



Contents

PRE-CONVERSION PROCESS OVERVIEW..... 2

Introduction.....2

Pre-Requisites.....3

Building the System Object Model3

APPENDIX I..... 8

Conversion Dictionary.....8

 Global Definitions8

 CV_DICTIONARY9

 CV_PROGRAMS.....10

 CV_PROGRAM_USES12

 CV_PANELS.....13

 CV_PANEL_FIELDS15

 CV_FUNCTIONS17

 CV_DATAVIEWS19

 CV_DATAVIEW_FIELDS20

 CV_COLUMN_USE.....21

Pre-Conversion Tool Tables22

 CHK_ENTITY_TAG22

 CHK-ENTITY-NOTES22

APPENDIX II..... 23

Pre-Conversion Sample Reports23

 Program Resources (By system)23

 Sub-Entity References30

 Dataview Usage (by Dataview).....32

 Un-Referenced Dataviews.....36

Pre-Conversion Process Overview

Introduction

When converting from a CA-IDEAL/CA-DATACOM operating environment to another programming language we have to consider the following issues as factors in the success or failure of any conversion project:

- The m2o conversion environment supports many operating systems and run time environments which differ in both operability and structure to the CA-IDEAL\CA-DATACOM environment.
- Usually there are functional differences which mean that a specific method or process defined in CA-IDEAL will have to be re-engineered in order to function in the same manner in the new environment
- As the new environment uses a different database engine, there will be issues relating to how the converted system emulates the database calls made by the original CA-IDEAL.
- Some of the techniques used when programming in CA-IDEAL create performance problems when converted into another language.
- Most production systems evolve over their lifetime, giving rise to duplicated functionality within the code.
- The structure of the data held within the CA-DATACOM database system may become de-normalized and lead to increased data redundancy
- The destination language environment may well differ in structure considerably from the CA-IDEAL environment and as such it may be necessary to restructure the original system in order to achieve the same functionality as the original system.

Pre-Requisites

In order to perform the pre-conversion analyses the following are required:

- A copy of all CA-IDEAL source code including any other programs which are part of the system that be written in a different language such as Assembler or Cobol.
- An electronic copy of the data dictionary in both BTG and CXX format.
- The specification of the destination environment, which contains details of the destination language, database and operating system.

Building the System Object Model

The first stage in building the system object model is to create a conversion repository, using the data dictionary supplied by the client. This contains definitions of all of the tables, views and indexes, which are currently held within the system to be converted.

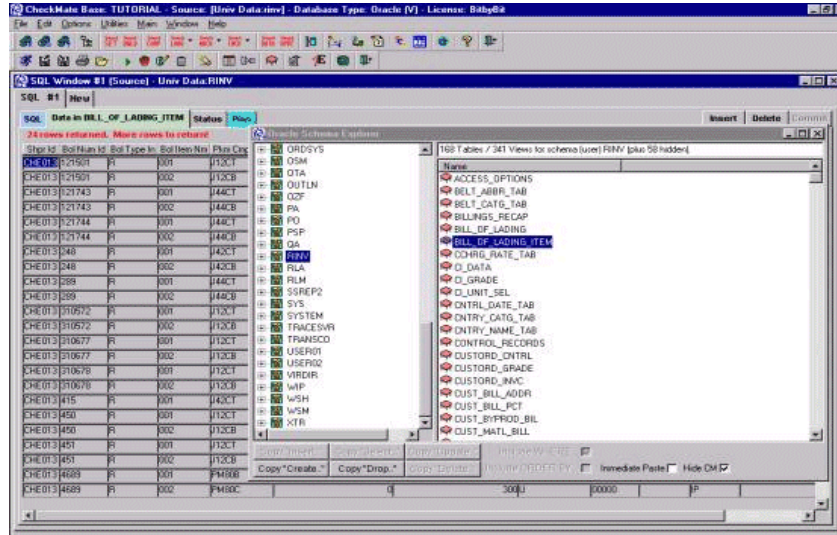
During the process of loading the database definition the integrity of all of the data structures loaded is verified along with any implied relationships that may exist.



www.move2open.com

Pre-Conversion Process Overview

EXAMINING DATA STRUCTURES



This repository also contains detailed information such as field definitions, external references, and database calls, for each source module, which exists within the source code (which has been supplied by the client).



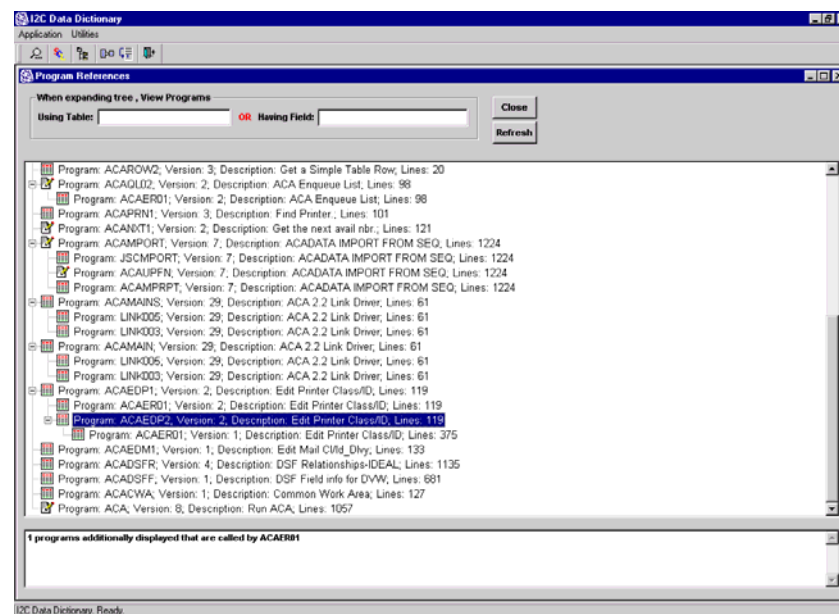


www.move2open.com

Pre-Conversion Process Overview

The next stage in the process of building the system object model involves examining the source code, which has been supplied using the m2o pre-conversion tool. This tool allows the user to identify common objects, normalise any tables or views, and to clean up the code (which the client has supplied).

CHECKING SOURCE USING THE PRE- CONVERSION TOOL



It is possible to navigate through the hierarchy of code object dependencies within the current system model by using the right mouse button, while positioned on the map, to expand the current object. Alternatively use the module tree to navigate through all of the related code objects.

The users can at all times view either the code or all related tables and views again by right clicking on the appropriate code object.

This tool will identify any orphaned programs /subprograms which may not need to be included in the conversion process.





www.move2open.com

Pre-Conversion Process Overview

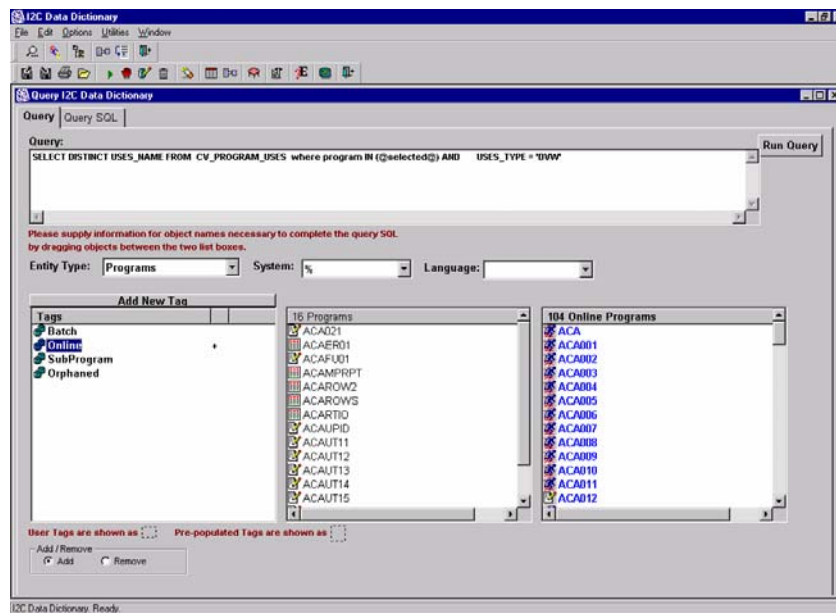
Identify and remove any references to dataviews and/or tables, which are no longer used within the system or a specific code module.

Any subroutines, which are called only from one place, may be included in the invoking program for ease of conversion and future maintenance.

During this process it is possible to normalise the tables/views used within the system using the field tree and dataview map windows that exist within the pre-conversion tool to identify and resolve any data redundancy.

It is also possible to determine which paths are used most frequently to access the various tables or views that exist, and if necessary re-structure them accordingly to enhance performance.

REVIEWING SOURCE ASSOCIATIONS



This tool performs a number of functions as part of the processing: it examines the code and attempts to identify and highlight any redundant code, or code sections, as well as any code that may exist in different modules and may have the same functionality.





www.move2open.com

Pre-Conversion Process Overview

While parsing the code variable type checking is performed and any discrepancies are highlighted so that they may be corrected. Checks are also performed so that any variables, which are not referenced, are also highlighted for removal.

The pre-conversion software also performs validation to ensure that the integrity of data structures being used for inter-program communication are consistent in structure and form between invoking and called modules.

All of the processes performed during this phase are designed to optimise the code and ensure that once converted, the resultant code is both robust and reliable.



Appendix I

Conversion Dictionary

Global Definitions

| | | |
|--------|--------------------------|---|
| SLANG | Source code language | NULL = IDEAL, "C" = COBOL "D" = DATACOM |
| SYSTEM | Unique system identifier | |

The following values are valid in the 'uses' field within this dictionary

| | |
|-----|----------------------------|
| PGM | Program source file |
| PNL | Panel source file |
| RPT | Report source files |
| DVW | View/Table reference |
| FLD | Field reference |
| SEC | Code section reference |
| TAB | Sequential File/Table name |
| ARA | Area reference |
| ELM | Element reference |
| KEY | Key path definition |
| BAS | Database reference |

The SYSTEM column contains a three character value which identifies the subsystem that the specific entity belongs to. This is currently extracted from the header in the ideal source file.



www.move2open.com

Pre-Conversion Process Overview

CV_DICTIONARY

Holds the majority of a converted program's data area in the order it would appear

Prime Key: SYSTEM, PROGRAM, PROGRAM_SEQ

Key 1: SYSTEM, PROGRAM, PROGRAM_SEQ, LEVEL

Key 2: SYSTEM, FIELD, AREA

| | | |
|----------------|--------------|---|
| PROGRAM | VARCHAR2(8) | Program name |
| VERSION | NUMBER(4, 0) | Program version (latest) |
| SYSTEM | VARCHAR2(3) | See:GLOBAL |
| PROGRAM_SEQ | NUMBER(4, 0) | Sequence in program |
| AREA | CHAR(3) | FIL=File area, SQL=SQL Host Vars area, WOR=WORKING-DATA area, PAR=PARAMETER-DATA area |
| USES_COPY_NAME | VARCHAR2(30) | USES-PROGRAM name or USES-DATAVIEW name or USES-PANEL name or USES-REPORT name or COPY DATAVIEW name |
| USES_COPY_TYPE | CHAR(4) | UPGM=USES-PROGRAM UDVW=USES-DATAVIEW UPNL=USES-PANEL URPT=USES-REPORT COPY=COPY statement |
| TABLE_NAME | VARCHAR2(30) | Dataview/file/table name |
| LEVEL | NUMBER(2, 0) | Level 01-49 |
| GROUP | CHAR(1) | Y=group |
| FIELD | VARCHAR2(30) | Field name |
| FIELD_TYPE | CHAR(1) | X=Alphanumeric, V=Varlength alphanumeric, N=Signed numeric, U=Unsigned numeric, D=Date (numeric), F=Flag (true or false), C=Condition (other value) |
| INTERNAL_TYPE | CHAR(1) | Z=Zoned decimal format, P=Packed decimal format, B=Binary format, Space=Same as for Zoned |
| LEFT_DIGITS | NUMBER(5, 0) | Picture left digits |
| RIGHT_DIGITS | NUMBER(5, 0) | Picture right digits |





www.move2open.com

Pre-Conversion Process Overview

| | | |
|----------------|---------------|--|
| PIC_SIZE | NUMBER(10, 0) | Picture left+right digits |
| LEVEL_SIZE | NUMBER(12, 0) | Group/field size |
| LEVEL_MEM_SIZE | NUMBER(12, 0) | Level size in memory |
| OCCURS | NUMBER(5, 0) | Number of occurrences |
| KEY_USAGE | CHAR(1) | P=Primary key, I=index key, Etc, ... |
| REDEF_FIELD | VARCHAR2(30) | If REDEF, field name |
| DEPON_FIELD | VARCHAR2(30) | If DEPON, field name |
| VALUE | VARCHAR2(200) | If has value, the value |
| REDEF_DEPTH | NUMBER(2, 0) | Number of REDEF levels |
| OCCURS_DEPTH | NUMBER(2, 0) | Number of OCCURS levels |
| SLANG | CHAR (1) | Global |

CV_PROGRAMS

Contains all descriptive text and function references contained within a program module.

Prime Key: SYSTEM, PROGRAM

| | | |
|----------------|--------------|--|
| PROGRAM | VARCHAR2(8) | Program name |
| VERSION | NUMBER(4, 0) | Program version (latest) |
| SYSTEM | VARCHAR2(3) | See: Global |
| LINE_COUNT | NUMBER(6, 0) | Number of lines (excl. blank lines & comments) |
| SHORT_DESC | VARCHAR2(58) | Short description comment |
| TEXT1 | VARCHAR2(58) | Comment text line 1 |
| TEXT2 | VARCHAR2(58) | Comment text line 2 |
| TEXT3 | VARCHAR2(58) | Comment text line 3 |
| TEXT4 | VARCHAR2(58) | Comment text line 4 |
| TEXT5 | VARCHAR2(58) | Comment text line 5 |
| TEXT6 | VARCHAR2(58) | Comment text line 6 |
| TEXT7 | VARCHAR2(58) | Comment text line 7 |
| TEXT8 | VARCHAR2(58) | Comment text line 8 |
| TEXT9 | VARCHAR2(58) | Comment text line 9 |
| USES_PROGRAM | NUMBER(3, 0) | Calls in program |
| USES_DATAVIEW | NUMBER(3, 0) | Dataviews in program |
| USES_PANEL | NUMBER(3, 0) | Panels in program |
| USES_REPORT | NUMBER(3, 0) | Reports in program |
| COPY | NUMBER(3, 0) | Number in program |
| WORKING_DATA | CHAR(1) | Y=Has WORKING-DATA area |
| PARAMETER_DATA | CHAR(1) | Y=Has PARAMETER-DATA area |





www.move2open.com

Pre-Conversion Process Overview

| | | |
|---------------------|--------------|---------------------------|
| PROCEDURE_DATA | CHAR(1) | Y=Has PROCEDURE-DATA area |
| PROC_LABEL | NUMBER(4, 0) | Number in program |
| BACKOUT | NUMBER(4, 0) | Number in program |
| CALL | NUMBER(4, 0) | Number in program |
| CHECKPOINT | NUMBER(4, 0) | Number in program |
| DELETE | NUMBER(4, 0) | Number in program |
| EXEC_SQL | NUMBER(4, 0) | Number in program |
| FOR_NEW | NUMBER(4, 0) | Number in program |
| FOR_EACH | NUMBER(4, 0) | Number in program |
| FOR_ANY | NUMBER(4, 0) | Number in program |
| FOR_FIRST | NUMBER(4, 0) | Number in program |
| FOR_NEXT | NUMBER(4, 0) | Number in program |
| LIST | NUMBER(4, 0) | Number in program |
| MOVE_BY_NAME | NUMBER(4, 0) | Number in program |
| MOVE_BY_POSITION | NUMBER(4, 0) | Number in program |
| NOTIFY | NUMBER(4, 0) | Number in program |
| PROCESS_NEXT | NUMBER(4, 0) | Number in program |
| PRODUCE | NUMBER(4, 0) | Number in program |
| QUIT | NUMBER(4, 0) | Number in program |
| QUIT_PROGRAM | NUMBER(4, 0) | Number in program |
| QUIT_RUN | NUMBER(4, 0) | Number in program |
| QUIT_ALL_TRANSFER | NUMBER(4, 0) | Number in program |
| QUIT_LABEL | NUMBER(4, 0) | Number in program |
| QUIT_LABEL_TRANSFER | NUMBER(4, 0) | Number in program |
| QUIT_PROC | NUMBER(4, 0) | Number in program |
| QUIT_PROC_TRANSFER | NUMBER(4, 0) | Number in program |
| REFRESH | NUMBER(4, 0) | Number in program |
| RELEASE_PROG | NUMBER(4, 0) | Number in program |
| RELEASE_PANEL | NUMBER(4, 0) | Number in program |
| RELEASE_REPORT | NUMBER(4, 0) | Number in program |
| RESET | NUMBER(4, 0) | Number in program |
| SET_BY_NAME | NUMBER(4, 0) | Number in program |
| SET_BY_POSITION | NUMBER(4, 0) | Number in program |
| TRANSMIT | NUMBER(4, 0) | Number in program |
| SLANG | CHAR(1) | Global |





www.move2open.com

Pre-Conversion Process Overview

CV_PROGRAM_USES

This table contains a cross reference of object references within a specific level.

Prime Key: SYSTEM, PROGRAM,USES_TYPE, USES_NAME

| | | |
|--------------|--------------|---|
| SYSTEM | VARCHAR2(3) | Global system name |
| PROGRAM | VARCHAR2(8) | Object name (program) |
| USES_TYPE | CHAR(3) | See: Global |
| USES_NAME | VARCHAR2(32) | Object name referenced |
| USES_VERSION | NUMBER(4, 0) | OBSOLETE |
| USES_SYSTEM | CHAR(3) | Referenced objects system. If blank the same as SYSTEM. |
| TOP_LEVEL | CHAR(1) | "Y" = Not referenced by any other objects |
| DOES_INSERT | CHAR(1) | Inserts "Y" or "N" |
| DOES_UPDATE | CHAR(1) | Updates "Y" or "N" |
| DOES_DELETE | CHAR(1) | Deletes "Y" or "N" |
| SLANG | CHAR(1) | Global |

Only For Program source

When TOP_LEVEL is 'Y' this is the first program in calling sequence.
When USES-NAME is spaces this is the last entry in a program calling sequence.



CV_PANELS

Panel header information as defined in the panel definitions

Prime Key: SYSTEM, PANEL

Child: CV_PANEL_FIELDS, CV_FUNCTIONS

| | | |
|----------------|--------------|---|
| PANEL | VARCHAR2(8) | Panel Name |
| VERSION | NUMBER(4, 0) | Panel version (latest) |
| SYSTEM | VARCHAR2(3) | See: Global |
| LINE_COUNT | NUMBER(6, 0) | Number of lines |
| FIELDS | NUMBER(3, 0) | Number of fields |
| FIELD_LITERALS | NUMBER(3, 0) | Number of literals |
| OCCUR_TYPE | CHAR(1) | Value=fixed occurs, *=expanding occurs |
| OCCURS | NUMBER(5, 0) | Fixed occurs=Number otherwise spaces |
| LAST_ROW | NUMBER(3, 0) | Last row of panel |
| LAST_COLUMN | NUMBER(3, 0) | Last column of panel |
| HELP_PANEL | VARCHAR2(8) | Help panel name (opt) |
| PREFIX_PANEL | VARCHAR2(8) | Prefix panel name (opt) |
| SUFFIX_PANEL | VARCHAR2(8) | Suffix panel name (opt) |
| STATUS | VARCHAR2(4) | Should be PROD |
| DATE_CREATED | NUMBER(6, 0) | Informational only |
| TIME_CREATED | NUMBER(4, 0) | Informational only |
| CREATED_BY | VARCHAR2(20) | Informational only |
| DATE_MODIFIED | NUMBER(6, 0) | Informational only |
| TIME_MODIFIED | NUMBER(4, 0) | Informational only |
| MODIFIED_BY | VARCHAR2(20) | Informational only |
| RUN_STATUS | VARCHAR2(10) | Informational only |
| SHORT_DESC | VARCHAR2(60) | Comment entry |
| WIDTH | NUMBER(3, 0) | 80-240 columns |
| LARGE_PANEL | CHAR(1) | Y=large panel |
| INFILL | CHAR(1) | Global input fill char: _=underscores, Z=zeros, Low values=low values |
| OUTFILL | CHAR(1) | Global output fill char: _=underscores, Z=zeros, Low values=low values, Other = itself |
| NONDISPL | CHAR(1) | Global non-display |



www.move2open.com

Pre-Conversion Process Overview

| | | |
|-------------------|---------|---|
| | | char |
| ERRORFIL | CHAR(1) | Global error fill char: Y or N |
| CASETRAN | CHAR(1) | Global Case translation: U=Upper case chars M=Mixed case chars |
| REQUIRED | CHAR(1) | Global Input required: Y or N |
| ERRHANDL | CHAR(1) | Global Error handling flag: Y or N |
| PF1PF3 | CHAR(1) | Global PF1/PF3 keys help/clarify keys: Y or N |
| PF781011 | CHAR(1) | Global PF7/PF8/PF10/PF11 scroll keys: Y, Z, N |
| ALLOW_EOF | CHAR(1) | Global allow end-of- field char: Y or N |
| EDIT_RULE_PROCESS | CHAR(1) | Global edit rule process: A=Application, C=Conditional |
| PROCESS_ON_SCROLL | CHAR(1) | Process on scroll: Y or N |
| PTYPE | CHAR(1) | Source type NULL = IDEAL "X" = CICS |



www.move2open.com

Pre-Conversion Process Overview

CV_PANEL_FIELDS

Holds panel field and value information

Prime Key: SYSTEM, PANEL, SEQ

Parent: CV_PANEL

| | | |
|------------|---------------|---|
| PANEL | VARCHAR2(8) | Panel name |
| SYSTEM | VARCHAR2(3) | See: Global |
| SEQ | NUMBER(4, 0) | Field sequence number |
| FIELD | VARCHAR2(30) | Field name |
| TYPE | CHAR(1) | Field type: G=Group (only 1 allowed), X=Alphanumeric, N=Numeric |
| LEVEL | NUMBER(2, 0) | Level number: 2=non-occurring, 3=occurring |
| ROW | NUMBER(3, 0) | Row number: 1-999 |
| COLUMN | NUMBER(3, 0) | Column number: 1-240 |
| LENGTH | NUMBER(3, 0) | Field/literal length: 1-240 |
| OCCUR_TYPE | CHAR(1) | Occurrence type: Value=Fixed occur, *=Expanding occur |
| OCCUR | NUMBER(5, 0) | Fixed occur=value, Expanding occur=blank |
| VALUE | VARCHAR2(200) | Alphanumeric value |
| ATTRIBUTES | VARCHAR2(14) | Combinations of: A=Alphanumeric, N=Numeric, P=Protected, U=Unprotected (default), S=Skip, I=Invisible, H=Highlight, L=Lowlight (default), E=Ensure received, C=Cursor |
| COLOR | CHAR(1) | N=Neutral, B=Blue, G=Green, T=Turquoise, R=Red, P=Pink, Y=Yellow, W=White |





www.move2open.com

Pre-Conversion Process Overview

| | | |
|-----------------|--------------|---|
| EXTENDED_HILITE | CHAR(1) | U=Underscores, B=Blinking, R=Reverse video N=None |
| JUSTIFY | CHAR(1) | L=Left justify R=Right justify, A=Align by decimal point |
| MUSTFILL | CHAR(1) | Y=Field all be typed into or must be empty |
| NULLABLE | CHAR(1) | Y=Nullable |
| INFILL | CHAR(1) | Input fill char: _=Underscores, Z=Zeros, Low value=Low values |
| OUTFILL | CHAR(1) | Output fill char: _=Underscores, Z=Zeros, Low value=Low values Other=Itself |
| NONDISPL | CHAR(1) | Non display char |
| ERRORFIL | CHAR(1) | Error fill char |
| CASETRAN | CHAR(1) | Case translation: U=Upper case chars, M=Mixed case chars |
| REQUIRED | CHAR(1) | Field input is required: Y or N |
| ERRHANDL | CHAR(1) | Error handling flag: Y or N |
| ALLOW_EOF | CHAR(1) | Allow end-of-field char: Y or N |
| IN | NUMBER(3, 0) | Numeric fields: Number of left digits allowed |
| DP | NUMBER(3, 0) | Numeric fields: Number of right digits allowed |
| MIN_REQ_DEC | NUMBER(3, 0) | Numeric fields: Minimum required decimals allowed |
| EDIT_PATTERN | VARCHAR2(30) | Numeric fields: Edit pattern (refer to Ideal manual) |
| DIGIT_SEP | CHAR(1) | Numeric fields: Comma separator allowed |
| MINUS_SIGN | CHAR(1) | Numeric fields: Minus sign allowed |
| CURRENCY | CHAR(1) | Numeric fields: |





www.move2open.com

Pre-Conversion Process Overview

| | | |
|-------------|--------------|--|
| | | Currency symbol allowed |
| CHECK_DIGIT | CHAR(1) | Numeric fields: T=Modulus 10 or E=Modulus 11 check digit validation |
| MIN_RANGE | VARCHAR2(20) | Minimum range value |
| MAX_RANGE | VARCHAR2(20) | Maximum range value |
| OUTZERO | CHAR(1) | Obsolete |

CV_FUNCTIONS

Panel function definitions and information

Prime Key: SYSTEM, PROGRAM, SEQ

Key 1: PANEL, FIELD

Parent: CV_PANELS

| | | |
|--------------------|--------------|--|
| SYSTEM | VARCHAR2(3) | See: Global |
| PROGRAM | VARCHAR2(8) | Program containing panel |
| SEQ | NUMBER(6, 0) | Sequence number |
| REFRESH | CHAR(1) | Y=Relates to REFRESH statement |
| TRANSMIT | CHAR(1) | Y=Relates to TRANSMIT statement |
| RESET | CHAR(1) | Y=Relates to RESET statement |
| USED_FIELD | CHAR(1) | Y=Panel field is used in this function |
| EMPTY | CHAR(1) | Y=\$EMPTY() is used |
| RECEIVED | CHAR(1) | Y=\$RECEIVED() is used |
| PANEL_ERROR | CHAR(1) | Y=\$PANEL_ERROR is used |
| PANEL_FIELD_ERROR | CHAR(1) | Y=\$PANEL_FIELD_ERROR is used |
| PANEL_GROUP_OCCURS | CHAR(1) | Y=\$PANEL_GROUP_OCCURS is used |
| ATTR_NA | CHAR(1) | Y=Numeric/Alphanumeric attribute is used |
| ATTR_HL | CHAR(1) | Y=Highlight/Lowlight attribute is used |
| ATTR_PU | CHAR(1) | Y=Protected/Unprotected attributes is used |
| ATTR_S | CHAR(1) | Y=Skip attribute is used |
| ATTR_I | CHAR(1) | Y=Invisible attributes is used |
| ATTR_E | CHAR(1) | Y=Ensure Received attribute is used |





www.move2open.com

Pre-Conversion Process Overview

| | | |
|------------|---------------|--|
| ATTR_C | CHAR(1) | Y=Cursor attribute is used |
| COLOR | CHAR(1) | N=Neutral, B=Blue, G=Green, T=Turquoise, R=Red, P=Pink, Y=Yellow, W=White |
| XHIGHLIGHT | CHAR(1) | U=Underscores, B=Blinking, R=Reverse video N=None |
| TEMP | CHAR(1) | T=Temporary attributes or P=Permanent attributes |
| KEY | CHAR(1) | Y=\$KEY function is used |
| ENTER_KEY | CHAR(1) | Y=\$ENTER_KEY function is used |
| PF1 | CHAR(1) | Y=\$PF1 function is used |
| PF2 | CHAR(1) | Y=\$PF2 function is used |
| PF3 | CHAR(1) | Y=\$PF3 function is used |
| PF4 | CHAR(1) | Y=\$PF4 function is used |
| PF5 | CHAR(1) | Y=\$PF5 function is used |
| PF6 | CHAR(1) | Y=\$PF6 function is used |
| PF7 | CHAR(1) | Y=\$PF7 function is used |
| PF8 | CHAR(1) | Y=\$PF8 function is used |
| PF9 | CHAR(1) | Y=\$PF9 function is used |
| PF10 | CHAR(1) | Y=\$PF10 function is used |
| PF11 | CHAR(1) | Y=\$PF11 function is used |
| PF12 | CHAR(1) | Y=\$PF12 function is used |
| PF13 | CHAR(1) | Y=\$PF13 function is used |
| PF14 | CHAR(1) | Y=\$PF14 function is used |
| PF15 | CHAR(1) | Y=\$PF15 function is used |
| PF16 | CHAR(1) | Y=\$PF16 function is used |
| PF17 | CHAR(1) | Y=\$PF17 function is used |
| PF18 | CHAR(1) | Y=\$PF18 function is used |
| PF19 | CHAR(1) | Y=\$PF19 function is used |
| PF20 | CHAR(1) | Y=\$PF20 function is used |
| PF21 | CHAR(1) | Y=\$PF21 function is used |
| PF22 | CHAR(1) | Y=\$PF22 function is used |
| PF23 | CHAR(1) | Y=\$PF23 function is used |
| PF24 | CHAR(1) | Y=\$PF24 function is used |
| PANEL | VARCHAR2 (8) | Panel in function |
| FIELD | VARCHAR2 (30) | Panel field in function |





www.move2open.com

Pre-Conversion Process Overview

CV_DATAVIEWS

Holds dataview header information from a project's print and/or sequential dataview files

Prime Key: DATAVIEW
Key 1: DATA_MEDIUM

Child: CV_DATAVIEWS_FIELDS

| | | |
|-------------|---------------|---|
| DATAVIEW | VARCHAR2(30) | Dataview name |
| TABLE_NAME | VARCHAR2(30) | Base table name |
| SEQUENCE | NUMBER(10, 0) | Sequence number |
| VERSION | NUMBER(4, 0) | Version number |
| STATUS | VARCHAR2(15) | Should be PROD |
| DATE | NUMBER(8, 0) | Informational only |
| TIME | NUMBER(6, 0) | Informational only |
| CAT_DATE | NUMBER(6, 0) | Informational only |
| CAT_TIME | NUMBER(4, 0) | Informational only |
| DATA_MEDIUM | VARCHAR2(10) | DATA COM/DB = Database view/table SEQUENTIAL = Flat file COBCOPY = Cobol copyfile |
| UPDATEABLE | CHAR(1) | Y=Updateable |
| DBID | NUMBER(6, 0) | The database id |
| FILENAME | VARCHAR2(30) | Sequential Only: The file-name |
| DEVICE | VARCHAR2(12) | Informational only |
| RECSIZE | NUMBER(4, 0) | Informational only |
| BLKSIZE | NUMBER(4, 0) | Informational only |
| FROM_LINE | NUMBER(6, 0) | Start line number in print/sequential dataview file |
| NUM_FIELDS | NUMBER(4, 0) | The number of fields in the view |



