&).

D     = Deletion of a line.

B     = Deletion of the database block and of its de-
        pendent lines.
```

DATABASE BLOCK DESCRIPTION

BATCH FORM

Batch Form 'L2' is used to describe a hierarchical block.

The General Documentation associated with description lines is entered on Batch Form 'V3' using field 8 (the ENTITY LINE NUMBER) for the association.

ACTION CODES

```
C     = Creation of a line in the library.

M     = Modification of a line.

Blank = Creation or modification of a line,  depending
        on its presence or absence in the library.

X     = Creation or modification with possible use  of
        ampersands (&).

D     = Deletion of a line.

B     = Deletion of database block starting with this
        line (including associated documentation
        lines).

R     = End of multiple deletion.
```

If a 'B' line is not followed by an 'R' line, the deletion ends with the last line of the block.

3.5. GENERATION AND/OR PRINTING

GENERATION AND/OR PRINTING

The generation and printing of Database Blocks is requested on-line on the Generation and Print Commands screen (CH: GP) or in batch mode on Batch Form 'Z'.

LISTS

```
LTB      Lists all database blocks of the libraries from the
         selected sub-network, sorted by type.

        .C1 OPTION: Without keywords,
        .C2 OPTION: With explicit keywords.

LCB      Identical to 'LTB' but sorted by code.

LEB      Identical to 'LTB' but sorted by external name.
```

It is possible to request a list of Database Blocks related by keyword(s). The corresponding command must be accompanied by a continuation line, on which the keywords used as selection criteria are indicated (refer to the USER'S Reference Manual). The list is sorted by code.

```
LKB      Same as 'LCB' but sorted by keyword.
         Option 'C2' cannot be used.
```

DESCRIPTION

```
DTB      Description of the database block whose code is in-
         dicated in the entity field, description of all da-
         tabase blocks if the field is not entered.
         In the latter case,  it is possible to request the
         descriptions of all blocks of a given type, by spe-
         cifying it in the printing request.
```

GENERATION OPTION

```
GCB      Generation of a Database Block whose code must be
         indicated.
         Same printing option as for DTB.
```

4. EXAMPLE

4.1. INTRODUCTION TO THE EXAMPLE

INTRODUCTION TO THE EXAMPLE

The purpose of this chapter is to give the user an overall view of the various steps leading to the generation of a Physical File, and provide additional information on the processing of this data.

This is not a comprehensive example. It does not cover all of the functionality of the DBD function.

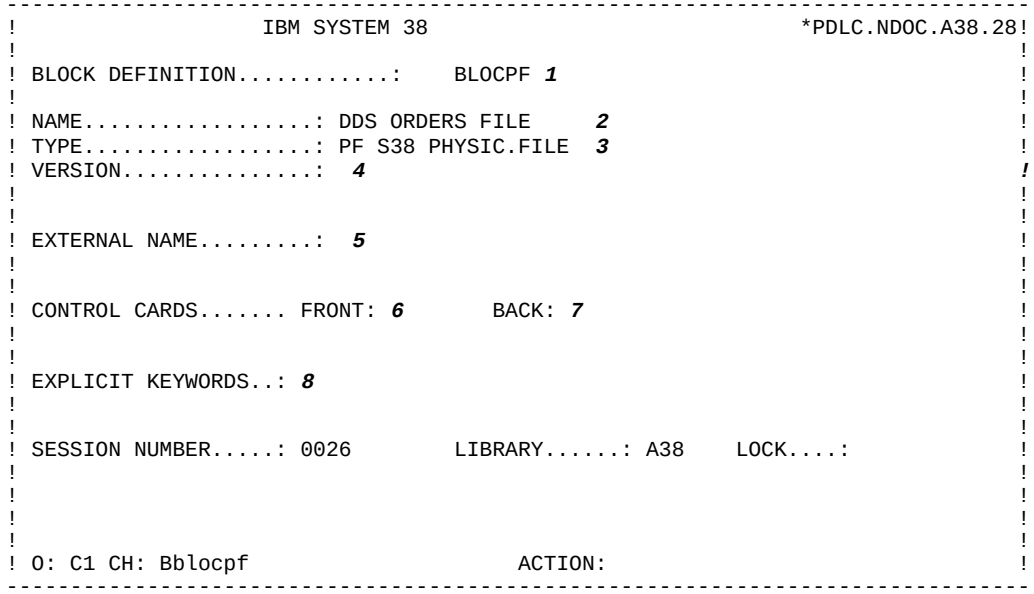
The example includes the main screens used for generating the Database Block, and the generated code itself.

4.2. PACBASE SCREENS

```
--------------------------------------------------------------------------------
!                   IBM SYSTEM 38                              *PDLC.NDOC.A38.28!
!                                                                              !
! SEGMENT DEFINITION.........: PF05                                           !
!                                                                              !
! NAME.......................: ORDER INFORMATIONS                              !
!                                                                              !
! OCCUR. OF SEGMENT IN TABLE:                                                  !
! EST. NUMBER OF INSTANCES..:                                                  !
!                                                                              !
!                                                                              !
!                                                                              !
! VALUE OF RECORD TYPE ELEM.: 'ORDER   '                                       !
! CODE OF ACTION CODE ELEM..:                                                  !
! PRESENCE...................: CR:       MO:       DE:                         !
!                              M4:       M5:       M6:                         !
!                                                                              !
!                                                                              !
! EXPLICIT KEYWORDS..:                                                         !
!                                                                              !
!                                                                              !
! SESSION NUMBER.....: 0026        LIBRARY......: A38    LOCK....:             !
!                                                                              !
!                                                                              !
! O: C1 CH: SPF05                       ACTION:                                !
--------------------------------------------------------------------------------
```

```
--------------------------------------------------------------------------------
!                   IBM SYSTEM 38                          *PDLC.NDOC.A38.28!
! SEGMENT CALL OF ELEMENTS PF05 ORDER INFORMATIONS                             !
!                                                                              !
! A LIN : ELEM.  INT.FORM. U OCC GR K CMD456 CONT VALUE/SFC  UPD/TRGET DOC LIBR!
!   100 : NUCLIE 9(8)      D        1                                  0026!
!   110 : DATE   D         D                                           0026!
!   120 : RELEA  X(3)      D                                           0026!
!   130 : REFCLI X(30)     D                                           0026!
!   140 : RUE    X(40)     D                                           0026!
!   150 : COPOS  X(5)      D                                           0026!
!   155 : VILLE  X(20)     D                                           0026!
!   160 : CORRES X(25)     D                                           0026!
!   170 : REMIS  S9(4)V99  D                                           0026!
!   180 : MATE   X(8)      D                                           0026!
!   185 : LANGU  X         D                                           0026!
!       :                                                                  !
!       :                                                                  !
!       :                                                                  !
!       :                                                                  !
!       :                                                                  !
!       :                                                                  !
!       : NAME    :                                                        !
! *** END ***                                                              !
! O: C2 CH: -CE                                                            !
--------------------------------------------------------------------------------
```

EXAMPLE **4**
PACBASE SCREENS **2**



```
--------------------------------------------------------------------------------
!                    IBM SYSTEM 38                          *PDLC.NDOC.A38.28!
! DDL BLOCK PHYSICAL FILES            BLOCPF DDS ORDERS FILE                    !
!                                                                               !
! A LIN : SEGM      MODEL    DOC       COMMENT                          LIBR.!
!   100 : PF05                 *                                        0026 !
!       :                                                                       !
!       :                                                                       !
!       :                                                                       !
!       :                                                                       !
!       :                                                                       !
!       :                                                                       !
!       :                                                                       !
!       :                                                                       !
!       :                                                                       !
!       :                                                                       !
!       :                                                                       !
!       :                                                                       !
!       :                                                                       !
!       :                                                                       !
!       :                                                                       !
!       :                                                                       !
!       :                                                                       !
! *** END ***                                                                   !
! O: C1 CH: -DH                                                                 !
--------------------------------------------------------------------------------
```

```
--------------------------------------------------------------------------------
!                   IBM SYSTEM 38                          *PDLC.NDOC.A38.28!
! BLOCK DESC GENERAL DOC.         BLOCPF DDS ORDERS FILE                 100!
!                                                                          !
! A LIN : T COMMENT                                                 LIB    !
! * 100 : G RECORD LEVEL                                            *VIRT  !
! * 200 : G ---> FIELD  BEGIN ('FUNCTIONS:')                        *VIRT  !
!   210 : G <DATE  >                                                0026   !
!   220 : G FUNCTIONS:COLHDG('DATE' 'ORDERS')                       0026   !
! * 400 : G ---> FIELD  END                                         *VIRT  !
! * 600 : G ---> KEY    BEGIN ('FUNCTIONS:')                        *VIRT  !
!   610 : G <NUCLIE>                                                0026   !
!   620 : G FUNCTIONS:DESCEND                                       0026   !
! * 800 : G ---> KEY    END                                         *VIRT  !
!   850 : G ---> ADD  KEY  BEGIN                                           !
!   900 : G ---> ADD  KEY  END                                             !
!       :                                                                  !
!       :                                                                  !
!       :                                                                  !
!       :                                                                  !
!       :                                                                  !
!       :                                                                  !
!       :                                                                  !
! *** END ***                                                              !
! O: C1 CH: -DH100G                                                        !
--------------------------------------------------------------------------------
```

4.3. GENERATED DATABASE BLOCK

```
A      R COMMDE                    TEXT('ORDER PHYSICAL FILE   ')
A        NUCLIE          8S
A                                  TEXT('CLIENT NUMBER')
A        DATE            6A        COLHDG('DATE' 'ORDERS')
A                                  TEXT('DATE')
A        RELEA           3A
A                                  TEXT('RELEASE')
A        REFCLI         30A
A                                  TEXT('CLIENT REFERENCE')
A        RUE            40A
A                                  TEXT('STREET')
A        COPOS           5A
A                                  TEXT('POSTCODE')
A        VILLE          20A
A                                  TEXT('TOWN')
A        CORRES         25A
A                                  TEXT('CORRESPONDENT')
A        REMIS           6S 2
A                                  TEXT('DISCOUNT')
A        MATE            8A
A                                  TEXT('HARDWARE')
A        LANGU           1A
A                                  TEXT('LANGUAGE')
A    K NUCLIE            DESCEND
```