CA IDMS - 19.0
Using CA IDMS Extractor

Date: 10-Jan-2018
This Documentation, which includes embedded help systems and electronically distributed materials, (hereinafter referred to as the "Documentation") is for your informational purposes only and is subject to change or withdrawal by CA at any time. This Documentation is proprietary information of CA and may not be copied, transferred, reproduced, disclosed, modified or duplicated, in whole or in part, without the prior written consent of CA.

If you are a licensed user of the software product(s) addressed in the Documentation, you may print or otherwise make available a reasonable number of copies of the Documentation for internal use by you and your employees in connection with that software, provided that all CA copyright notices and legends are affixed to each reproduced copy.

The right to print or otherwise make available copies of the Documentation is limited to the period during which the applicable license for such software remains in full force and effect. Should the license terminate for any reason, it is your responsibility to certify in writing to CA that all copies and partial copies of the Documentation have been returned to CA or destroyed.

TO THE EXTENT PERMITTED BY APPLICABLE LAW, CA PROVIDES THIS DOCUMENTATION “AS IS” WITHOUT WARRANTY OF ANY KIND, INCLUDING WITHOUT LIMITATION, ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NONINFRINGEMENT. IN NO EVENT WILL CA BE LIABLE TO YOU OR ANY THIRD PARTY FOR ANY LOSS OR DAMAGE, DIRECT OR INDIRECT, FROM THE USE OF THIS DOCUMENTATION, INCLUDING WITHOUT LIMITATION, LOST PROFITS, LOST INVESTMENT, BUSINESS INTERRUPTION, GOODWILL, OR LOST DATA, EVEN IF CA IS EXPRESSLY ADVISED IN ADVANCE OF THE POSSIBILITY OF SUCH LOSS OR DAMAGE.

The use of any software product referenced in the Documentation is governed by the applicable license agreement and such license agreement is not modified in any way by the terms of this notice.

The manufacturer of this Documentation is CA.

Provided with “Restricted Rights.” Use, duplication or disclosure by the United States Government is subject to the restrictions set forth in FAR Sections 12.212, 52.227-14, and 52.227-19(c)(1) - (2) and DFARS Section 252.227-7014(b)(3), as applicable, or their successors.

Copyright © 2017 CA. All rights reserved. All trademarks, trade names, service marks, and logos referenced herein belong to their respective companies.
Table of Contents

IDMS Extractor Information ................................................................. 12
Flexible Online Sorting with CA IDMS/DC Sort .................................. 12
Easy Selection of Criteria ................................................................... 13
Multiple Sorts at One Terminal ......................................................... 13
Multiple Sort Keys in Each Sequence ................................................ 13
Optional Online Criteria at Runtime ................................................. 14
Quick and Easy Sorts ....................................................................... 14
CA IDMS Dictionary Access ............................................................. 15
Preprocessor Support ....................................................................... 15
Flexible Retrieval of Sorted Data ...................................................... 15
Session Kept Active Across Pseudo Converses ................................ 15
Processing Environment .................................................................... 15
  System Output .................................................................................. 16
    Reports by CA IDMS Log Analyzer ............................................... 16
    About CA IDMS Log Analyzer Billing Reports ................................. 17
    Overview of Billing Reports .......................................................... 18
    About CA IDMS Log Analyzer Program Reports ............................... 31
    Overview Of Program Reports ...................................................... 32
    About CA IDMS Log Analyzer Management Reports ...................... 41
Journal Reports ................................................................................ 60
  Activity Report ................................................................................ 60
  Program Reports ............................................................................. 61
  Special Reports ............................................................................... 61
  Management Reports ...................................................................... 61
Journal Displays ................................................................................ 61
  Record Display ............................................................................... 61
  Database Key Display .................................................................... 61
  Program Display ............................................................................. 62
Audit Report ....................................................................................... 62
Control Parameters .......................................................................... 62
  SUPPRESS Parameter .................................................................... 62
  REPORT Parameter ........................................................................ 63
  DISPLAY Parameter ....................................................................... 63
  DLIMITS Parameter ....................................................................... 63
  DSUPPS Parameter ........................................................................ 63
Entering Commands ......................................................................................................................... 80
  Scroll Options ........................................................................................................................ 80
  Primary Commands ................................................................................................................ 81
  Line Commands ..................................................................................................................... 81
  Conventions, Rules and Syntax ............................................................................................... 81
  Functions ............................................................................................................................... 82
  Command Syntax Summary ...................................................................................................... 83

Entering Commands ....................................................................................................................... 84
  Line Commands 1 .................................................................................................................... 85
    Entering Line Commands .................................................................................................... 86
    B (before) and A (after) Commands .................................................................................... 86
    C (copy) Command ............................................................................................................. 87
    M (move) Command ........................................................................................................... 88
    R (repeat) Command ......................................................................................................... 89
    TABS Command 1 ............................................................................................................... 90
    D (delete) Command ......................................................................................................... 90
    X (exclude) Command ...................................................................................................... 91
    I (insert) Command ........................................................................................................... 92
    COLS (columns) Command ............................................................................................... 92
    A (after) Command ........................................................................................................... 92
    B (before) Command ........................................................................................................ 92
    BNDS (bounds) Command ................................................................................................. 92

Primary Commands 1 .................................................................................................................... 93
  UP and DOWN Commands ..................................................................................................... 93
  RIGHT and LEFT Commands ............................................................................................... 93
  LOCATE Command ............................................................................................................. 94
  MEMORY Command ............................................................................................................ 95
  FIND and RFIND Commands ............................................................................................. 95
  CHANGE and RCHANGE Commands .................................................................................. 96
  RESET Command ............................................................................................................... 98
  RESHOW Command ........................................................................................................... 98
  RFIND Command ................................................................................................................ 98
  RIGHT Command ................................................................................................................ 99
  SAVE Command .................................................................................................................. 99
  TABB Command .................................................................................................................. 99
  TABF Command .................................................................................................................. 99
  TABS Command .................................................................................................................. 99
  TIME Command .................................................................................................................. 99
  TOP Command .................................................................................................................... 100
  UP Command ...................................................................................................................... 100
  CANCEL Command ............................................................................................................ 100
  END/RETRY Command ....................................................................................................... 100
Field Level Selection Criteria Screen ................................................................. 139
Field-Level Selection Criteria Screen ............................................................... 139
Record/Element Review Screen ........................................................................ 140
Example ........................................................................................................... 140
Record/Element Review Screen 1 ..................................................................... 140
Record/Element Display & Modification Commands .......................................... 141
DISPLAY Command ..................................................................................... 141
Rule for the DISPLAY Command ................................................................. 141
EXIT Command ............................................................................................ 141
INITIALIZE Command .................................................................................. 141
QUIT Command ............................................................................................. 141
SET AUTOHEX Command ............................................................................. 142
SET HEX/NATIVE Command ........................................................................ 142
SET LOWERCASE Command ....................................................................... 142
Field Level Criteria Deselection Screen .......................................................... 142
Field Level Criteria Deselection Screen 1 ....................................................... 143
Confirm Subschema Record Date Screen ....................................................... 143
Confirm Subschema Record Date Screen 1 .................................................... 143
Field Mismatch Screen .................................................................................. 144
Field Mismatch Screen 1 .............................................................................. 144
Utilities Menu Screen .................................................................................. 145
Deleting, Printing, or Renaming Selection Criteria Specifications .................. 145
Deleting, Printing, or Renaming JCL Members ............................................. 145
Copying Selection Criteria Specifications or JCL Members ......................... 145
Utilities Menu Info ....................................................................................... 145
Specification Utilities Screen .......................................................................... 145
Deleting a Selection Specification .................................................................. 145
Renaming a Selection Criteria Specification ................................................. 146
Printing a Selection Criteria Specification .................................................... 146
Specification Utilities Screen 1 .................................................................... 146
JCL Utilities Screen ....................................................................................... 146
Deleting a JCL Member .................................................................................. 146
Renaming a JCL Member ............................................................................... 147
Printing a JCL Member ................................................................................ 147
JCL Utilities Screen 1 .................................................................................. 147
Copy Utility Screen ....................................................................................... 147
Copying a Selection Criteria Specification ................................................... 147
Copying a JCL Member ............................................................................... 148
Copy Utility Screen Info ............................................................................... 148

Operations ...................................................................................................... 148
CA IDMS Extractor Requirements .................................................................................................................. 149
System Requirements ........................................................................................................................................ 149
  CA CIS Requirements for CA IDMS ................................................................................................................ 149
  CA IDMS System Requirements ...................................................................................................................... 150
Environments Supported ................................................................................................................................. 151
Minidisk Space Requirements .......................................................................................................................... 151
Operating Environment ..................................................................................................................................... 154
Terminal Type ..................................................................................................................................................... 155
Security ............................................................................................................................................................. 155
Online Storage Requirements ............................................................................................................................ 155
  Program Storage ................................................................................................................................................ 155
  Working Storage ............................................................................................................................................... 155
  Scratch Storage ................................................................................................................................................. 155
  Queue Storage .................................................................................................................................................. 155
Batch Storage Requirements ............................................................................................................................. 156
  Batch Storage Requirements--EXTRACT Step .................................................................................................. 156
  Batch Storage Requirements--LOAD Step ...................................................................................................... 156
CA IDMS Journal Analyzer Control Statements .............................................................................................. 157
z/OS Operation .................................................................................................................................................. 157
  Execution JCL (z/OS) ..................................................................................................................................... 157
z/VSE Operation ............................................................................................................................................... 160
  z/VSE File Assignments ................................................................................................................................. 161
  Execution JCL (z/VSE) ................................................................................................................................. 161
Sample z/VM EXEC--USVEXEC .......................................................................................................................... 164
Recovery Procedure .......................................................................................................................................... 167
Executing CA IDMS Extractor ............................................................................................................................. 167
  Sample JCL ..................................................................................................................................................... 168
    Extract Information from the Source Dictionary - z/VSE ............................................................................ 168
PROCESS Parameter ......................................................................................................................................... 204
  Extracts ......................................................................................................................................................... 205
  ALL .............................................................................................................................................................. 205
  Displays ......................................................................................................................................................... 205
  PROCESS Parameter Keywords .................................................................................................................... 205
Performing the EXTRACT and LOAD Steps--USVEXEC ................................................................................... 207
  z/OS JCL ....................................................................................................................................................... 210
    Running in Local Mode Under z/OS ............................................................................................................. 212
  z/VSE JCL ..................................................................................................................................................... 212
    Running in Local Mode Under z/VSE ........................................................................................................ 215
  z/VM EXEC .................................................................................................................................................. 216
    Running in Local Mode Under z/VM ......................................................................................................... 222
Printing JCL--USVPJCL ..................................................................................................................................... 222
Using CA IDMS Extractor

This chapter provides the information you need to significantly speed up the testing and maintenance phases of applications development by using CA IDMS Extractor. In addition, the many features that CA IDMS Extractor offers are documented.

CA IDMS Extractor is an application development tool that significantly speeds up the testing and maintenance phases of applications development by reducing or eliminating the need to develop special test database load programs.

To use CA IDMS Extractor, three basic steps are necessary:

1. Choose an appropriate CA IDMS database as input (that is, the source database).
2. Use the Selection Criteria Specification Component of CA IDMS Extractor to describe the records of the source database that you want extracted from the source database and loaded to the target database.
3. Tell CA IDMS Extractor to extract the selected records from the source database and load them to the target database. The source database remains unchanged.

This section introduces CA IDMS Extractor and describes the following components and features:
- IDMS Extractor Information (see page 12)
- Concepts (see page 71)
- JCL Editing Commands (see page 80)
- Online Session (see page 114)
- Operations (see page 148)

IDMS Extractor Information

Flexible Online Sorting with CA IDMS/DC Sort

CA IDMS/DC Sort is a fast and efficient online sort utility for both the CA IDMS/DC Sort and CICS environments. It provides a versatile means to sort any information online, regardless of the file structure or original sequence.

CA IDMS/DC Sort eliminates the need to design and maintain specific sequences in your files or on your database. In addition, fewer separate sorting and reporting jobs need to be maintained at your site. Therefore, the use of CA IDMS/DC Sort reduces design and maintenance time and improves performance.
Easy Selection of Criteria

For each sort to be performed, you can select criteria easily, either before or during execution of a program. To specify the criteria before execution, you enter the sort criteria within the application program. As an alternative, in the program you can specify that each user define necessary sort criteria whenever a sort is required during execution.

Multiple Sorts at One Terminal

CA IDMS/DC Sort accommodates complex sequence requests. Within an application it can sort as many as 10 different data structures concurrently. It can also sort a single data structure into many different sequences.

For more information, see the following topics:
- Multiple Sort Keys in Each Sequence (see page 13)
- Optional Online Criteria at Runtime (see page 14)
- Quick and Easy Sorts (see page 14)
- CA IDMS Dictionary Access (see page 15)
- Preprocessor Support (see page 15)
- Flexible Retrieval of Sorted Data (see page 15)
- Session Kept Active Across Pseudo Converses (see page 15)
- Processing Environment (see page 15)
- Control Parameters (see page 62)
- CA IDMS Presspack Decompression Support (see page 66)
- Need For CA IDMS Journal Analyzer (see page 66)
- What Is CA IDMS Journal Analyzer? (see page 67)
- CA IDMS Extractor Components (see page 67)

Multiple Sort Keys in Each Sequence

When specifying a sort of a particular data structure, you can specify up to 16 elements as sort keys. These elements can be:

- Selected and sequenced when setting up the sort in the program or
- Specified by the user when a program is run.
Optional Online Criteria at Runtime

In the application program, the system designer can set a user option (USER) that allows the user to specify sort criteria at processing time.

When USER is specified in the program, at processing time CA IDMS/DC Sort displays a screen for defining sort criteria. At this time the user can select the sequence of key elements and specify whether each element is to be sorted in ascending or descending order.

Quick and Easy Sorts

CA IDMS/DC Sort makes it possible to respond quickly to a request for a new sequence of data.

- Sort sequences can be specified immediately online.
- There is no need to spend extra time restructuring either a file, a database, or an index.
- Permanent changes to files or to the database are not necessary.

CA IDMS/DC Sort increases flexibility.

- Future sort requirements are taken care of. If the USER function is specified, the user can even specify sequence and sort order at runtime.
- Easy-to-use parameter statements allow ordering of data from one file in many different sequences.

CA IDMS/DC Sort reduces time spent by program designers, because they do not have to decide in advance which is the best sequence to store the data; how many sequences are needed to support processing requirements; which sequences are most important; or whether the sequence will meet future processing requirements. Each program uses whatever sequence is necessary, no matter how the data is stored.

CA IDMS/DC Sort reduces time spent reorganizing and restructuring.

- It is not necessary to reorganize a data structure when a new sort sequence is proposed.

CA IDMS/DC Sort eliminates the need to create and maintain redundant files, databases, or indexes simply to satisfy new sort requests.

- System overhead is not required to maintain a variety of sorted sequences.
- Personnel time is not required to create and maintain programs to keep redundant data synchronized.
- Additional DASD storage space is not required.
CA IDMS Dictionary Access

If your system operates under CA IDMS, CA IDMS/DC Sort allows you to specify a CA IDMS dictionary. Use of the dictionary eliminates the need to specify various parameters in the SETSORT statement, since CA IDMS/DC Sort can extract those values from the dictionary.

If you are specifying the sort criteria at runtime, the elements extracted from the dictionary are automatically displayed by CA IDMS/DC Sort on the Sort Selection Display, where you can easily select the sequence and sort order. See sequence and sort order. See Chapter 4, Examples for more information on the display.

Preprocessor Support

CA IDMS/DC Sort provides a preprocessor for use with COBOL, Assembler, PLI, and CA ADS. The preprocessor uses the CA IDMS/DC Sort parameter statements to generate programming statements that fit your sorting requirements. The preprocessor also identifies errors.

Flexible Retrieval of Sorted Data

CA IDMS/DC Sort offers several alternatives for retrieving sorted records. These alternatives are specified in GETSORT parameter statements. The sorted records can be retrieved:

- From the beginning of the list of the sorted records (FIRST).
- From the end of the list of the sorted records (LAST).
- By moving forward (NEXT) or backward (PRIOR) within the list of sorted records.

Multiple GETSORT statements with different retrieval parameters can be issued. For example, CA IDMS/DC Sort may have sorted a list of salespersons in descending order by sales. By specifying FIRST and NEXT in GETSORT statements, you can obtain the top five from the list. Similarly, by specifying LAST and PRIOR, you can obtain the lowest five from the same ordered list.

Session Kept Active Across Pseudo Converses

CA IDMS/DC Sort automatically keeps track of your location within the sorted file. For example, if the sorted file contains more than one screenful of records, CA IDMS/DC Sort allows you to move from screen to screen without additional programming.

Processing Environment

CA IDMS Journal Analyzer supports CA IDMS Release 19.0.00.
System Output

CA IDMS Journal Analyzer produces three classes of printed system output for user analysis:

- Journal Reports -- provide statistical information that give detailed and/or summarized views of the database activity.
- Journal Displays -- provide a detailed view of specific database changes.
- Audit Report -- provides a listing of the messages generated by CA IDMS Journal Analyzer.

Reports by CA IDMS Log Analyzer

CA IDMS Log Analyzer provides a variety of Log Reports using information taken from the CA IDMS Log. Log Reports can be broken down into three categories: Billing Reports, Program Reports, and Management Reports.

- Billing Reports (see page 16)
- Program Reports (see page 16)
- Management Reports (see page 17)
- Audit Report (see page 17)

Billing Reports

Billing Reports are available at four levels of detail. CA IDMS Log Analyzer uses the same information to produce four different billing reports. The reports are selected and sequenced on fields that are meaningful in a billing/chargeback system. User IDs are given for CICS, CA IDMS/DC, and CA ADS. Batch account billing information is provided for all batch programs.

The four Billing Reports are:

- Billing Details Report
- Billing Summary Report
- Billing System Summary Report
- Billing Grand Summary Report

Program Reports

Program Reports (similar to the Billing Reports) are available at four levels of detail. Program Reports are selected and sequenced by the name of the program that generates the run-unit activity. CA IDMS Log Analyzer uses the same information to produce four different program reports.

They include:

- Program Details Report
- Program Summary Report
- Program System Summary Report
Management Reports

There are three types of Management Highlights/Summary Reports:

- Highlights Program Summary Report
- Highlights System Summary Report
- Highlights Grand Summary Report

CA IDMS Log Analyzer also produces two other Management Reports:

- Highlights/Buffer Pool Utilization Report
- Ranking Report

Audit Report

The Audit Report lists all parameters input and processed, and also present a list of all messages that were generated during execution.

About CA IDMS Log Analyzer Billing Reports

Contents

- Tying Run-Unit Activity to an ID and a time (see page 17)
- Four Reports--One Set of CA IDMS Log Statistics (see page 18)
- Hierarchical Nature of Reports (see page 18)
- Customizing Billing Reports (see page 18)

The Billing Reports use information from the CA IDMS Log to produce both detailed and summarized report statistics. Billing Reports can serve as a functional model for building an effective billing system in your environment. You will get this report by specifying REPORT = BILLING on the parameter statement.

Tying Run-Unit Activity to an ID and a time

Depending on the parameters you choose, run-unit activity can be tied to a specific user, transaction, terminal, or account number. For CA IDMS/DC run-units, the data reported under the headings OPER-ID, TERM-ID, or TRANS-ID comes from the log record. (For CICS, this information is taken from the External Request Element (ERE) extension as it is built by the CA IDMS Log Analyzer version of the CA IDMS SVC exit routine. The data reported as ACCOUNT for batch jobs also originates from the ERE extension.)

In addition, the Billing Reports present this information within the framework of the time interval you select. Run-unit totals are shown; they are also reflected as a ratio of the system totals (i.e., the percentage of all CA IDMS resources consumed during the specified time interval).
Four Reports--One Set of CA IDMS Log Statistics

Physically, there are four Billing Reports (and the Billing Record File) to choose from. It is important to understand, however, that each report is produced from the statistics that are found on the CA IDMS Log. Statistics in the reports are presented in various formats and at different levels of summarization. The Billing Report is available at the detail, summary, system summary, and grand summary level. The grand summary is produced automatically if you choose to have information reported for multiple time intervals.

Hierarchical Nature of Reports

Billing Reports are produced on a hierarchical level: if you ask for the lowest level report (LEVEL = DETAIL), you will also receive the higher-level reports. These would include the Billing Summary Report, which summarizes the data of the Billing Details Report (LEVEL = SUMMARY), and also the Billing System Summary Report (LEVEL = SYSTEM). The Billing Grand Summary Report is controlled by the INTERVAL parameter.

Customizing Billing Reports

While the Billing File and the Billing Reports are functional models, CA realizes that users will want to tailor CA IDMS Log Analyzer billing information and use it in different ways. Source code for the Billing File and Billing Reports is included with CA IDMS Log Analyzer. Sectionlines for users who want to customize the billing data appear in customizing the Billing Reports. Instructions for printing source code are included in USLBILX and USLRPT5 Source Code.

Overview of Billing Reports

Contents
- Billing Details Report (see page 19)
  - Billing Details Report Fields (see page 23)
- Billing Summary Report (see page 24)
  - Billing Summary Report Fields (see page 25)
- Billing System Summary Report (see page 26)
  - Billing System Summary Report Fields (see page 27)
- Billing Grand Summary Report (see page 29)
  - Billing Grand Summary Report Fields (see page 30)

Billing Details Report-presents detailed information for each run-unit, reported in termination time sequence. Depending on the parameter combination selected, this report allows you to identify run-unit activity by account number, terminal ID, operator ID, or transaction ID.

Billing Summary Report-records the sum of all run-units invoked by an account number, terminal ID, operator ID, or transaction ID within the time interval you selected.

Billing System Summary Report-presents a sum of all Billing Summaries within the time interval you selected.

Billing Grand Summary Report-produced automatically when you request multiple time intervals. The Billing Grand Summary is simply a total of all the Billing System Summaries.
# Billing Details Report

The Billing Details Report presents a detailed view of each run-unit's activity reported in time sequence, based on the parameters selected. Depending on the parameter combination you select, this report allows you to identify run-unit activity by account number, terminal ID, operator ID, or transaction ID. CA ADS dialogs are reported as online programs.

You will get the Billing Details Report only if you specify LEVEL = DETAIL. In addition, with this specification, you will receive the Billing Summary Report, the Billing System Summary Report and possibly the Billing Grand Summary Report. (The Grand Summary Report is produced only if more than one time interval is reported.)

<table>
<thead>
<tr>
<th>CA /LOG ANALYZER</th>
<th>PAGE nn</th>
<th>CA IDMS BILLING REPORT</th>
</tr>
</thead>
<tbody>
<tr>
<td>mm/dd</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### DETAILS FOR OPER ID ADE

<table>
<thead>
<tr>
<th>mm - mm/dd/yy</th>
<th>hh:mm</th>
</tr>
</thead>
</table>

### TRANS ID TERM ID ---END DATE-TIME-- --------PAGES--------- ----RECORDS----- ---CALC RECORDS--- --VIA RECORDS--- ----TOTAL--- ---READ--- ---WRITTEN--- ---REQUEST--- ---REQUEST--- ---CURRENT--- ---HOME---

<table>
<thead>
<tr>
<th>PAGE</th>
<th>OVERFLW</th>
<th>OWNR PG</th>
<th>OVERFLW</th>
<th>I/O</th>
<th>CPU TIME</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>ADSGT ss 2</th>
<th>VTAMLT01 mm/dd/yy hh:mm</th>
<th>188</th>
<th>180</th>
<th>36</th>
<th>2</th>
<th>0</th>
<th>6</th>
<th>0</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>TCF ss 0</td>
<td>VTAMLT01 mm/dd/yy hh:mm</td>
<td>114</td>
<td>114</td>
<td>20</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>ADS ss 4</td>
<td>VTAMLT01 mm/dd/yy hh:mm</td>
<td>16</td>
<td>16</td>
<td>4</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>4</td>
</tr>
<tr>
<td>ADS ss 0</td>
<td>VTAMLT01 mm/dd/yy hh:mm</td>
<td>8</td>
<td>8</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>ADS ss 2</td>
<td>VTAMLT01 mm/dd/yy hh:mm</td>
<td>146</td>
<td>160</td>
<td>64</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>18</td>
</tr>
<tr>
<td>ADS2 ss 20</td>
<td>VTAMLT01 mm/dd/yy hh:mm</td>
<td>268</td>
<td>252</td>
<td>54</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>32</td>
</tr>
<tr>
<td>ADS2 ss 138</td>
<td>VTAMLT01 mm/dd/yy hh:mm</td>
<td>1338</td>
<td>1338</td>
<td>488</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>138</td>
</tr>
<tr>
<td>ADS2 ss 0</td>
<td>VTAMLT01 mm/dd/yy hh:mm</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>ADS2 ss 130</td>
<td>VTAMLT01 mm/dd/yy hh:mm</td>
<td>1274</td>
<td>1274</td>
<td>462</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>130</td>
</tr>
<tr>
<td>ADS2 ss 0</td>
<td>VTAMLT01 mm/dd/yy hh:mm</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>ADS2 ss 130</td>
<td>VTAMLT01 mm/dd/yy hh:mm</td>
<td>1274</td>
<td>1274</td>
<td>462</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>130</td>
</tr>
<tr>
<td>ADS2 ss 0</td>
<td>VTAMLT01 mm/dd/yy hh:mm</td>
<td>14</td>
<td>28</td>
<td>6</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>ADS2 ss 130</td>
<td>VTAMLT01 mm/dd/yy hh:mm</td>
<td>1274</td>
<td>1274</td>
<td>462</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>130</td>
</tr>
<tr>
<td>ADS2 ss 145</td>
<td>VTAMLT01 mm/dd/yy hh:mm</td>
<td>1274</td>
<td>1274</td>
<td>462</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>130</td>
</tr>
<tr>
<td>ADS2 ss 130</td>
<td>VTAMLT01 mm/dd/yy hh:mm</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>
Billing Details Report:

Note: Report for online run-units where OPER-ID is selected as RUNAME; TRANS-ID and TERM-ID appear as the first two columns on this report.
Note: Report for online run-units where TRANS-ID is selected as RUNAME; OPER-ID and TERM-ID appear as the first two columns on this report.

<table>
<thead>
<tr>
<th>TRANS_ID</th>
<th>OPER_ID</th>
<th>--END DATE-TIME---</th>
<th>--------PAGES--------</th>
<th>----RECORDS-----</th>
<th>---CALC RECORDS---</th>
<th>--VIA RECORDS---</th>
<th>----TOTAL---</th>
</tr>
</thead>
<tbody>
<tr>
<td>DCUF</td>
<td></td>
<td>mm/dd/yy hh:mm:ss</td>
<td>10 18 6 0 0 0 0 2</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SADS@OH</td>
<td></td>
<td>mm/dd/yy hh:mm:ss</td>
<td>24 24 10 0 0 0 0 4</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>DME</td>
<td></td>
<td>mm/dd/yy hh:mm:ss</td>
<td>618 618 262 0 0 0 0 66</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ADS2</td>
<td></td>
<td>mm/dd/yy hh:mm:ss</td>
<td>942 942 386 0 0 0 0 70</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ADS2</td>
<td></td>
<td>mm/dd/yy hh:mm:ss</td>
<td>84 84 22 0 0 0 0 10</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ADS2</td>
<td></td>
<td>mm/dd/yy hh:mm:ss</td>
<td>586 586 250 0 0 0 0 66</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ADS2</td>
<td></td>
<td>mm/dd/yy hh:mm:ss</td>
<td>0 0 0 0 0 0 0 0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ADS2</td>
<td></td>
<td>mm/dd/yy hh:mm:ss</td>
<td>0 0 0 0 0 0 0 0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ADS2</td>
<td></td>
<td>mm/dd/yy hh:mm:ss</td>
<td>284 284 98 0 0 0 0 58</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ADS2</td>
<td></td>
<td>mm/dd/yy hh:mm:ss</td>
<td>244 244 88 0 0 0 0 44</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ADS2</td>
<td></td>
<td>mm/dd/yy hh:mm:ss</td>
<td>252 252 80 0 0 0 0 52</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ADS2</td>
<td></td>
<td>mm/dd/yy hh:mm:ss</td>
<td>0 0 0 0 0 0 0 0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ADS2</td>
<td></td>
<td>mm/dd/yy hh:mm:ss</td>
<td>0 0 0 0 0 0 0 0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Billing Details Report:

Note: Report for online run-units where TERMIN-ID is selected as RUNAME; TRANS-ID and OPER-ID appear as the first two columns on this report.
<table>
<thead>
<tr>
<th>PROGRAM</th>
<th>END DATE-TIME</th>
<th>Pages</th>
<th>RECORDS</th>
<th>TOTAL</th>
<th>CPU TIME</th>
</tr>
</thead>
<tbody>
<tr>
<td>CALC RECORDS</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>VIA RECORDS</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>GE OVERFLW</td>
<td>OWNR PG OVERFLW I/O</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

| IDMSCHEM    | mm/dd/yy hh:mm:ss | 4408 1398 46544 37874 9296 90 104 492 256 5 |
|-------------|-------------------|-------|---------|-------|----------|
| IDMSCHEM    | 6 3 30 30 6 0 0 0 0 |
| IDMSCHEM    | 5368 2541 53952 50848 10040 91 102 511 242 7 |
| IDMSCHEM    | 8 0 70 70 8 0 0 0 0 |
| IDMSCHEM    | 938 22 10578 10552 3014 0 0 2 4 |
| IDMSCHEM    | 5376 2556 54028 50988 10040 91 102 511 242 7 |
| IDMSCHEM    | 5804 2768 60674 57098 11650 107 122 589 306 8 |
| IDMSCHEM    | 6030 3016 63392 59660 11902 107 122 600 308 9 |
| IDMSCHEM    | 6030 3020 63522 59790 11904 107 122 600 308 9 |
| IDMSCHEM    | 6036 3022 63532 59800 11906 107 122 600 308 9 |
| IDMSCHEM    | 6248 1958 60910 57168 14328 121 158 747 406 8 |
| IDMSCHEM    | 7654 3462 79762 74960 15206 121 158 747 406 11 |
| IDMSCHEM    | 6124 1920 61812 57812 14278 115 162 746 392 8 |
| IDMSCHEM    | 940 324 4950 7096 1822 25 38 95 2 1 |
| IDMSCHEM    | 916 385 430 6108 1612 23 38 87 2 |
| IDMSCHEM    | 40 4 20 12 0 0 0 0 0 |
| IDMSCHEM    | 4 27 |
| IDMSCHEM    | 0 0 20 32 12 0 0 0 0 |
| IDMSCHEM    | 0 27 |
| IDMSCHEM    | 4608 1183 30082 29146 8238 85 106 230 44 5 |
Billing Details Report Fields

Below is a description of the various fields that make up the Billing Details Report. Figure 2-1 through Figure 2-4 show the four possible types of Billing Details Reports CA IDMS Log Analyzer will generate.

**REPORT TITLE**: The title line of this report varies depending on what you select on the RUNAME, NAME, and RUTYPE parameter statements.

**INTERVAL**: This line lists the start and stop date/time of the time interval being reported. The data displayed in this line depends on what you select using the START, STOP, and INTERVAL parameters and on the actual date/time of the run-unit activity being reported.

**VARIABLE COLUMN(S)**: This column(s) varies both in content and number of columns. For online run-units, any combination of OPER-ID, TERM-ID, or TRANS-ID can appear in the first two columns. (Online RUNAME types that do not appear in the REPORT TITLE will appear in these two columns.) For BATCH run-units, one column appears, and it contains PROGRAM. See Figure 2-1 through Figure 2-4 for some sample report formats.

**END DATE-TIME**: Ending date/time of run-unit.

**IDMS STATISTICS** (taken from the log record)

- **PAGES READ**: Number of pages read from the database.
- **PAGES WRITTEN**: Number of pages written to the database.
- **PAGES REQUEST**: Number of pages requested from the database.
- **RECORDS REQUEST**: Number of records requested from the database.
- **RECORDS CURRENT**: Number of records made current of run-unit.
- **CALC RECORDS HOME PG**: Number of CALC records stored on the home page.
- **CALC RECORDS OVERFLW**: Number of CALC records stored on an overflow page.
- **VIA RECORDS OWNR PG**: Number of VIA records stored on the owner page.
- **VIA RECORDS OVERFLW**: Number of VIA records stored on an overflow page.
**TOTAL I/O**-Total number of input/output operations performed by the run-unit; the sum of PAGES READ plus PAGES WRITTEN.

**TOTAL CPU TIME**-Total CPU time used by the run-unit, reported in units of 1/100 seconds; this is the sum of USER-MODE-TIME plus SYSTEM-MODE-TIME as reported by CA IDMS in the log record.

**Billing Summary Report**

The Billing Summary Report summarizes all run-units executed for a terminal ID, operator ID, transaction ID, or account within the time interval you select. It is a summary of information from the Billing Details Report.

You will get this report if you specify LEVEL = SUMMARY (or if you specify LEVEL = DETAIL). In addition to the Billing Summary Report, you will also receive the Billing System Summary Report and possibly the Grand Summary Report.

When you look at the Billing Summary Report, focus on the COUNTS (IDMS STATISTICS) and RATIOS. These statistics reveal trends on the vitality of your database environment. MEAN VALUE, ACCUMULATED VALUE, and PERCENTAGE OF SYSTEM OCCURRENCES are reported for each statistical category.
Billing Summary Report

Here is an explanation of the fields that make up the Billing Summary Report. A sample report is shown in Figure 2-5.

**REPORT TITLE**--The title line of this report varies depending on what you selected on the RUNAME, NAME, and RUTYPE parameter statements.

**INTERVAL**--This line lists the start and stop date/time of the time interval being reported. The data displayed in this line depends on what you select using the START, STOP, and INTERVAL parameters and on the actual date/time of the run-unit activity being reported.

**RUN UNITS TOTAL**--Total number of run-units terminated within the reported time interval.

**COUNTS**--The IDMS STATISTICS (taken from the log record) are reported.

- **PAGES READ**--Number of pages read from the database.
- **PAGES WRITTEN**--Number of pages written to the database.
- **PAGES REQUESTED**--Number of pages requested from the database.
- **CALC RCDS ON HOME PAGE**--Number of CALC records stored on the home page.
- **CALC RCDS OVERFLOW**--Number of CALC records stored on an overflow page.
- **VIA RCDS ON OWNER PAGE**--Number of VIA records stored on the owner page.
- **VIA RCDS OVERFLOW**--Number of VIA records stored on an overflow page.
- **RECORDS REQUESTED**--Number of records requested from the database.
- **RECORDS BECOMING CURRENT**--Number of records made current of run-unit.
- **CALLS TO IDMSDBMS**--Number of DML verbs executed.
- **FRAGMENTS STORED**--Number of record fragments stored.
- **ROOTS OR RCDS RELOCATED**--Number of records relocated because of fragment recomposition.

**TOTAL I/O**--Total number of database input/output operations the run-unit performed.
TOTAL CPU (100THS SEC)--Total CPU time needed to execute the run-unit. (Reported in units of 1/100 seconds.)

MEAN VALUE--Average value per run-unit occurrence within the reported time interval.

ACCUMULATED VALUE--Total value for all run unit occurrences within the reported time interval.

% OF SYSTEM OCCURRENCES--This ratio (expressed as a percentage) is the accumulated value for this run-unit against the accumulated value for all selected run-units active within the reported time interval. This highlights the run-units that are consuming the largest amount of system resources.

RATIOS--A list of five ratios follows.

- **PAGES REQUESTED / PAGES READ**--This ratio measures the effectiveness of buffer pool size and allocation. Small ratios (less than 2.00) can indicate random processing, inadequate buffer pool size, or the need for additional buffer pools. A ratio of 20 is generally considered high.

- **RECORDS REQUESTED / PAGES READ**--This ratio measures the overall effectiveness of space management, CALC synonym handling, VIA options, and buffer management. Large ratios usually indicate effective buffering (i.e., the minimizing of database I/O). A ratio of 20 is generally considered high.

- **RECORDS REQUESTED / RECORDS BECOMING CURRENT**--This ratio measures the amount of processing transparency provided by CA IDMS. High ratios (a ratio of 20 is generally considered high) indicate that an excessive amount of database traversing is occurring before target records are retrieved. Pay close attention to sorted sets, sets without PRIOR or OWNER pointers, or program strategy that does not use currency efficiently.

- **CALC RCDS OVERFLOW / CALC RCDS ON HOME PAGE**--This ratio measures the randomness of the CALC field values, or how full the database areas are. Large ratios or steadily rising ratio values show that there are either a large number of CALC synonyms (multiple values that CALC to the same database page), or that space may be getting scarce and that one or more areas may need to be enlarged. Ideally, this field will show a ratio of less than one (1).

- **VIA RCDS OVERFLOW / VIA RCDS ON OWNER PAGE**--This ratio measures the effectiveness of the storage of VIA records, or how full database areas are. Large ratios or steadily increasing ratio values can show that there is a lack of clustering or packing of VIA records (near the associated OWNER record), or a lack of randomness of the OWNER record types of VIA member records. Space may be getting scarce and one or more areas may need to be enlarged. Ideally, this field shows ratios of less than one (1).

**Billing System Summary Report**

The Billing System Summary Report presents a sum-total of all Billing Summaries within the time interval you select. All statistical categories are reported within run-unit type: ONLINE, BATCH, and SYSTEM (SYSTEM is the total of both ONLINE and BATCH run-unit activity during the specified interval). The report shows actual accumulated values for RUN UNITS, COUNTS, and RATIOS. For COUNTS, it also presents the percentage that the value is of total system resources.

You will get this report by specifying LEVEL = SYSTEM. (This report will also be created if you specify LEVEL = DETAIL or LEVEL = SUMMARY.) You may also get the Grand Summary Report.
When you review this report, focus on the COUNTS (IDMS STATISTICS) and RATIOS. These statistics reflect trends on the vitality of your database environment.

BILLING REPORT

SYSTEM SUMMARY

Here is a description of the various fields that make up the System Summary Report. See Figure 2-6.

**INTERVAL**--This line lists the start and stop date/time of the time interval being reported. The data displayed in this line depends on what you selected using the START, STOP, and INTERVAL parameters and on the actual date/time of the run-unit activity being reported.
RUN UNITS TOTAL --Total number of run-units terminated within the reported time interval.

COUNTS--The IDMS STATISTICS are reported. For a detailed explanation of CA IDMS statistics, see Billing Summary Report.

- PAGES READ
- PAGES WRITTEN
- PAGES REQUESTED
- CALC RCDS ON HOME PAGE
- CALC RCDS OVERFLOW
- VIA RCDS ON OWNER PAGE
- VIA RCDS OVERFLOW
- RECORDS REQUESTED
- RECORDS BECOMING CURRENT
- CALLS TO IDMSDBMS
- FRAGMENTS STORED
- ROOTS OR RCDS RELOCATED

TOTAL I/O--Total number of input/output operations performed during the time interval.

TOTAL CPU (100THS SEC)--Total CPU time used during the time interval, reported in units of 1/100 seconds.

RATIOS--A list of five ratios follows. For a detailed explanation of the ratios that appear on the report, see Billing Summary Report.

- PAGES REQUESTED / PAGES READ
- RECORDS REQUESTED / PAGES READ
- RECORDS REQUESTED / RECORDS BECOMING CURRENT
- CALC RCDS OVERFLOW / CALC RCDS ON HOME PAGE
- VIA RCDS OVERFLOW / VIA RCDS ON OWNER PAGE

ONLINE--The accumulated value of COUNTS and RATIOS for all run-units that executed online during the time interval. Also, for COUNTS, the % OF SYSTEM OCCURRENCES is shown.

BATCH--The accumulated value of COUNTS and RATIOS for all run-units that executed batch during the time interval. Also, for COUNTS, the % OF SYSTEM OCCURRENCES is shown.
**SYSTEM**—The accumulated value of COUNTS and RATIOS for all run-units that executed during the time interval.

**ACCUMULATED VALUE**—Total value for all run-unit occurrences within the reported time interval.

**% OF SYSTEM OCCURRENCES**—For ONLINE and BATCH, this ratio (expressed as a percentage) is the accumulated value for this COUNT against the accumulated value for all selected (SYSTEM) run-units active within the reported time interval. This highlights the run-units that are consuming the largest amount of system resources.

### Billing Grand Summary Report

CA IDMS Log Analyzer automatically produces a Billing Grand Summary Report whenever multiple time intervals are selected (i.e., when the INTERVAL is less than the entire START STOP DATE/TIME period). This report is a total of all Billing System Summaries.

<table>
<thead>
<tr>
<th>CA / LOG ANALYZER</th>
<th>CA IDMS / PAGE nn</th>
<th>BILLING REPORT</th>
</tr>
</thead>
<tbody>
<tr>
<td>mm/dd yy hh:mm:ss</td>
<td>GL AND ANALYZER</td>
<td>PAGE nn</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>RUNUNITS</th>
<th>TOTAL</th>
<th>100.00</th>
<th>0</th>
<th>.</th>
</tr>
</thead>
<tbody>
<tr>
<td>ONLINE</td>
<td>67</td>
<td>100.00</td>
<td>0</td>
<td>.</td>
</tr>
<tr>
<td>BATCH</td>
<td>67</td>
<td>100.00</td>
<td>0</td>
<td>.</td>
</tr>
<tr>
<td>SYSTEM</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>COUNTS</th>
<th>PAGES READ</th>
<th>3,353</th>
<th>100.00</th>
<th>0</th>
<th>.</th>
</tr>
</thead>
<tbody>
<tr>
<td>ONLINE</td>
<td>3,353</td>
<td>3,353</td>
<td>100.00</td>
<td>0</td>
<td>.</td>
</tr>
<tr>
<td>BATCH</td>
<td>3,353</td>
<td>3,353</td>
<td>100.00</td>
<td>0</td>
<td>.</td>
</tr>
<tr>
<td>SYSTEM</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>PAGES WRITTEN</th>
<th>6</th>
<th>100.00</th>
</tr>
</thead>
<tbody>
<tr>
<td>6</td>
<td>100.00</td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>13,129</td>
<td>100.00</td>
</tr>
<tr>
<td>3</td>
<td>100.00</td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>6</td>
<td>100.00</td>
</tr>
<tr>
<td>0</td>
<td>6</td>
<td>100.00</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>13,219</td>
<td>100.00</td>
</tr>
<tr>
<td>0</td>
<td>4,389</td>
<td>100.00</td>
</tr>
<tr>
<td>0</td>
<td>2,166</td>
<td>100.00</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>3,359</td>
<td>100.00</td>
</tr>
<tr>
<td>0</td>
<td>2,085</td>
<td>100.00</td>
</tr>
<tr>
<td>0</td>
<td>3.92</td>
<td>3.92</td>
</tr>
</tbody>
</table>

**RATIOS**—

- PAGES REQUESTED / PAGES READ: 3.92
- RECORDS REQUESTED / RECORDS STORED: 0.00
- RECORDS REQUESTED / RECORDS STORED: 3.92

---

10-Jan-2018
Billing Grand Summary Report Fields

The report fields for the Billing Grand Summary Report are identical to those on the Billing System Summary Report. See Billing System Summary Report for a description of these fields. Figure 2-7 shows a Billing Grand Summary Report.

**INTERVAL**--This line lists the start and stop date/time of the time interval being reported. The data displayed in this line depends on what you selected using the START, STOP, and INTERVAL parameters and on the actual date/time of the run-unit activity being reported.

**RUN UNITS TOTAL**--Total number of run-units terminated within the reported time interval selected.

**COUNTS**--IDMS STATISTICS are listed below. For a detailed explanation of CA IDMS statistics, see Billing Summary Report.

- **PAGES READ**
- **PAGES WRITTEN**
- **PAGES REQUESTED**
- **CALC RCDS ON HOME PAGE**
- **CALC RCDS OVERFLOW**
- **VIA RCDS ON OWNER PAGE**
- **VIA RCDS OVERFLOW**
- **RECORDS REQUESTED**
- **RECORDS BECOMING CURRENT**
- **CALLS TO IDMSDBMS**
- **FRAGMENTS STORED**
- **ROOTS OR RCDS RELOCATED**

**TOTAL I/O**--Total number of input/output operations performed during the time interval.

**TOTAL CPU (100THS SEC)**--Total CPU time used during the time interval, reported in units of 1/100 seconds.

**RATIOS**--A list of five ratios follows. For a detailed explanation of the ratios, see the Billing System Summary Report.
About CA IDMS Log Analyzer Program Reports

Contents

- Four Reports--One Set of CA IDMS Log Statistics (see page 31)
- Hierarchical Nature of Reports (see page 32)

The Program Reports are similar to the Billing Reports. They use information from the CA IDMS Log to offer detailed and summarized statistical reports that show how efficiently application programs are using CA IDMS.

Program Report differs from Billing Reports in that they are organized by the run-unit's program name. (For tasks executed within CA IDMS/DC, these reports are organized by the run-unit's task-ID.) Information about the application program is grouped by selected time intervals. First, each program's totals are reported, then they are reflected as a ratio of the system totals (i.e., the percentage of all CA IDMS resources consumed during the time interval by programs that were selected by CA IDMS Log Analyzer).

You will get this report by specifying REPORT = PROGRAM.

Four Reports--One Set of CA IDMS Log Statistics

Physically, there are four Program Reports to choose from. It is important to understand, however, that each report is produced from the same statistics taken from the CA IDMS Log. Statistics are presented in different formats and at various levels of summarization. The Program Report is available at the detail, summary, system summary, and grand summary level. The Program Grand Summary Report is produced automatically if you choose multiple time intervals.
Hierarchical Nature of Reports

Program reports are produced for several hierarchical levels: if you ask for the lowest level report (LEVEL = DETAIL), you will also receive the higher-level reports. These would include the Program Summary Report, which summarizes the data of the Program Details Report (LEVEL = SUMMARY), as well as the Program System Summary Report (LEVEL = SYSTEM). A Program Grand Summary Report is simply a summary of all Program System Summary Reports. This report is controlled by the INTERVAL parameter.

Overview Of Program Reports

Contents
- Program Details Report (see page 32)
  - Program Details Report Fields (see page 33)
- Program Summary Report (see page 34)
  - Program Summary Report Fields (see page 35)
- Program System Summary Report (see page 37)
  - Program System Summary Report Fields (see page 38)
- Program Grand Summary Report (see page 40)
  - Program Grand Summary Report Fields (see page 40)

Program Details Report--This report shows the information for each run-unit of a program, (or CA IDMS/DC task) reported in time sequence, within the selected time interval. You may specify which program or class of programs CA IDMS Log Analyzer is to report on. You also control the time period and duration of the time interval.

Program Summary Report--records the sum for all run-units of an application program within the time interval you select.

Program System Summary Report--presents a sum of all Program Summary Reports within the time interval you select.

Program Grand Summary Report--produced automatically when you request multiple time intervals. The Program Grand Summary Report is simply a total of all the Program System Summaries.

Program Details Report

The Program Details Report presents a detailed view of application program activity (or CA IDMS/DC task activity or CA ADS dialog activity). Most of the statistics in the report are IDMS STATISTICS taken from the CA IDMS Log. You will get this report when you request LEVEL = DETAILS.

Request this report when there are more than 20 run-unit occurrences for a program, when a strictly chronological sequence of all program activity is desired, or when you want to check every execution of a program to investigate a problem.
Program Details Report Fields

Here is a description of the various fields that make up the Program Details Report. See Figure 2-8.

REPORT TITLE--The title line contains the name of the application program (or CA IDMS/DC task) being reported.

INTERVAL--This line lists the start and stop date/time of the time interval being reported. The data displayed in this line depends on what you selected using the START, STOP, and INTERVAL parameters and on the actual date/time of the run-unit activity being reported.

RUTYPE--This line tells whether the application program executed as a BATCH or ONLINE run-unit.

END DATE-TIME--Ending date/time of run-unit.
IDMS STATISTICS--The IDMS STATISTICS (listed below) are taken from the log record.

- **PAGES READ**--Number of pages read from the database.
- **PAGES WRITTEN**--Number of pages written to the database.
- **PAGES REQUEST**--Number of pages requested from the database.
- **RECORDS REQUEST**--Number of records requested from the database.
- **RECORDS CURRENT**--Number of records made current of run-unit.
- **CALC RECORDS HOME PG**--Number of CALC records stored on the home page.
- **CALC RECORDS OVERFLW**--Number of CALC records stored on an overflow page.
- **VIA RECORDS OWNR PG**--Number of VIA records stored on the owner page.
- **VIA RECORDS OVERFLW**--Number of VIA records stored on an overflow page.

**TOTAL I/O**--Total number of input/output operations performed by the run-unit; the sum of PAGES READ plus PAGES WRITTEN.

**TOTAL CPU TIME**--Total CPU time used by the run-unit; reported in units of 1/100 seconds. TOTAL CPU is the sum of USER-MODE-TIME plus SYSTEM-MODE-TIME as reported by CA IDMS in the log record.

Program Summary Report

The Program Summary Report summarizes all run-unit executions of an application program (or CA IDMS/DC task) within a time interval. The Program Summary Report summarizes information taken from the Program Details Report. You will get this report by specifying LEVEL = SUMMARY (or LEVEL = DETAIL). In addition, you will also receive all higher-level reports.

When you look at the Program Summary Totals Report, focus on the COUNTS (IDMS STATISTICS) and RATIOS. These statistics reveal trends on the vitality of your database environment. This report also presents other CA IDMS statistics unavailable on the Program Details Report. MEAN VALUE, ACCUMULATED VALUE, and PERCENTAGE OF SYSTEM OCCURRENCES are reported for each statistical category.
Program Summary Report:

Here is an explanation of the fields that make up the Program Summary Report. See Figure 2-9.

**REPORT TITLE**--The title line contains the name of the report, the name of the application program (or CA IDMS/DC task) being reported, and whether that program executed as an ONLINE or BATCH run-unit.

**INTERVAL**--This line lists the start and stop date/time of the time interval being reported. The data displayed here depends on what you selected using the START, STOP, and INTERVAL parameters and on the actual date/time of the program activity being reported.

**RUN UNITS TOTAL**--Total number of run-units terminated within the reported time interval.

**COUNTS**--The following IDMS STATISTICS (taken from the log record) are reported.

- **PAGES READ**--Number of pages read from the database.
- **PAGES WRITTEN**--Number of pages written to the database.
- **PAGES REQUESTED**--Number of pages requested from the database.
- **CALC RCDS ON HOME PAGE**--Number of CALC records stored on the home page.
- **CALC RCDS OVERFLOW**--Number of CALC records stored on an overflow page.
- **VIA RCDS ON OWNER PAGE**--Number of VIA records stored on the owner page.

- **VIA RCDS OVERFLOW**--Number of VIA records stored on an overflow page.

- **RECORDS REQUESTED**--Number of records requested from the database.

- **RECORDS BECOMING CURRENT**--Number of records made current of run-unit.

- **CALLS TO IDMSDBMS**--Number of DML verbs executed.

- **FRAGMENTS STORED**--Number of record fragments stored.

- **ROOTS OR RCDS RELOCATED**--Number of records relocated because of fragment recomposition.

**TOTAL I/O**--Total number of database input/output operations the program performed during the time interval.

**TOTAL CPU (100THS SEC)**--Total CPU time needed to execute the program during the time interval specified reported in units of 1/100 seconds.

**MEAN VALUE**--Average value per run-unit occurrence within the reported time interval.

**ACCUMULATED VALUE**--Total value for all run-unit occurrences within the reported time interval.

**% OF SYSTEM OCCURRENCES**--This ratio (expressed as a percentage) is the accumulated value for this run-unit against the accumulated value for all selected run-units active within the reported time interval. This highlights the run-units that are consuming the largest amount of system resources.

**RATIOS**--A list of five ratios follows.

- **PAGES REQUESTED / PAGES READ**--This ratio measures the effectiveness of buffer pool size and allocation. Small ratios (less than 2.00) can indicate random processing, inadequate buffer pool size, or the need for additional buffer pools. A ratio of 20 is generally considered high.

- **RECORDS REQUESTED / PAGES READ**--This ratio measures the overall effectiveness of space management, CALC synonym handling, VIA options, and buffer management. Large ratios usually indicate effective buffering (i.e., the minimizing of database I/O). A ratio of 20 is generally considered high.

- **RECORDS REQUESTED / RECORDS BECOMING CURRENT**--This ratio measures the amount of processing transparency provided by CA-IDMS. High ratios (a ratio of 20 is generally considered high) indicate that an excessive amount of database traversing is occurring before target records are retrieved. Pay close attention to sorted sets, sets without PRIOR or OWNER pointers, or program strategy that does not use currency efficiently.

- **CALC RCDS OVERFLOW / CALC RCDS ON HOME PAGE**--This ratio measures the randomness of the CALC field values, or how full particular database areas are. Large ratios or steadily rising ratio values show that there are either a large number of CALC synonyms, or that space may be getting scarce and that one or more areas may need to be enlarged. Ideally, this field will show ratios of less than one (1).
VIA RCDS OVERFLOW / VIA RCDS ON OWNER PAGE--This ratio measures the effectiveness of the storage of VIA records, or how full database areas are. Large ratios or steadily increasing ratio values can show that there is a lack of clustering or packing of VIA records (near the associated OWNER record), or a lack of randomness of the OWNER record types of VIA member records. Space may be getting scarce and one or more areas may need to be enlarged. Ideally, this field shows ratios of less than one (1).

Program System Summary Report

The Program System Summary Report presents a sum-total of all Program Summaries within the time interval you select. All statistical categories are reported within run-unit type: ONLINE, BATCH, and SYSTEM (SYSTEM is the total of both ONLINE and BATCH run-unit activity during the specified interval). The report shows the actual accumulated values for RUN UNITS, COUNTS, and RATIOS. For COUNTS, it also shows the mean value and the percentage that the value is of total system occurrences. You will get this report by specifying LEVEL = SYSTEM.

When you review this report, focus on COUNTS (IDMS STATISTICS) and RATIOS. These statistics reflect trends on the vitality of your database environment.
Program System Summary Report Fields

Here is a description of the various fields that make up the Program System Summary Report. See Figure 2-10.

**INTERVAL**--This line lists the start and stop date/time of the time interval being reported. The data displayed here depends on what you selected using the START, STOP, and INTERVAL parameters and on the actual date/time of the run-unit activity being reported.

**RUN UNITS TOTAL**--Total number of run-units terminated within the reported time interval.

**COUNTS**--The IDMS STATISTICS are reported. For a detailed explanation of CA IDMS statistics, see Program Summary Report.

- PAGES READ
- PAGES WRITTEN
- PAGES REQUESTED
- CALC RCDS ON HOME PAGE
- CALC RCDS OVERFLOW
- VIA RCDS ON OWNER PAGE
- VIA RCDS OVERFLOW
- RECORDS REQUESTED
- RECORDS BECOMING CURRENT
- CALLS TO IDMSDBMS
- FRAGMENTS STORED
- ROOTS OR RCDS RELOCATED

**TOTAL I/O**--Total number of input/output operations performed during the time interval.

**TOTAL CPU (100THS SEC)**--Total CPU time used during the time interval, reported in units of 1/100 seconds.
**RATIOS**—A list of five ratios follows. For a detailed explanation of the ratios, see Program Summary Report.

- **PAGES REQUESTED / PAGES READ**
- **RECORDS REQUESTED / PAGES READ**
- **RECORDS REQUESTED / RECORDS**
- **BECOMING CURRENT**
- **CALC RCDS OVERFLOW / CALC RCDS ON HOME PAGE**
- **VIA RCDS OVERFLOW / VIA RCDS ON OWNER PAGE**

**ONLINE**—The accumulated value of COUNTS and RATIOS for all run-units that executed online during the time interval. Also, for COUNTS, the PERCENTAGE OF SYSTEM OCCURRENCES is shown.

**BATCH**—The accumulated value of COUNTS and RATIOS for all run-units that executed batch during the time interval. Also, for COUNTS, the % OF SYSTEM OCCURRENCES is shown.

**SYSTEM**—The accumulated value of COUNTS and RATIOS for all run-units that executed during the time interval.

**ACCUMULATED VALUE**—Total value for all run-unit occurrences within the reported time interval.

**% OF SYSTEM OCCURRENCES**—For ONLINE and BATCH, this ratio (expressed as a percentage) is the accumulated value for this COUNT against the accumulated value for all selected (SYSTEM) run-units active within the reported time interval. This highlights the run-units that are consuming the largest amount of system resources.

<table>
<thead>
<tr>
<th>CATEGORY</th>
<th>ONLINE</th>
<th>BATCH</th>
<th>SYSTEM</th>
</tr>
</thead>
<tbody>
<tr>
<td>RUNUNITS</td>
<td>67</td>
<td>67</td>
<td></td>
</tr>
<tr>
<td>TOTAL</td>
<td>67</td>
<td>67</td>
<td></td>
</tr>
<tr>
<td>COUNTS</td>
<td>3,353</td>
<td></td>
<td>6</td>
</tr>
<tr>
<td>PAGES READ</td>
<td>3,353</td>
<td></td>
<td>6</td>
</tr>
<tr>
<td>PAGES WRITTEN</td>
<td>6</td>
<td>13,129</td>
<td>3</td>
</tr>
<tr>
<td>PAGES REQUESTED</td>
<td>6</td>
<td>13,129</td>
<td>3</td>
</tr>
<tr>
<td>CALC RCDS ON HOME</td>
<td>3</td>
<td></td>
<td>0</td>
</tr>
<tr>
<td>CALC RCDS OVERFLOW</td>
<td></td>
<td></td>
<td>0</td>
</tr>
<tr>
<td>VIA RCDS ON OWNER</td>
<td>6</td>
<td>100.00</td>
<td>0</td>
</tr>
<tr>
<td>Grand Summa</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Program Grand Summary Report

CA IDMS Log Analyzer automatically produces a Program Grand Summary Report whenever multiple time intervals are selected (i.e., when the INTERVAL is less than the entire START STOP DATE/TIME period.) This report is a total of all Program System Summaries.

Program Grand Summary Report Fields

The report fields for the Program Grand Summary Report are identical to those on the Program System Summary Report. See Program System Summary Report for a description of these fields. Figure 2-11 shows an example of a Program Grand Summary Report.

INTERVAL--This line lists the start and stop date/time of the time interval being reported. The data displayed in this line depends on what you selected using the START and STOP parameters.

RUN-UNITS TOTAL--Total number of run-units terminated within the reported time interval.

COUNTS--The IDMS STATISTICS are reported. For a detailed explanation of CA IDMS statistics, see Program Summary Report.

- PAGES READ
- PAGES WRITTEN
- PAGES REQUESTED
- CALC RCDS ON HOME PAGE
- CALC RCDS OVERFLOW
- VIA RCDS ON OWNER PAGE
- VIA RCDS OVERFLOW
- RECORDS REQUESTED
- RECORDS BECOMING CURRENT
- CALLS TO IDMSDBMS
- FRAGMENTS STORED
- ROOTS OR RCDS RELOCATED

**TOTAL I/O**—Total number of input/output operations performed during the time interval.

**TOTAL CPU (100THS SEC)**—Total CPU time used during the time interval, reported in units of 1/100 seconds.

**RATIOS**—A list of five ratios follows. For a detailed explanation of the ratios, see Program Summary Report.

- PAGES REQUESTED / PAGES READ
- RECORDS REQUESTED / PAGES READ
- RECORDS REQUESTED / RECORDS BECOMING CURRENT
- CALC RCDS OVERFLOW / CALC RCDS HOME PAGE
- VIA RCDS OVERFLOW / VIA RCDS ON HOME PAGE

**ONLINE**—The accumulated value of COUNTS and RATIOS for all run-units that executed online during the time interval. Also, for COUNTS, the PERCENTAGE OF SYSTEM OCCURRENCES is shown.

**BATCH**—The accumulated value of COUNTS and RATIOS for all run-units that executed batch during the time interval. Also, for COUNTS, the % OF SYSTEM OCCURRENCES is shown.

**SYSTEM**—The accumulated value of COUNTS and RATIOS for all run-units that executed during the time interval.

**ACCUMULATED VALUE**—Total value for all run unit occurrences within the reported time interval.

**% OF SYSTEM OCCURRENCES**—For ONLINE and BATCH, this ratio (expressed as a percentage) is the accumulated value for this COUNT against the accumulated value for all selected (SYSTEM) run-units active within the reported time interval. This highlights the run-units that are consuming the largest amount of system resources.

---

**About CA IDMS Log Analyzer Management Reports**

**Contents**

- Three Highlights/Summary Reports—One Set of CA IDMS Statistics (see page 43)
- Hierarchical Nature of Highlights/Summary Reports (see page 43)
- Management Highlights/Program Summary Report (see page 43)
There are three types of Management Reports: Highlights/Summaries (three reports), Highlights/Buffer Pool Utilization Report, and the Ranking Report (one report each).

**Highlights/Summaries** are produced at three levels of detail:

- HIGHLIGHTS/SUMMARIES
  - Highlights Program Summary
  - Highlights System Summary
  - Highlights Grand Summary
- HIGHLIGHTS/BUFFER POOL UTILIZATION
- RANKING REPORT

**Highlights/Summary Reports**—These reports are summaries of system performance and resource consumption derived from the Program Report.

Highlights/Summary Reports present totals, highest, lowest, means, and medians of program attributes and program consumption (IDMS STATISTICS). These are presented at program summary, system summary, and grand summary levels. You will get this report by specifying REPORT = HI-SUM on the parameter statement.

**Highlights/Buffer Pool Utilization Report**—This report presents the number of run-units (and percentage of the total that these represent) falling into each of several categories as viewed by Buffer Pool Ratio (PAGES REQUESTED divided by PAGES READ) for each hour of each day within the time period you selected. You will get this report by specifying REPORT = HI-BPU on the parameter statement.

**Ranking Report**—This report type also uses system performance and resource consumption information derived from the Program Details Report. However, the Ranking Report presents the information in ranked order, rather than in summarized form. You designate the run-unit category to be selected for ranking, the particular attribute to be ranked, and value type of ranking process to be used.
For example, you may specify that all online run-units are to be ranked by the median value of their pages read, that this information be presented in a highest to lowest sequence and that you only want to see the first 10 run-units sequenced in this way. You will get this report by specifying REPORT = RANK on the parameter statement.

Three Highlights/Summary Reports--One Set of CA IDMS Statistics

Physically, there are three Highlights/Summary Reports to choose from. However, it is important to understand that each report is produced from the same statistics as found on the CA IDMS Log. Statistics are presented in different formats and at different levels of summarization. The Highlights/Summary Report is available at the program, system, and grand summary level.

Hierarchical Nature of Highlights/Summary Reports

Highlights/Summary Reports are produced on a hierarchical basis: if you ask for the lowest level report (LEVEL = PROGRAM), you will also receive the higher-level reports. These would include the Highlights/Summary System Report, which summarizes the Highlights/Summary Program Report (LEVEL = SYSTEM) and possibly the Highlights/Summary Grand Report. The Grand Report is produced if there are multiple intervals within the reported period, or if the report is explicitly requested (LEVEL = GRAND).

Management Highlights/Program Summary Report

The Highlights/Program Summary Report provides a summary of system performance and database resource consumption for an individual program within a single time interval. You will get this report if you specified LEVEL = PROGRAM on the parameter statement.
Management Highlights/Program Summary Report Fields

These fields make up the Management Highlights/Program Summary Report. See Figure 2-12.

REPORT TITLE--This line lists the title of the report.

PROGRAM NAME/TYPE/INTERVAL--This line lists the name of the application program whose run-units are selected for reporting, its processing type (whether the run-units are ONLINE or BATCH), and start/stop date and time of report.

RUN UNITS TOTAL--Total number of run-units terminated within the reported time interval.

LOWEST--Lowest value encountered for any run-unit within the reported time interval.

HIGHEST--Highest value encountered for any run-unit within the reported time interval.

MEAN--Average value for all run-units within the reported time interval.

MEDIAN--Median value for all run-units within the reported time interval.

COUNTS--The IDMS STATISTICS (taken from the log record) are reported.

- PAGES READ--Number of pages read from the database.
- PAGES WRITTEN--Number of pages written to the database.
- PAGES REQUESTED--Number of pages requested from the database.
- CALC RCDS ON HOME PAGE--Number of CALC records stored on the home page.
- CALC RCDS OVERFLOW--Number of CALC records stored on an overflow page.
- VIA RCDS ON OWNER PAGE--Number of VIA records stored on the owner page.
- VIA RCDS OVERFLOW--Number of VIA records stored on an overflow page.
- **RECORDS REQUESTED**—Number of records requested.
- **RECORDS BECOMING CURRENT**—Number of records made current of run-unit.
- **CALLS TO IDMSDBMS**—Number of DML verbs executed.
- **FRAGMENTS STORED**—Number of record fragments stored.
- **ROOTS OR RCDS RELOCATED**—Number of records relocated because of fragment recomposition.

**TOTAL I/O**—Total number of database input/output operations performed by the program during the time interval.

**TOTAL CPU (100THS SEC)**—Total CPU time used by the program during the time interval specified.

**RATIOS**—The Ratios are reported in four values: Lowest, Highest, Mean, and Median.

- **PAGES REQUESTED / PAGES READ**—This ratio measures the effectiveness of buffer pool size and allocation. Small ratios (less than 2.00) can indicate random processing, inadequate buffer pool size, or the need for additional buffer pools. A ratio of 20 is generally considered high.

- **RECORDS REQUESTED / PAGES READ**—This ratio measures the overall effectiveness of space management, CALC synonym handling, VIA options, and buffer management. Large ratios usually indicate effective buffering (i.e., the minimizing of database I/O). A ratio of 20 is generally considered high.

- **RECORDS REQUESTED / RECORDS BECOMING CURRENT**—This ratio measures the amount of processing transparency provided by CA IDMS. High ratios (a ratio of 20 is generally considered high) indicate that an excessive amount of database traversing is occurring before target records are retrieved. Pay close attention to sorted sets, sets without PRIOR or OWNER pointers, or program strategy that does not use currency efficiently.

- **CALC RCDS OVERFLOW / CALC RCDS ON HOME PAGE**—This ratio measures the randomness of the CALC field values, or how full particular database areas are. Large ratios or steadily rising ratio values show that there are either a large number of CALC synonyms, or that space may be getting scarce and that one or more areas may need to be enlarged. Ideally, this field will show ratios of less than one (1).

- **VIA RCDS OVERFLOW / VIA RCDS ON OWNER PAGE**—This ratio measures the effectiveness of the storage of VIA records, or how full database areas are. Large ratios or steadily increasing ratio values can show that there is a lack of clustering or packing of VIA records (near the associated OWNER record), or a lack of randomness of the OWNER record types of VIA member records. Space may be getting scarce and one or more areas may need to be enlarged. Ideally, this ratio will show values of less than one (1).
### Management Highlights/System Summary Report (page 1):

<table>
<thead>
<tr>
<th>Metric</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>RUN UNITS</td>
<td></td>
</tr>
<tr>
<td>TOTAL</td>
<td>2 100.00</td>
</tr>
<tr>
<td>RATIOS</td>
<td></td>
</tr>
<tr>
<td>/PAGES READ</td>
<td></td>
</tr>
<tr>
<td>LOWEST</td>
<td>5.00</td>
</tr>
<tr>
<td>HIGHEST</td>
<td>5.00</td>
</tr>
<tr>
<td>MEAN</td>
<td>5.00</td>
</tr>
<tr>
<td>RECORDS REQUESTED</td>
<td></td>
</tr>
<tr>
<td>/PAGES READ</td>
<td></td>
</tr>
<tr>
<td>LOWEST</td>
<td>8.00</td>
</tr>
<tr>
<td>HIGHEST</td>
<td>8.00</td>
</tr>
<tr>
<td>MEAN</td>
<td>8.00</td>
</tr>
<tr>
<td>RECORDS REQUESTED</td>
<td></td>
</tr>
<tr>
<td>/RECORDS BECOMING CURRENT</td>
<td></td>
</tr>
<tr>
<td>LOWEST</td>
<td>2.67</td>
</tr>
<tr>
<td>HIGHEST</td>
<td>3.00</td>
</tr>
<tr>
<td>MEAN</td>
<td>2.84</td>
</tr>
<tr>
<td>CALC RCDS ON HOME PAGE</td>
<td></td>
</tr>
<tr>
<td>LOWEST</td>
<td>.00</td>
</tr>
<tr>
<td>HIGHEST</td>
<td>.00</td>
</tr>
<tr>
<td>MEAN</td>
<td>.00</td>
</tr>
<tr>
<td>VIA RCDS OVERFLOW</td>
<td></td>
</tr>
<tr>
<td>LOWEST</td>
<td>.00</td>
</tr>
<tr>
<td>HIGHEST</td>
<td>.00</td>
</tr>
<tr>
<td>MEAN</td>
<td>.00</td>
</tr>
<tr>
<td>CALC RCDS OVERFLOW</td>
<td></td>
</tr>
<tr>
<td>LOWEST</td>
<td>.00</td>
</tr>
<tr>
<td>HIGHEST</td>
<td>.00</td>
</tr>
<tr>
<td>MEAN</td>
<td>.00</td>
</tr>
<tr>
<td>PAGES READ</td>
<td></td>
</tr>
<tr>
<td>LOWEST</td>
<td>2</td>
</tr>
<tr>
<td>HIGHEST</td>
<td>2</td>
</tr>
<tr>
<td>MEAN</td>
<td>2.00</td>
</tr>
<tr>
<td>PAGES WRITT</td>
<td></td>
</tr>
<tr>
<td>LOWEST</td>
<td>2</td>
</tr>
<tr>
<td>HIGHEST</td>
<td>2</td>
</tr>
<tr>
<td>MEAN</td>
<td>2.00</td>
</tr>
<tr>
<td>PAGES REQUESTED</td>
<td></td>
</tr>
<tr>
<td>LOWEST</td>
<td>1</td>
</tr>
<tr>
<td>HIGHEST</td>
<td>10</td>
</tr>
<tr>
<td>MEAN</td>
<td>10.00</td>
</tr>
<tr>
<td>CALC RCDS ON HOME PAGE</td>
<td></td>
</tr>
<tr>
<td>LOWEST</td>
<td></td>
</tr>
<tr>
<td>HIGHEST</td>
<td></td>
</tr>
<tr>
<td>MEAN</td>
<td></td>
</tr>
<tr>
<td>CALC RCDS ON OWNER PAGE</td>
<td></td>
</tr>
<tr>
<td>LOWEST</td>
<td></td>
</tr>
<tr>
<td>HIGHEST</td>
<td></td>
</tr>
<tr>
<td>MEAN</td>
<td></td>
</tr>
<tr>
<td>PAGES REQUESTED</td>
<td></td>
</tr>
<tr>
<td>LOWEST</td>
<td></td>
</tr>
<tr>
<td>HIGHEST</td>
<td></td>
</tr>
<tr>
<td>MEAN</td>
<td></td>
</tr>
<tr>
<td>CALC RCDS ON OWNER PAGE</td>
<td></td>
</tr>
<tr>
<td>LOWEST</td>
<td></td>
</tr>
<tr>
<td>HIGHEST</td>
<td></td>
</tr>
<tr>
<td>MEAN</td>
<td></td>
</tr>
<tr>
<td>ONLINE</td>
<td></td>
</tr>
<tr>
<td>BATCH</td>
<td></td>
</tr>
<tr>
<td>SYSTEM</td>
<td></td>
</tr>
<tr>
<td>MANAGEMENT HIGHLIGHTS</td>
<td></td>
</tr>
<tr>
<td>PAGE nn</td>
<td></td>
</tr>
<tr>
<td>mm/dd/yy</td>
<td></td>
</tr>
<tr>
<td>SYSTEM SUMMARY</td>
<td></td>
</tr>
<tr>
<td>hh:mm:ss</td>
<td>mm/dd/yy hh:mm:ss</td>
</tr>
</tbody>
</table>
### Management Highlights/System Summary Report (page 2):

<table>
<thead>
<tr>
<th>Metric</th>
<th>Lowest</th>
<th>Mean</th>
<th>Highest</th>
</tr>
</thead>
<tbody>
<tr>
<td>Records Requested</td>
<td>1</td>
<td>16.00</td>
<td>18</td>
</tr>
<tr>
<td>Records Becoming Current</td>
<td>6</td>
<td>16.00</td>
<td>18</td>
</tr>
<tr>
<td>Calls to IDMSDBMS</td>
<td>6</td>
<td>6</td>
<td>7</td>
</tr>
<tr>
<td>Fragments Stored</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Roots or RCDS Relocated</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>TOTAL I/O</td>
<td>2</td>
<td>2.00</td>
<td>2</td>
</tr>
<tr>
<td>Total CPU (100ths Sec)</td>
<td>5</td>
<td>2.00</td>
<td>59</td>
</tr>
</tbody>
</table>

Mean values are also provided for each metric.
This report, a summary of system performance and resource consumption, is derived from the Program Report. The Highlights/System Summary Report gives you a summary of system performance and resource consumption accumulated for one time interval. This report presents highest, lowest, and means of program attributes, lists them under ONLINE, BATCH, and SYSTEM categories, for program consumption (IDMS STATISTICS). You will get this report if you select LEVEL = SYSTEM on the parameter statement. In addition, you will also receive all higher-level reports. When you look at this report, focus on the COUNTS (IDMS STATISTICS) and RATIOS. These statistics reveal the trends on the vitality of your database environment.

Management Highlights/System Summary Report Fields

These fields make up the two-page Highlights/System Summary Report. See Figure 2-13 and Figure 2-14.

REPORT TITLE--This line lists the title of the report.

INTERVAL--This line lists the start/stop date and time of the report.

RUN UNITS TOTAL--Number of run-units that terminated during this time interval.

RATIOS--The Ratios are reported in three values (LOWEST, HIGHEST, MEAN) in three categories (ONLINE, BATCH, SYSTEM). See Program Summary Report Fields for a complete description of the ratios.

- PAGES REQUESTED / PAGES READ
- RECORDS REQUESTED / PAGES READ
- RECORDS REQUESTED / RECORDS
- BECOMING CURRENT
- CALC RCDS OVERFLOW / CALC RCDS ON HOME PAGE
- VIA RCDS OVERFLOW / VIA RCDS ON OWNER PAGE

COUNTS--IDMS STATISTICS are reported in three values: LOWEST, HIGHEST, and MEAN. For a detailed explanation of CA IDMS statistics, see Program Summary Report Fields.

- PAGES READ
- PAGES WRITTEN
- PAGES REQUESTED
- CALC RCDS ON HOME PAGE
- CALC RCDS OVERFLOW
- VIA RCDS ON OWNER PAGE
- VIA RCDS OVERFLOW
- RECORDS REQUESTED
- RECORDS BECOMING CURRENT
- CALLS TO IDMSDBMS
- FRAGMENTS STORED
- ROOTS OR RCDS RELOCATED

**TOTAL I/O**—Total number of database input/output operations the run-unit performed during the time interval.

**TOTAL CPU (100THS SEC)**—Total CPU time needed to execute the run-unit during the time interval specified.

**LOWEST**—Lowest value encountered for run-unit within the reported time interval.

**HIGHEST**—Highest value encountered for run-unit within the reported time interval.

**MEAN**—Average value for run-unit within the reported time interval.

**ONLINE**—Shows the LOWEST, HIGHEST, and MEAN RATIOS and COUNTS for all ONLINE run-units during the time interval.

**BATCH**—Shows the LOWEST, HIGHEST, and MEAN RATIOS and COUNTS for all BATCH run-units during the time interval. **SYSTEM**—Shows the LOWEST, HIGHEST, and MEAN RATIOS and COUNTS for all run-units during the time interval.

### Management Highlights/Grand Summary Report

<table>
<thead>
<tr>
<th>CA IDMS - 19.0</th>
</tr>
</thead>
<tbody>
<tr>
<td>CA LOG ANALYZER</td>
</tr>
<tr>
<td>PAGE nn</td>
</tr>
<tr>
<td>mm/dd/yy</td>
</tr>
<tr>
<td>GRAND SUMMARY</td>
</tr>
<tr>
<td>hh:mm:ss</td>
</tr>
<tr>
<td>mm/dd/yy hh:mm - mm</td>
</tr>
<tr>
<td>NYLINE BATCH SYSTEM 0</td>
</tr>
<tr>
<td>RUN UNITS.................</td>
</tr>
<tr>
<td>TOTAL 00 67</td>
</tr>
<tr>
<td>RATIO00 0.00 100.00 0.00 0.00</td>
</tr>
<tr>
<td>/ PAGES READ LOWEST 0.00 0.00 0.00</td>
</tr>
<tr>
<td>/ PAGES READ HIGHEST 1.00 21.00</td>
</tr>
<tr>
<td>/ PAGES READ MEAN 6.45</td>
</tr>
<tr>
<td>/ PAGES READ RECORDS REQUESTED LOWEST 0.00 0.00</td>
</tr>
<tr>
<td>/ PAGES READ HIGHEST 21.00</td>
</tr>
<tr>
<td>/ PAGES READ MEAN 6.78</td>
</tr>
</tbody>
</table>

---

10-Jan-2018 49/240
<table>
<thead>
<tr>
<th>Records Requested</th>
<th>Records becoming current</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lowest</td>
<td>.00</td>
</tr>
<tr>
<td>.27</td>
<td>.00</td>
</tr>
<tr>
<td>.55</td>
<td>.00</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Calc RCDS on Home Page</th>
<th>Lowest</th>
<th>.00</th>
<th>.00</th>
<th>HIGHEST</th>
<th>.00</th>
</tr>
</thead>
<tbody>
<tr>
<td>.00</td>
<td>.00</td>
<td>.00</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.00</td>
<td>.00</td>
<td>.00</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Via RCDS on Owner Page</th>
<th>Lowest</th>
<th>.00</th>
<th>.00</th>
<th>HIGHEST</th>
<th>.00</th>
</tr>
</thead>
<tbody>
<tr>
<td>.00</td>
<td>.00</td>
<td>.00</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>.00</td>
<td>.00</td>
<td>.00</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Management Highlights/Grand Summary Report:

- CA / LOG ANALYZER
- PAGE nn
- GRAND SUMMARY
- MM/DD/YY HH:MM:SS
- Management Highlights/Grand Summary Report (page 1):
- CA IDMS

ONLINE BATCH SYSTEM

Counts:

<table>
<thead>
<tr>
<th>PAGES READ</th>
<th>LOWEST</th>
<th>0</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>HIGHEST</td>
<td>2,008</td>
<td>0</td>
<td>2,008</td>
</tr>
<tr>
<td>MEAN</td>
<td>50.04</td>
<td>.00</td>
<td>50.04</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>PAGES WRITTEN</th>
<th>LOWEST</th>
<th>0</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>HIGHEST</td>
<td>4</td>
<td>0</td>
<td>4</td>
</tr>
<tr>
<td>MEAN</td>
<td>.09</td>
<td>.00</td>
<td>.09</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>PAGES REQUESTED</th>
<th>LOWEST</th>
<th>0</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>HIGHEST</td>
<td>2,434</td>
<td>0</td>
<td>2,434</td>
</tr>
<tr>
<td>MEAN</td>
<td>195.96</td>
<td>.00</td>
<td>195.96</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Calc RCDS on Home Page</th>
<th>LOWEST</th>
<th>0</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>HIGHEST</td>
<td>2</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>MEAN</td>
<td>.04</td>
<td>.00</td>
<td>.04</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Calc RCDS Overflow</th>
<th>LOWEST</th>
<th>0</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>HIGHEST</td>
<td>.00</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Via RCDS on Owner Page</th>
<th>LOWEST</th>
<th>0</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>HIGHEST</td>
<td>2</td>
<td>0</td>
<td>4</td>
</tr>
<tr>
<td>MEAN</td>
<td>.09</td>
<td>.00</td>
<td>.09</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Via RCDS Overflow</th>
<th>LOWEST</th>
<th>0</th>
<th>0</th>
</tr>
</thead>
</table>
The Highlights/Grand Summary Report contains summarized information of system performance and resource consumption accumulated for all System Summaries whenever multiple time intervals are reported.

Management Highlights/Grand Summary Report Fields

These fields make up the Management Highlights/Grand Summary Report. See Figure 2-15 and Figure 2-16.

**REPORT TITLE**--This lists the title of the report.

**INTERVAL**--This line lists the interval and start/stop date and time of report.

**RUN UNITS TOTAL**--Total number of run-units terminated within the reported time interval.
RATIOS--are reported in three values: LOWEST, HIGHEST, and MEAN. For a detailed explanation of the ratios, see Program Summary Report Fields.

- PAGES REQUESTED / PAGES READ
- RECORDS REQUESTED / PAGES READ
- RECORDS REQUESTED / RECORDS BECOMING CURRENT
- CALC RCDS OVERFLOW / CALC RCDS ON HOME PAGE
- VIA RCDS OVERFLOW / VIA RCDS ON OWNER PAGE

COUNTS--IDMS STATISTICS are reported in three values: LOWEST, HIGHEST, and MEAN. For a detailed explanation of CA IDMS statistics, see Program Summary Report Fields.

- PAGES READ
- PAGES WRITTEN
- PAGES REQUESTED
- CALC RCDS ON HOME PAGE
- CALC RCDS OVERFLOW
- VIA RCDS ON OWNER PAGE
- VIA RCDS OVERFLOW
- RECORDS REQUESTED
- RECORDS BECOMING CURRENT
- CALLS TO IDMSDBMS
- FRAGMENTS STORED
- ROOTS OR RCDS RELOCATED

TOTAL I/O--Total number of database input/output operations the run-unit performed during the time interval.

TOTAL CPU (100THS SEC)--Total CPU time needed to execute the run-unit during the time interval specified.

LOWEST--Lowest value encountered for run-unit within the reported time interval.

HIGHEST--Highest value encountered for run-unit within the reported time interval.

MEAN--Average value for run-unit within the reported time interval.
**ONLINE**--Shows the LOWEST, HIGHEST, and MEAN RATIOS and COUNTS for all ONLINE run-units during the time interval.

**BATCH**--Shows the LOWEST, HIGHEST, and MEAN RATIOS and COUNTS for all BATCH run-units during the time interval.

**SYSTEM**--Shows the LOWEST, HIGHEST, and MEAN RATIOS and COUNTS for all run-units during the time interval.

### Management Highlights/Buffer Pool Utilization Report

The Highlights/Buffer Pool Utilization Report distributes the ratio of PAGES REQUESTED to PAGES READ into time brackets for system activity on the calendar days you specify.

<table>
<thead>
<tr>
<th>HOUR</th>
<th>0.00 - 0.99</th>
<th>1.00 - 1.99</th>
<th>2.00 - 3.99</th>
<th>4.00 - 7.99</th>
<th>8.00 - 15.99</th>
<th>16.00 - 18.00</th>
</tr>
</thead>
<tbody>
<tr>
<td>TOTAL</td>
<td>50</td>
<td>6</td>
<td>6.90</td>
<td>5.76</td>
<td>2</td>
<td>56</td>
</tr>
<tr>
<td>7:00</td>
<td>0</td>
<td>1 .50</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1.38</td>
<td>50</td>
<td>0</td>
<td>2</td>
<td>6.19</td>
<td>1.38</td>
<td></td>
</tr>
<tr>
<td>9:00</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1.00</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>2.50</td>
<td>2.50</td>
<td></td>
</tr>
<tr>
<td>22:00</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>12:00</td>
<td>1</td>
<td>0 .50</td>
<td>0</td>
<td>0</td>
<td>0 .50</td>
<td>0</td>
</tr>
<tr>
<td>1.00</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>1.00</td>
<td>1.00</td>
<td></td>
</tr>
<tr>
<td>13:00</td>
<td>4</td>
<td>1 .13</td>
<td>2 .25</td>
<td>1 .13</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>1.00</td>
<td>0</td>
<td>0</td>
<td>8</td>
<td>1.50</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>14:00</td>
<td>3</td>
<td>0 .30</td>
<td>0</td>
<td>0 .00</td>
<td>5 .50</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>0</td>
<td>10</td>
<td>4.95</td>
<td>5.75</td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>92</td>
<td>5 .56</td>
<td>0</td>
<td>2 .01</td>
<td>14 .09</td>
<td>27</td>
</tr>
<tr>
<td>15:00</td>
<td>28</td>
<td>9 .17</td>
<td>163</td>
<td>9.64</td>
<td>.00</td>
<td></td>
</tr>
<tr>
<td>16:00</td>
<td>169</td>
<td>.20</td>
<td>337</td>
<td>11.19</td>
<td>.00</td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>66</td>
<td>0 .00</td>
<td>0</td>
<td>16 .05</td>
<td>30 .09</td>
<td>56</td>
</tr>
<tr>
<td>18:00</td>
<td>1</td>
<td>0 .20</td>
<td>0</td>
<td>0 .00</td>
<td>0 .00</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>5</td>
<td>8.16</td>
<td>10.09</td>
<td></td>
</tr>
<tr>
<td>19:00</td>
<td>1</td>
<td>0 .25</td>
<td>0</td>
<td>0 .00</td>
<td>3 .75</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>4</td>
<td>4.12</td>
<td>5.00</td>
<td></td>
</tr>
<tr>
<td>20:00</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0 .00</td>
<td>0 .00</td>
<td>3</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>3</td>
<td>6.90</td>
<td>6.53</td>
<td></td>
</tr>
<tr>
<td>21:00</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0 .00</td>
<td>2 .50</td>
<td>2</td>
</tr>
<tr>
<td>50</td>
<td>0</td>
<td>0</td>
<td>4</td>
<td>7.47</td>
<td>4.31</td>
<td></td>
</tr>
<tr>
<td>22:00</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0 .00</td>
<td>1 .00</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>6.13</td>
<td>6.13</td>
<td></td>
</tr>
<tr>
<td>TOTAL</td>
<td>272</td>
<td>.50</td>
<td>2</td>
<td>.00</td>
<td>22 .04</td>
<td>62 .11</td>
</tr>
</tbody>
</table>
Management Highlights/Buffer Pool Utilization Report Fields

These fields make up the Management Highlights/Buffer Pool Utilization Report. See Figure 2-17.

**DATE**--The 24-hour period (from midnight to midnight) monitored by CA IDMS Log Analyzer to produce this report.

**HOUR**--Buffer Pool Utilization (BPU) statistics by hour (using the 24-hour clock) for the day the report was generated. This report line will not appear if no run-units terminated during a particular hour.

**The next six columns**--contain six BPU ranges into which all run-units are grouped.

- The number of run-units with a BPU ratio within the given range. The percentage that this number of run-units is of the total run-units.

**TOTAL**--Number of run-units reported during the particular hour.

**MEAN RATIO**--Average ratio value of run-units reported.

**MEDN RATIO**--Median ratio value of run-units reported.

**TOTAL**--Total number of run-units and percentage reported within each range for a given day.

Management Ranking Report

The Ranking Report uses system performance and resource consumption information derived from the original Program Details Report. Statistics are ranked under ABSOLUTE, MEDIAN, or MEAN, depending on the RANKVALU parameter you select.

Unlike the Program Details Report, however, which contains information for all attributes of a run-unit presented in time sequence, the Ranking Report presents the specific run-unit attribute you select, in the sequence you specify. You may also specify whether the ABSOLUTE value of the attribute is to be ranked or whether to rank the MEAN or MEDIAN occurrence of the attribute.

You will get this report if you specify REPORT = RANK on the parameter statement.
### Management Ranking Report (ABSOLUTE Value Type):

<table>
<thead>
<tr>
<th>RANK</th>
<th>PROGRAM</th>
<th>VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>DMLO</td>
<td>NL</td>
</tr>
<tr>
<td>2</td>
<td>MAX</td>
<td>NL</td>
</tr>
<tr>
<td>3</td>
<td>ICD</td>
<td>NL</td>
</tr>
<tr>
<td>4</td>
<td>D000</td>
<td>NL</td>
</tr>
<tr>
<td>5</td>
<td>DCMT</td>
<td>NL</td>
</tr>
<tr>
<td>6</td>
<td>OFF</td>
<td>NL</td>
</tr>
<tr>
<td>7</td>
<td>SIGNON</td>
<td>NL</td>
</tr>
<tr>
<td>8</td>
<td>SIGNOFF</td>
<td>NL</td>
</tr>
<tr>
<td>9</td>
<td>CPUX</td>
<td>NL</td>
</tr>
<tr>
<td>10</td>
<td>MENU</td>
<td>NL</td>
</tr>
<tr>
<td>11</td>
<td>*NULL LV</td>
<td>NL</td>
</tr>
<tr>
<td>12</td>
<td>MKTG</td>
<td>NL</td>
</tr>
<tr>
<td>13</td>
<td>POSI</td>
<td>NL</td>
</tr>
</tbody>
</table>
Management Ranking Report (MEDIAN Value Type):

<table>
<thead>
<tr>
<th>RANK</th>
<th>PROGRAM</th>
<th>VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>IDMSDMLX</td>
<td>17.1</td>
</tr>
<tr>
<td>2</td>
<td>ICDWPL01</td>
<td>15.1</td>
</tr>
<tr>
<td>3</td>
<td>ICDWPL02</td>
<td>9.6</td>
</tr>
<tr>
<td>4</td>
<td>SCHEMA</td>
<td>8.6</td>
</tr>
<tr>
<td>5</td>
<td>DMLO</td>
<td>8.1</td>
</tr>
<tr>
<td>6</td>
<td>RHSCMPUT</td>
<td>3.1</td>
</tr>
<tr>
<td>7</td>
<td>QUED</td>
<td>2.5</td>
</tr>
<tr>
<td>8</td>
<td>IDMSDDDL</td>
<td>2.4</td>
</tr>
<tr>
<td>9</td>
<td>AS$F$</td>
<td>2.0</td>
</tr>
<tr>
<td>10</td>
<td>A$DSOAPCH$</td>
<td>2.0</td>
</tr>
<tr>
<td>11</td>
<td>USSCMPR</td>
<td>1.7</td>
</tr>
<tr>
<td>12</td>
<td>$-$</td>
<td>1.5</td>
</tr>
<tr>
<td>13</td>
<td>ICDPRT01</td>
<td>1.3</td>
</tr>
<tr>
<td>14</td>
<td>USMEXTR</td>
<td>1.3</td>
</tr>
<tr>
<td>15</td>
<td>ADSG</td>
<td>1.1</td>
</tr>
<tr>
<td>16</td>
<td>USBCOMP</td>
<td>0.9</td>
</tr>
<tr>
<td>17</td>
<td>ICDWAR01</td>
<td>0.7</td>
</tr>
<tr>
<td>18</td>
<td>USMVALD</td>
<td>0.7</td>
</tr>
<tr>
<td>19</td>
<td>IDMSDMCL</td>
<td>0.7</td>
</tr>
<tr>
<td>20</td>
<td>OLQ</td>
<td>0.5</td>
</tr>
</tbody>
</table>

Management Ranking Report (MEAN Value Type):
Management Ranking Report Fields

These fields make up the three types of Management Ranking Reports. Note that the three sample reports shown (see Figure 2-18 and Figure 2-43) are identical except for the MEAN, MEDIAN, and ABSOLUTE headings. The report fields defined below are valid for all three versions of the Management Ranking Report.

Start and stop date/time of the report.

Description of report function. The Management Ranking Report reflects the parameter choices you made: the type of ranking process used (RANKHOW); the number of items to be ranked (RANK#); run-unit attribute to be ranked (RANKWHAT), etc.

RANK—Ranking of run-unit or program as determined by CA IDMS Log Analyzer.

PROGRAM—Program name and processing type. BATCH is represented by BTC; ONLINE is represented by ONL.

VALUE—Value of RANKVALU for the run-unit attribute, (seconds, percentage, etc.).

Audit Report

The Audit Report contains a list of the parameters input to CA IDMS Log Analyzer, all informative and error messages dynamically generated by CA IDMS Log Analyzer during execution, and also provide a summary of all processing that occurred.
I003 - PROCESSING OPTIONS

PROCESSING CONTINUES WITH THE VALID REPORT STATEMENTS

I003 - PROCESSING OPTIONS

IDMS RUN-UNITS WILL BE SELECTED

I003 - PROCESSING OPTIONS

DC INTEGRAL RUN-UNITS WILL BE SELECTED

I004 - ARCHIVE RECORDS WILL BE PROCESSED FOR BILLING REPORTS

I005 - EXTRACT PROCESSING

STARTED

mm/dd/yy hh:mm:ss

I006 - ARCHIVE RECORDS PROCESSED FOR THIS TYPE.

55

1

0

2

0

3

4

5

2,131

6

0

1

6

0

2

6

0

3

6

0

4

6

0

5

6

0

6

6

0

7

6

0

8

6

0

9

6

0

10

6

0

11

6

0

12

6

0

13

6

0

14

6

0

15

6

0

16

6

0

17

6

0

18

6

0

19

6

0

Audit Report (page 1):

CA

CA LOG ANALYZER

PA PAGE nn

mm/dd/yy

AUDIT REPORT

hh:mm:ss

I006 - ARCHIVE RECORDS PROCESSED FOR THIS TYPE.

6

0

20

6

0
I006 - ARCHIVE RECORDS PROCESSED FOR THIS TYPE. 6-

I006 - ARCHIVE RECORDS PROCESSED FOR THIS TYPE. 6-

I006 - ARCHIVE RECORDS PROCESSED FOR THIS TYPE. 6-

I006 - ARCHIVE RECORDS PROCESSED FOR THIS TYPE. 6-

I006 - ARCHIVE RECORDS PROCESSED FOR THIS TYPE. 6-

26  2

I006 - ARCHIVE RECORDS PROCESSED FOR THIS TYPE. 6-

I006 - ARCHIVE RECORDS PROCESSED FOR THIS TYPE. 6-

I006 - ARCHIVE RECORDS PROCESSED FOR THIS TYPE. 6-

I006 - ARCHIVE RECORDS PROCESSED FOR THIS TYPE. 6-

I006 - ARCHIVE RECORDS PROCESSED FOR THIS TYPE. 6-

I006 - ARCHIVE RECORDS PROCESSED FOR THIS TYPE. 6-

I006 - ARCHIVE RECORDS PROCESSED FOR THIS TYPE. 6-

I006 - ARCHIVE RECORDS PROCESSED FOR THIS TYPE. 6-

I006 - ARCHIVE RECORDS PROCESSED FOR THIS TYPE. 6-

I006 - TOTAL ARCHIVE RECORDS PROCESSED. 3,763

I008 - EXTRACT RECORDS WERE CREATED FOR PROGRAM 0

I008 - EXTRACT RECORDS WERE CREATED FOR HI-SUM 0

I008 - EXTRACT RECORDS WERE CREATED FOR HI-BPU 0

I008 - EXTRACT RECORDS WERE CREATED FOR RANK 0

I008 - EXTRACT RECORDS WERE CREATED FOR BILLING 68

I009 - TOTAL EXTRACT RECORDS CREATED 68

I005 - EXTRACT PROCESSING ENDED mm/dd/yy hh:mm:ss

I010 - REPORT PROCESSING STARTED mm/dd/yy hh:mm:ss

I011 - A REPORT WAS CREATED FOR YOUR PARAMETER REPORT=BILLING, START=mmddyyhhmm, STOP=mmddyyhhmm, LEVEL=DETAILS, NAME=name, FILE=YES, RUTYPE=SYSTEM, RUNAME=TERM-ID

I013 - TOTAL RECORDS ADDED TO THE BILLING FILE 67

I010 - REPORT PROCESSING ENDED mm/dd/yy hh:mm:ss

I001 - Log Analyzer ENDED mm/dd/yy hh:mm:ss

Audit Report (page 2):
Audit Report Fields

These fields make up the CA IDMS Log Analyzer Audit Report. See Figure 2-21 and Figure 2-22. This report and the fields that appear on the report vary, depending on the parameters input and what happened during execution of CA IDMS Log Analyzer.

Informative messages--Informative messages are indicated by an I at the beginning of the message number. See Messages for more information on informative messages.

Error messages--Error messages are indicated by an E at the beginning of each message number. See Messages for more information on error messages.

PROCESS parameters--The input PROCESS parameters are listed

REPORT parameters--The input REPORT parameters are listed under the message I002.

Extract phase--The extract phase of CA IDMS Log Analyzer is detailed by messages I005, I008, and I009.

Report phase--The report phase of CA IDMS Log Analyzer is detailed by messages I010 and I011

Counts for record types--CA IDMS Log Analyzer reports on records of type 6-28, the log task statistic record type. The log statistic records for CA ADS dialogs are of type 6-34 and type 6-35. CA IDMS Log Analyzer reformats records of type 6-34 so that the new format is like the format of record type 6-28. Records of type 6-35 contain special CA ADS information that is not needed by CA IDMS Log Analyzer.

Processing messages--The processing messages indicate the successful or unsuccessful completion of steps, or list processing errors.

Journal Reports

The first class of printed output is the Journal Reports. These reports contain statistical information and provide users with numerous views of database activity. Journal Reports are divided into four (4) major types:

- Activity Report
- Program Reports
- Special Reports
- Management Reports

Activity Report

The Activity Report summarizes run unit activity over a period of time. The statistics are reported by user-specified time intervals.
Program Reports

The Program Reports feature detailed and summarized statistics by application program within selected time intervals. The level of reporting (details vs. summary) selection of time interval length, and the programs to be analyzed are controlled by the user. The detail Program Reports also indicate the node name of the CA IDMS system within a data sharing group where data sharing is in operation.

Special Reports

The Special Reports provide key information with which to solve existing problems and also identify potential problems within the database environment.

Management Reports

The Management Reports provide summarized highlights and rankings of program performance and resource consumption. See System Output (see page 16) for more detailed information on each of these Journal Reports.

Journal Displays

The second class of printed output is the Journal Displays. These displays allow the user to view specific database changes. In addition, the before and after images of database record modifications are reported by record ID, database key, or program name. Prior to being reported, spanned or fragmented records are recomposed, and compressed records are decompressed. The journal displays character/hexadecimal representations of the database changes and optional subschema view of field changes using the Program Display.

Journal Displays provide the user the ability to effectively monitor changes to sensitive record types, scrutinize performance problems, and to assist in program debugging. Journal Displays are divided into three major types:

- Record Displays
- Database Key Displays
- Program Displays

Record Display

The Record Display shows changes for particular record types during specific time periods.

Database Key Display

The Database Key Display shows changes for particular record occurrences or particular database pages during specific time periods.
Program Display

The Program Display shows database changes effected by particular application programs during specific time periods. The Program Display can be optionally formatted according to the program's subschema view of the database records (i.e., Subschema Display).

For More information about Journal Displays see System Output (see page 16)

Audit Report

The third class of printed output is the Audit Report. This report contains the informative, error, and processing messages generated by CA IDMS Journal Analyzer, and provides a summary of all processing.

For More information about Audit Reports see System Output (see page 16)

Control Parameters

The user has extensive control over CA IDMS Journal Analyzer processing through the use of the following parameters.

- PROCESS
- SUPPRESS
- REPORT
- BYPASS
- DISPLAY
- DLIMITS
- DSUPPS

SUPPRESS Parameter

SUPPRESS is an optional parameter that is used to inhibit the generation of extract records for a specific Journal Report type. The use of this parameter can have a significant impact on the operating efficiency of CA IDMS Journal Analyzer.
REPORT Parameter

The REPORT parameter specifies that a Journal Report is to be printed. This parameter also specifies the criteria by which to select extract records for the report. It is possible to define multiple selection criteria for a particular report type by submitting multiple REPORT parameters for that report type.

DISPLAY Parameter

The operands of the DISPLAY parameter are used to select individual before and after images of database records in the Archive Journal file.

DLIMITS Parameter

The DLIMITS parameter is used in conjunction with the DISPLAY parameter to limit the quantity of display output.

DSUPPS Parameter

The DSUPPS parameter is used in conjunction with the DISPLAY parameter to suppress the display output for selected record IDs. The DSUPPS parameter affects only the PROGRAM Display. You can find more information on these Control Parameters and their operands in Parameters (http://wiki-dev.ca.com/display/IDMS/Parameters)
IDMSDB--DSUPPS Parameter
IDMSDB--DSUPPS Parameter (2)
BYPASS Parameter

The BYPASS parameter allows the User to define up to ten different program names that will be bypassed during Data Extract Processing and Reporting in Journal Analyzer.

CA IDMS Presspack Decompression Support

CA IDMS Presspack is a tool for compression and decompression of CA IDMS records or tables. Its compression efficiency makes it ideal for compressing large volume databases. CA IDMS Journal Analyzer supports decompression of records that were compressed using CA IDMS Presspack and displays them in the Journal Displays.

CA IDMS Journal Analyzer supports compression techniques that use BUILTIN or custom built Data Characteristic Tables (DCTs). To decompress records using custom DCTs, the appropriate DCT load modules must be available in the JNLA loadlib concatenation. The highest version number schema record only for a given schema within the OOAK-S set is processed. This is in line with the current mode of operation. Additionally, CA IDMS Presspack must be used for both record compression and record decompression as indicated in the dictionary SRCALL-040 records.

Journal Displays can be created for both CA IDMS Presspack and CA IDMS Presspack compressed records in the same run.

Need For CA IDMS Journal Analyzer

CA IDMS system represents a company's commitment and investment to the management of the data resource. However, management means more than simply the manipulation of the data resource; it also means having the ability to:

- Measure and evaluate system performance and resource consumption
- Impose system controls with discretion
- Forecast undesirable trends
- Understand the performance capabilities of your data resource

The lack of these management functions can cause your database system to generate many more questions than it can provide answers.

In addition, as a Database Administrator, you need more sources of management and performance related information, or methods of using the readily available sources. For example, the CA IDMS Disk Journal file is the Central Version (CV) run-time audit file which is offloaded to the Archive Journal file. This promising source of information is available, yet needs to be reported or used by existing facilities more efficiently. Therefore, you need a facility that can properly document this valuable information and begin providing you with the necessary answers.
CA IDMS - 19.0

What the CA IDMS database administrator needs is a facility that gathers and combines reliable management and performance data, and reports it in concise logical formats. The facility must be flexible and comprehensive in its ability to assist the Administrators, Programmers/Analysts and Auditors in your organization in controlling the database environment, and realizing the fullest utilization of your data resource.

This unique facility is now available in a software product called CA IDMS Journal Analyzer.

What Is CA IDMS Journal Analyzer?

CA IDMS Journal Analyzer is a comprehensive data analysis facility that enables the CA IDMS user to perform the following activities:

- Audit run-time statistics recorded in the Archive Journal file.
- Sample the audit data from broad surveys of entire programs to precise examinations of individual data fields.
- Evaluate the CA IDMS Central Version (CV) configuration, CV run-time performance, and run-unit consumption of resources.
- Resolve database performance problems.
- Understand performance capabilities of the database system.
- Evaluate user applications and system responsiveness.
- Allows the journal reports and displays to be shown in UTC time.

CA IDMS Extractor Components

CA IDMS Extractor consists of several components, some of which execute in an online environment, and others in a batch environment.

For more information, see the following topics:

- Online Components (see page 67)
- Batch Components (see page 69)
- Customization Macro (see page 70)
- Writing a User Exit Module (see page 71)

Online Components

The CA IDMS Extractor online components are described below. They include:

- Selection Criteria Specification Component
The Selection Criteria Specification Component

The Selection Criteria Specification Component manages your library of selection criteria specifications. Use this component to:

- Describe the existing source database that is used as the basis for generating the target database
- Describe the paths to use when selecting the records from the source database
- Describe the selection criteria to be used when following these paths
- Save your specifications under a specification name of your choice.

The Selection Criteria Specification Component allows you to specify criteria for selection in these ways:

- Select particular areas/records/fields/sets
- Limit the selection of a record type to a specific FROM/TO page range
- Select a record by field value(s), including by CALC key and integrated index SORT key values
- Select a record by direct DBKEY
- Select a group of records within a specified page range, beginning with the first record occurrence within the range
- Select a group of records within a specified page range beginning with the \( nth \) occurrence within the range
- Skip a specified number of records before selecting the next record occurrence
- Specify the number of levels to be extracted in a recursive bill-of-materials structure
- Specify whether you want to extract all owners of extracted recursive records
- Limit the total number of records selected by record type
- Limit the number of records selected in each set occurrence
The Specification Utilities Component

Use the Specification Utilities Component to copy, delete, print, and rename your Selection Criteria Specifications.

The JCL Editor Component

Use the JCL Editor Component to create and modify the JCL used to run the Batch Components of CA IDMS Extractor. It contains sample JCL members that you can quickly modify online to suit the needs of your environment.

The JCL Utilities Component

Use the JCL Utilities Component to copy, delete, print, and rename JCL members.

The JCL Submission Component

Use the JCL Submission Component to submit JCL online to the internal reader for executing the batch extract and batch load components of CA IDMS Extractor.

The User Profile Component

Use the User Profile Component to tailor PF keys to suit your needs. Use the KEYS command to access the User Profile Component and change your PF key settings. The PF key assignments that you make are saved, and are in effect for every session of CA IDMS Extractor until you change them again.

The Online Documentation Component

The Online Documentation Component gives you information about the CA IDMS Extractor screens, commands, the Transfer Facility, available PF keys, and CA IDMS Extractor message text.

The CA Online Documentation Print Utility provided with CA IDMS Extractor allows you to print the information included in the Online Documentation Component. The sample JCL library member GSIPRINT contains the JCL to execute the Online Documentation Print Utility.

The printed version of the online documentation is presented one screen per page and includes page reference indexes for screen options. Characters highlighted in the online documentation appear bolded in the printed version.

For more information about the Online Documentation Print Utility, see the section “Operations.”

Batch Components

The CA IDMS Extractor Batch Components are described below. They include:

- Database Extract Component
- Database Load Component
- Specification Print Utility
The Database Extract Component

The Database Extract Component accesses the specified source database to extract record and set information. The number and type of set and record occurrences selected is governed by both the description you provided during the Selection Criteria Specification process and the structure and information contained within the source database.

Selected information is written to an Extract File and an audit report is produced. The audit report displays the parameters that you submitted and a summary of the information extracted from the source database.

The Database Load Component

The Database Load Component uses the extract file created during the database extract process and loads the target database with information extracted from the source database.

The Specification Print Utility

The Specification Print Utility allows you to print a Selection Criteria Specification that you created and saved using the online Selection Criteria Specification Component.

The JCL Member Print Utility

The JCL Member Print Utility allows you to print a JCL member that you created and saved using the JCL Editor Component.

Customization Macro

The CA IDMS Extractor customization macro gives you the ability to change the following operational parameters:

- The task code used to invoke CA IDMS Extractor.
- The dictionary into which the online documentation modules were loaded at installation.
- The number of entries to allocate for the CA IDMS Extractor set stack.
- From whom a user can copy other JCL members and Selection Criteria Specifications.
- The default for the RETAIN PHYSICAL SEQUENCE OF MEMBER RECORDS IN THE SET? field on the Record Level Selection Criteria screen.
- The default for the EXTRACT ALL OWNERS FOR EXTRACTED RECURSIVE RECORDS? field on the Record Level Selection Criteria screen.
- The default for the BEGIN VIEWING/EDITING IN THE MIDDLE OF A PATH DEFINITION? field on the Specify Database Extract Specification screen.
• Whether to have message NLYZ008 be a warning (W) message or an error (E) message. NLYZ008 is displayed at extract time when a mandatory member is being extracted without its owner. An error message prevents the Selection Criteria Specification from being used.

See the section "Operations" for information on customizing CA IDMS Extractor.

Writing a User Exit Module

CA IDMS Extractor Release gives you the ability to write an assembler user exit module to be called prior to a record's being written to the Extract File. This exit is called by the Database Extract Component when:

• CA IDMS Extractor is walking the source database looking for records to be extracted, and

• A record meets its Record Level Selection Criteria.

You can change the record data, add or delete fields to/from the record data, or prevent the record from being written to the file. We supply the descriptions of the parameters that are passed to your exit module in both Assembler and COBOL formats.

Concepts

This section briefly discusses the procedure for extracting a test database. Also presented are several important concepts about CA IDMS Extractor. Understanding these concepts should help you to more easily extract a test database. Using a test database allows you to overcome the problems encountered by creating a mirror copy of your existing production database to use for testing purposes.

Consider a Test Database

A very large database, by virtue of its size alone, prohibits creating a mirror copy for use as a test database. Logical segregation, to simplify testing and to help isolate problems discovered during the testing phase, is not easily achieved with mirror copies of large databases.

CA IDMS Extractor Implementation

CA IDMS Extractor allows you to create a test database that contains a meaningful sampling of data. Use this extracted data to test the known data interactions encountered by a unit of work in day-to-day processing.

To extract a test database, you must segregate data and all the programs that impact or are impacted by that same data. To do this:
1. Prepare an information model from business methods so that you can identify and group data according to business procedures. If test database implementation is considered in the design phase, this information model could evolve as a result of data flow diagrams.

2. Segregate all the dialogs that access (update or retrieve) a given set of record types:
   a. Select a major dialog and identify all the record types that it uses.
   b. Find all the dialogs that use the same record types used by the dialog in the previous step.

3. Create a subschema with the records and sets identified through the segregation process detailed above. If your existing database is already grouped into functional areas, an existing subschema can be used with CA IDMS Extractor. If no existing subschema is appropriate, create a new one.

4. Use the identified set of dialogs to begin testing.

This implementation method creates multiple test databases. Using this method actually simplifies the testing process and helps to isolate problems encountered.

**Phased Implementation**

CA IDMS Extractor can be used when you do not have a pre-existing database. Phased implementation allows data input programs to be implemented first. While this implementation does not create a test database, it does provide data for the subsequent implementation phases that require test data.

**CA IDMS Extractor Database Paths**

A database path describes the records/sets that CA IDMS Extractor looks at in your source database. For each record that CA IDMS Extractor retrieves in a path, Record Level Selection Criteria that you define are applied. If the record meets the selection criteria, it is written to an Extract File. The records on the Extract File are loaded onto the target database.

**Database Entry Point**

A database entry point defines the beginning of a database path. A database path describes the records/sets that CA IDMS Extractor walks in your source database. You can have one or more entry points or database paths depending on your particular database structure.

An entry point can be a **record**, or a **system-owned integrated index** that is used to retrieve a record:

- **If an Entry Record is used**, the Database Extract Component needs to sweep the area in which the record resides to retrieve the record
- **If an Entry Index is used**, the index is used to retrieve the record without having to perform an area sweep.
However, if you specify CALC or IIX SORT keys by means of Field Level Selection Criteria for an Entry Record or an Entry Index, CA IDMS Extractor retrieves the Entry Record or indexed record by the specified CALC or SORT key value. This specification eliminates the need for CA IDMS Extractor to sweep an area looking for occurrences of the Entity Record and significantly speeds up extract time. See the section "Online Sessions" for more information about the Field Level Selection Criteria screen.

Record Level Selection Criteria

Record Level Selection Criteria define what record occurrences in the source database are going to be extracted. The Criteria is maintained by set type and must be specified for each owner and member record of all sets selected in a path definition. If a record type is retrieved by two different sets, two sets of record level selection criteria must be specified.

For example, in Example Data Structure Diagram, IX-SUBJ-LNAME is selected as an entry index, the PREREQSFOR set is selected walking from owner to member, and the PREREQSARE set is selected walking member to owner. The SUBJECT record is retrieved twice in this path definition: once as member of the Entry Index and once as owner of the PREREQSARE set. Therefore, Record Level Selection Criteria must be defined for the SUBJECT record both as a member of the Entry Index and as owner of the PREREQSARE set. Thus, the Record Level Selection Criteria for the SUBJECT record as it is retrieved as a member of the entry index can be different than the Record Level Selection Criteria for the SUBJECT as it is retrieved as owner of the PREREQSARE set.

Records can be selected from the source database in a variety of ways. Refer to the example of Sample Record Level Selection Criteria Screen.

- **Direct DBKEY**—Extract the record occurrence with the DBKEY specified by the page/line fields. If you do not want to use Field Level Selection Criteria, use Direct DBKEY to select an OOAK Entry Record or to select a record containing specific values in a particular field.

- **Limit Page Range**—Limits the extraction of record occurrences to the page range specified by the from/to page numbers.

- **Limit Records of This Type**—Limit the number of record occurrences extracted from the source database by:
  - Total number of record occurrences that you want CA IDMS Extractor to extract from the source database. This value is the maximum number of record occurrences that CA IDMS Extractor extracts for the record type.
  - Maximum number of record occurrences that CA IDMS Extractor extracts for each set occurrence in which the record participates as a member.

- **Skip Between Records**—Bypasses the number of record occurrences specified by the skip count record type. For example, to extract every fifth record occurrence on the source database, specify a Skip Count of "4".

- **Specify First Within Range**—Tells CA IDMS Extractor to extract the first occurrence of a record type accessed within a record’s specified range. For a CALC record that is a member of a set being walked, the first occurrence retrieved may NOT be the first physical occurrence in the range.
Field values tell CA IDMS Extractor to compare field data in record occurrences to entered Field Level Selection Criteria. You may indicate that a record is to be extracted if the data in the record occurrence is:

- Equal
- Not Equal
- Less Than
- Less Than or Equal
- Greater Than
- Greater Than or Equal
to the data you specify, or:

- Within
- Not Within

a range of values you specify.

Sample Record Level Selection Criteria Screen

```
CAIDMS/DBX Rnn.nn   Record Level Selection Criteria  hh:mm mm/dd/yy
COMMAND ===> 

Entry Record: CRITERIA   From Page: 370000
To Page: 372999
Accessed By: AREA SWEEP of area USV-DATA-AREA
Enter S to Select, D or blank to Deselect selection criteria, press ENTER.
Note: The specification is currently not complete.

Direct DBKEY   Page ===> 0000000000 Line ===> 0000
Limit Page Range From Page ===> 0000370000 To Page ===> 0000372999
S Limit Records of This Type Total Record Count ===> 0000000050
Per Set Occurrence ===> 0000000000
Skip Between Records Skip Count ===> 0000000000
First Within Range
Nth Within Range Which Occurrence ===> 0000000000
Field Level Selection Criteria Display Criteria? ===> Y (Y/N)
Logical Key Selection Criteria Display Criteria? ===> Y (Y/N)
Retain Physical Sequence of Member Records in the Set? ===> Y (Y/N)
Extract All Owners of Extracted Recursive Records? ===> Y (Y/N)
Number of Levels to Extract in the Recursive Structure ===> 00000000
```

In a situation where a record is retrieved by two different sets, you must tell CA IDMS Extractor how many levels of the recursive structure are to be extracted.

Using the same example shown in Sample Record Level Selection Criteria Screen, where the IX-SUBJ-LNAME is selected as an Entry Index, the PREREQSFOR set is selected walking from owner to member, and the PREREQSARE set is selected walking member to owner, the SUBJECT record as owner of the PREREQSARE set would be retrieved a second time. You need to tell CA IDMS Extractor how many occurrences of the SUBJECT record are to be retrieved, as owner of the PREREQSARE set, after each occurrence of the SUBJECT is retrieved by the Entry Index.

A more detailed discussion of extracting from recursive structures is presented below.
When a selected set is accessed by walking from member to owner and the set is a non-sorted set, CA IDMS Extractor must perform an OBTAIN NEXT IN SET until end of set in order to keep the logical integrity of the set intact. For extremely large sets, this walking may impose a tremendous amount of overhead. An option is provided to have CA IDMS Extractor perform an OBTAIN OWNER in the set if you do not require CA IDMS Extractor to maintain logical set integrity for the member records.

Example Data Structure Diagram
Recursive Data Structures

A structure is considered to be recursive if a record type is retrieved by more than one set type in the same path. A Bill-of-Materials (BOM) structure is an example of a recursive structure.

CA IDMS Extractor lets you know that a recursive structure is in the path once you reach the Record Level Selection Criteria screen when a record type is retrieved a second or more time in the same path.

The following fields appear on the Record Level Selection Criteria screen only when a record type is retrieved a second or more time in the same path:

- **EXTRACT ALL OWNERS OF EXTRACTED RECURSIVE RECORDS?===>** is displayed if the current record is recursive and the set is walked member to owner. You must tell CA IDMS Extractor whether to extract all of the record’s owners. This action imposes a great deal of overhead. Enter an N in this field if you are certain that the owner records of all extracted recursive records are extracted because they are retrieved via some OTHER SET in your path definition. Otherwise, your target database will NOT be complete.

- **NUMBER OF LEVELS TO EXTRACT IN THE RECURSIVE STRUCTURE===>** In most recursive structures, this value should be set to zero (0), except for true BOM structures. Your situation may dictate otherwise. You must be aware, however, of the additional overhead needed to process recursive structures.

In the simplified data structure diagram shown below, the structure in Example 1 contains a BOM structure. In both examples, Record A is an Entry Record and SETS AB and BC1 are selected walking from owner to member. After defining Record Level Selection Criteria for record C as a member of the BC1 set, SET BC2 is selected walking from member to owner in Example 1. Set CA is selected walking owner to member in Example 2.

At this point, CA IDMS Extractor determines that Record B in Example 1 and Record A in Example 2 participate in recursive structures and are regarded as recursive records. These records/sets define the beginning of the recursive structure. You must tell CA IDMS Extractor how many times the recursive structure is to be walked once the beginning of the recursive structure is found.

In these examples, once Record B is retrieved through set BC2 and Record A is retrieved through set CA, you must tell CA IDMS Extractor how many more times sets selected in the path after Record B in Example 1 and Record A in Example 2 are to be walked. Specifying zero (0) in the NUMBER OF LEVELS TO EXTRACT IN THE RECURSIVE STRUCTURE field tells CA IDMS Extractor to not retrieve any more records/sets in the path at that point.

For each recursive record that is extracted, CA IDMS Extractor also extracts all owners of selected sets in which the recursive record is currently a member. In these examples, after Record B is retrieved and extracted through set BC2, CA IDMS Extractor extracts the owning Record A even though set AB is being walked from owner to member.

You may specify a value for NUMBER OF LEVELS TO EXTRACT IN THE RECURSIVE STRUCTURE that is greater than the actual number of levels present in your database. CA IDMS Extractor uses this value as the maximum number of levels that it traverses.
Creating CA IDMS Extractor Database Paths

There are several ways of creating a database path in CA IDMS Extractor:
- Manual Selection of Records/Sets (see page 77)
- Hierarchical Selection (see page 78)
- Network Selection (see page 79)
- Deselecting Previously Selected Records/Sets (see page 79)
- Terminating Record Level Selection Criteria (see page 80)

Manual Selection of Records/Sets

Manually selecting sets in a path is an iterative process that consists of selecting sets to be included in the path, then defining Record Level Selection Criteria for owner and member records of the selected sets.
To begin the process, you first select an Entry Record or Entry Index to define the beginning of a database path. CA IDMS Extractor then displays the Record Level Selection Criteria screen for the Entry Record or Entry Index record. You define the Record Level Selection Criteria for the record. CA IDMS Extractor then displays the Path-Record Set Selection list screen that shows the sets that may be walked from the entry record. Sets are displayed only if they are not already selected in a path for the current specification.

If you select any sets from the Path-Record Level Selection Criteria screen, and the selected set is walked from:

- **Owner to Member** -- The Record Level Selection Criteria screen is displayed for the **member** record. If the set is a multi-member set, a Record Level Selection Criteria screen is displayed for each member.

- **Member to Owner** -- The Record Level Selection Criteria screen is displayed for the **owner** record.

Record Level Selection Criteria must be specified for each owner/member of all selected sets.

The process of selecting sets to be included in a path and defining Record Level Selection Criteria for all owner and member records of selected sets continues until:

- No sets are selected from the Path-Record Set Selection List screen, or all sets accessible from path records are already included in a path; and

- Record Level Selection Criteria have been defined for all owner and member records of selected sets.

### Hierarchical Selection

With a hierarchical selection, CA IDMS Extractor **automatically** selects all records and sets accessible walking **down** the structure, that is, from **owner to member**. When you make a hierarchical selection of a record or index, CA IDMS Extractor marks it as an Entry Record or Entry Index. All members of sets owned by the Entry Record or Entry Indexed record, which do not already participate in a path and which are not indexed by an Entry Index, are selected in the path. All selected records that are themselves owners of sets are subject to the same selection process as the Entry Record.

There is an exception to the hierarchical selection process regarding a Bill-Of-Material (BOM) structure. The AUTOMATIC member of the BOM set is selected walking from the BOM record to the junction record. The MANUAL member of the BOM set is selected walking from the junction record to the BOM record.

The Record Level Selection Criteria from all records selected in this manner indicate to extract **all** record occurrences. You may modify the selection criteria of any or all of these records to limit the number of records extracted.
Network Selection

With a network selection, CA IDMS Extractor automatically selects all records and sets accessible in the entire structure. When you make a network selection of a record in index, CA IDMS Extractor marks it as an Entry Record or Entry Index. All owners of sets of which the Entry Record or Entry Indexed record is a member and, which do not already participate in a path and which are not indexed by an Entry Index, are selected in the path. All selected records that are themselves owners or members of sets are treated as if it were an Entry Record.

There is an exception to the network selection process regarding a Bill-Of-Material (BOM) structure. The AUTOMATIC member of the BOM set is selected walking from the BOM record to the junction record. The MANUAL member of the BOM set is selected walking from the junction record to the BOM record.

The Record Level Selection Criteria for all records selected in this manner indicate to extract all record occurrences. You may modify the selection criteria of any or all of these records to limit the number of records extracted.

You should verify that sets are walked in the manner in which you want them walked, that is, from owner to member or member to owner. CA IDMS Extractor may select a set as being walked from member to owner when you really want the set walked from owner to member. If this is the case, you must deselect the set and then select it again using a manual selection.

For a data structure containing recursive records (records that are accessed by more than one set type in the same path), CA IDMS Extractor sets the number of levels to extract to zero. You may need to modify this value. Refer to Recursive Data Structures for more information.

Deselecting Previously Selected Records/Sets

To remove records/sets from a path definition or to delete an entire path definition, you may deselect a path set, an Entry Record or Entry Index, or an area. Deselection of:

- An Entry Record/Entry Index deletes the entire path definition.
- An area deselects all Entry Records residing in the area.
- A path set deselects all sets in the path accessed after the deselected set. The deselection process stops at recursive records. However, sets selected in the path after recursive records are not deselected.

Deselection causes all Record Level Selection Criteria to be removed from the record, including all Field Level Selection Criteria. Once deselected, a record or set can be selected again.
Terminating Record Level Selection Criteria

When you tell CA IDMS Extractor to terminate the specification for the current specification session, by typing the END command or pressing the End PF key at the Record Level Selection screen, the session may not be complete. There may be record types in selected path-record sets for which Record Level Selection Criteria have not been specified. CA IDMS Extractor flags the specification so that you are not able to use it without completing it.

CA IDMS Extractor indicates whether or not the specification is complete at the time the Record Level Selection Criteria screen is displayed.

JCL Editing Commands

This section provides a guide to the JCL editing commands that are available in the CA IDMS Extractor JCL editor. This section describes each command and its syntax.

Program Function Keys

PF keys are set to many frequently used commands. Therefore, you can enter a command from any position on the Edit screen with one keystroke. In addition, the PA1, PA2, and CLEAR keys are set to redisplay the screen. Use the KEYS primary command to change EDITOR PF key settings.

To execute a single command set for a PF key, press that key. The command executes when you press the PF key.

Editing Commands

There are the following types of editing commands:

- Scroll Options
- Primary Commands
- Line Commands

Scroll Options

Scroll options are used to determine how many lines or columns of the JCL to scroll up, down, right, or left when using a primary command or a PF key.
Primary Commands

Primary commands are used to:

- Locate the desired line of the JCL
- Find the next occurrence of a string
- Reset the screen display to remove all line commands, column markers, and extraneous messages
- Cancel changes made with the editor to the JCL
- Turn the CAPS Mode on or off
- Display the time and date
- Terminate EDITOR session, save changes made to the JCL, and return to CA IDMS Extractor
- Save changes made to the JCL and remain in EDITOR session

Line Commands

Line commands are used to:

- Copy source lines within the JCL
- Move source lines within the JCL
- Specify the location at which source lines are to be copied or moved
- Repeat source lines in the JCL
- Delete source lines
- Insert blank source lines
- Display a line with column markings across the screen

Conventions, Rules and Syntax

Be sure to review these exhibits before you begin your first edit session:

- **Exhibit 6.4** -- Notation Conventions
- **Exhibit 6.5** -- Command Syntax Rules
- **Exhibit 6.6** -- Command Syntax Summary
• **Exhibit 6.7 -- Command Syntax Summary**

Scroll options, Primary Commands, Line Commands and Key Settings are covered in detail in the rest of this section.

**Functions**

The command and key functions are summarized in these exhibits:

- **Exhibit 6.8 -- Summary of Primary Commands**
- **Exhibit 6.9 -- Summary of Scroll Options**
- **Exhibit 6.10 -- Summary of Line Commands**
- **Exhibit 6.11 -- Summary of Key Settings**

<table>
<thead>
<tr>
<th>Example</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>FIND string</td>
<td>Variables appear in lowercase. you substitute an appropriate value for each variable.</td>
</tr>
<tr>
<td>UP [number-of-lines]</td>
<td>Brackets indicate optional clauses.</td>
</tr>
<tr>
<td>/ UP \</td>
<td>Braces enclose two or more options. You select one of them.</td>
</tr>
<tr>
<td>&lt; DOWN &gt;</td>
<td></td>
</tr>
<tr>
<td>\</td>
<td></td>
</tr>
</tbody>
</table>

*Exhibit 6.4: Notation Conventions*

<table>
<thead>
<tr>
<th>Item</th>
<th>Rule</th>
</tr>
</thead>
<tbody>
<tr>
<td>Order of Commands</td>
<td>You must enter a B (before) or an A (after) line command in conjunction with the C (copy) and M (move) line commands to indicate where to copy or move the lines.</td>
</tr>
<tr>
<td>Entering Blanks in Commands</td>
<td>Blanks (character spaces) are ignored in line command sequences, so you can enter blanks between a command and a value without affecting processing. You must enter at least one blank (character space) between a Primary Command and a primary command value field. You cannot embed blanks in a keyword.</td>
</tr>
</tbody>
</table>

*Exhibit 6.5: Command Syntax Rules*
Primary Commands

```
UP    \{ number-of-lines \}
DOWN \{ MAX \}

RIGHT \{ number-of-columns \}
LEFT \{ MAX \}
```

LOCATE line-number

FIND \{ string \}

REPLACE \{ string \} \{ replacement-string \} \{ ALL \}

RESET

CANCEL

END/RETRY

NULLS \{ ON \} \{ OFF \}

CAPS \{ ON \} \{ OFF \}

---

IDMSDB--Functions

Exhibit 6.6: Command Syntax Summary

Command Syntax Summary

**Scroll Options**

```
\{/ PAGE \} \(<\ \text{HALF}\ \>
\}
\CSSR\ \number-of-lines \\
```

**Line Commands**

```
\{/ B \} \(<\ A \ \>\ number-of-times \\
\}
```

```
\{/ C \} \(<\ \text{number-of-lines} \ \>
\}
```
Key Settings

PF1/PF13    HELP (currently unavailable)
PF2/PF14    END/RETRY
PF3/PF15    END/RETRY
PF4/PF16    CANCEL
PF5/PF17    RFIND
PF6/PF18    RCHANGE
PF7/PF19    UP
PF8/PF20    DOWN
PF9/PF21    RESET
PF10/PF22   LEFT
PF11/PF23   RIGHT
PF12/PF24   ENTER
PA1         RESHOW
PA2         RESHOW
CLEAR       CLEAR

Entering Commands

The following are descriptions of where commands are entered:

- **Scroll Options** - Enter these options at the far right side of the second line on the screen, after the word SCROLL.

- **Primary Commands** - Enter these commands at the left side of the second line, after the word COMMAND. This field is called the COMMAND line.

- **Line Commands** - Enter these commands in the line number fields at the left of the source.

Notation Conventions

<table>
<thead>
<tr>
<th>Example</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>RESet</td>
<td>Keywords appear in mixed case.</td>
</tr>
<tr>
<td></td>
<td>The minimum required portion of each keyword appears in uppercase.</td>
</tr>
<tr>
<td>Find string</td>
<td>Variables appear in lowercase. You substitute an appropriate value for each variable.</td>
</tr>
<tr>
<td>Up [number-of-lines]</td>
<td>Brackets indicate optional clauses or commands.</td>
</tr>
<tr>
<td>Ɪ</td>
<td>Braces enclose two or more options. Select an option.</td>
</tr>
</tbody>
</table>
Command Syntax Rules

<table>
<thead>
<tr>
<th>Item</th>
<th>Rule</th>
</tr>
</thead>
<tbody>
<tr>
<td>Order of Commands</td>
<td>You must enter a B (before) or an A (after) line command in conjunction with the COPY primary command and the C (copy) and M (move) line commands to indicate where to copy or move the lines.</td>
</tr>
<tr>
<td>Entering Blanks in</td>
<td>Blanks (character spaces) are ignored in line command sequences, so you can enter blanks between a command and a value without affecting processing.</td>
</tr>
<tr>
<td>Commands</td>
<td>You must enter at least one blank (character space) between a primary command and a primary command value. You cannot embed blanks in a keyword.</td>
</tr>
<tr>
<td>Command Stacking</td>
<td>You can enter multiple primary and line commands. Primary commands must be separated by a semicolon (;).</td>
</tr>
</tbody>
</table>

Line Commands 1

Line commands are entered in Edit Mode with the cursor positioned to the left of the source lines, in the line number fields. To use a line command, type over the line numbers.

Entering Line Commands

Line commands are entered within the line number at the left of the line data. A line command is considered to be any characters entered at or to the left of the cursor in the line sequence number fields.

How to Use Line Commands

If you wanted to repeat the line 10 times, here is how the line would appear:

000003 Before entering R (repeat) command R10003 After entering R (repeat) command

For the EDITOR to read the command as R10:

Type ‘R10’ in the line number field and press the ENTER key

- Position the cursor immediately after R10 (type ‘R1’ and move the cursor to the right one position) and press the ENTER key.
Entering Line Commands

When you enter a line command, the editor looks for the right-most value that you changed in the line number field, and it looks at the position of the cursor. For example, here is how the line number field looks before and after specifying with the R (repeat) command that line 3 is to be repeated 12 times.

*Before* entering R command 000003 *After* entering R command R12003

In this example, the editor sees that the right-most changed value is the 2, and everything to the left of and including the 2 is read as the line command (R12).

If you wanted to repeat the line ten times instead of 12 times, here is how the line number field would appear:

*Before* entering R command 000003 *After* entering R command R10003

For the editor to read the command as R10 (and not, for example, R1 or R100), the cursor must be positioned immediately after R10 in the line number field (at the position underlined above) when you press the ENTER key.

If you want to press the ENTER key with the cursor positioned elsewhere on the screen, you can key in a blank character space after the R10 command. The line number field would look like this:

*After* entering R command R10 03

Or, you could press the ERASE EOF (erase end of field) key after keying in the R10.

Note that the value you substitute for a numeric variable in a line command can be up to five digits in length.

B (before) and A (after) Commands

The B (before) and A (after) commands are used in conjunction with the C (copy) and M (move) line commands. Use either the B command or the A command in the line number field to indicate the location before or after the source line where lines are to be copied or moved. The syntax of the two commands is:

/ B \< A > [number-of-lines]\ /

where:

number-of-lines specifies that you want a number of copies of the line(s) or the module copied or moved.

⚠️ **Note:** Blanks are not a valid option; ensure that the command line contains no blanks.
C (copy) Command

\textbf{C[number-of-lines]}

Use the C (copy) line command to copy one line or block of lines. The B (before) and A (after) line commands are used to specify the destination of the line or block to be copied. No other line commands can be entered on the lines to be copied.

- \textbf{number-of-lines}
  
specifies the number of lines to be copied. The default is 1.

- \textbf{C}
  
  Specifies a single line to be copied.

- \textbf{Cn}
  
  Specifies the first of \textit{n} lines to be copied.

- \textbf{CC...CC}
  
  Specifies the first and last lines of a block of lines to be copied.

Rules for Using the Copy Line Command

When using the C \textit{number-of-lines} or the CC form of the command, you cannot enter any other commands on the lines being copied.

- Each CC must be paired with another CC.

- You must pair a B (before) or an A (after) line command with every C or pair of CC commands.

How to Use the C (copy) Command

Using the C command, you can specify:

- A single line to be copied (that line marked).

- A number of lines to be copied (the first line marked).

- A block of lines to be copied (the first and last lines marked).

Here is the syntax for copying a single line:

\textit{C}

The syntax for copying a specified number of lines is:

\textit{Cnumber-of-lines}

To copy a block of lines, mark both the first line and the last line of the block with:

\textit{CC}
Rules for C (copy) Command

When using the C number-of-lines or the CC form of the command, you cannot enter any other commands on the lines being copied.

A CC must be paired with another CC.

You must pair a B (before) or an A (after) command with every C or pair of CC commands.

M (move) Command

M[number-of-lines]

Use the M (move) command to move a line or block of lines. The B (before) or A (after) line commands are used to specify the destination of the lines to be moved. No other commands can be entered on the lines to be moved.

- number-of-lines
  Specifies the number of lines to be moved. The default is 1.

- M
  Specifies a single line to be moved.

- Mn
  Specifies the first of n lines to be moved.

- MM...MM
  Specifies the first and last lines of a block.

Rules for Using the Move Line Command

When using the M number-of-lines or the MM form of the command, you cannot enter any other commands on the lines being moved.

- Each MM must be paired with another MM.

- You must pair a B (before) or an A (after) line command with every M or pair of MM commands.

How to Use the M (move) Command

Using the M command, you can specify:

- A single line to be moved (that line marked).

- A number of lines to be moved (the first line marked).

- A block of lines to be moved (the first and last lines marked).

Here is the syntax for moving a single line:

M

The syntax for moving a specified number of lines is:
To move a block of lines, mark both the first line and the last line of the block with:

\( \text{MM} \)

**Rules for M (move) Command**

When using the M\{number-of-lines\} or the MM form of the command, you cannot enter any other commands on the lines being moved.

An MM must be paired with another MM.

You must pair a B (before) or an A (after) command with every M or pair of MM commands.

**R (repeat) Command**

\[ \text{R[number-of-times]} \]

Use the R (repeat) line command to repeat a line or block of lines directly after the last line to be repeated.

- **number-of-times**
  Specifies the number of times a line or block of lines is repeated. The default is 1.

- **R**
  Specifies a single line to be repeated.

- **Rn**
  Specifies a single line to be repeated \( n \) times.

- **RR**
  Specifies the first and last lines of a block to be repeated one time.

**Rules for Using the R (repeat) Line Command**

Pair each RR \{number-of-lines\} with another RR \{number-of-lines\} to complete a block command.

- If \{number-of-lines\} specified in the RR block command differ, the greater number is used.
- No other line commands can be used on lines being repeated.

**How to Use the R (repeat) Command**

You can specify:

- A single line to be repeated (that line marked).
- A single line to be repeated a number of times (that line marked).
- A block of lines to be repeated (the first and last lines marked).
- A block of lines to be repeated a number of times (the first and last lines marked).
Here is the syntax for repeating a single line:

\( R \)

The syntax for repeating a single line a specified number of times is:

\( R \text{number-of-times} \)

To repeat a block of lines once, mark both the first line and the last line of the block with:

\( RR \)

To repeat a block of lines a number of times, mark both the first line and the last line of the block with:

\( RR \text{number-of-times} \)

You can enter the number of times the block is to be repeated on either the first line or the last line of the block. If you specify a number of times on both lines, the number on the last line will be used.

Rules for R (repeat) Command

When using the \( R \text{number-of-times} \), the \( RR \), or the \( RR \text{number-of-times} \) form of the command, you cannot enter any other commands on the lines being repeated.

An RR must be paired with another.

TABS Command 1

TABS

Typing TABS in the line number field to view the current tab settings. You can also use the TABS command to change the tabs by overstriking the current setting (indicate by tab character) with the new positions you choose.

The TABS line may be deleted from the display by the D (delete) line command or the RESET primary command.

D (delete) Command

D[number-of-lines]

Use the D (delete) command to delete a line or block of lines. No other line commands can be entered on the lines to be deleted.

- number-of-lines
  Specifies the number of lines to be deleted.

- D
  Specifies a single line to be deleted.

- Dn
  Specifies the first of \( n \) lines to be deleted.
- **DD...DD**
  Specifies the first and last lines of a block of lines to be deleted.

### Rules for Using the D (delete) Line Command

- When using the D *number-of-lines* or the DD form of the command, you cannot enter any other commands on the lines being deleted.
- Each DD must be paired with another DD.

### How to Use the D (delete) Command

Using the D command, you can specify:

- A single line to be deleted (that line marked).
- A number of lines to be deleted (the first line marked).
- A block of lines to be deleted (the first and last lines marked).

Here is the syntax for deleting a single line:

```
D
```

The syntax for deleting a specified number of lines is:

```
D number-of-lines
```

To delete a block of lines, mark both the first line and the last line of the block with:

```
DD
```

### Rules for D (delete) Command

- When using the D *number-of-lines* or the DD form of the command, you cannot enter any other commands on the lines being deleted.
- A DD must be paired with another DD.

### X (exclude) Command

```
X[number-of-lines]
```

Use the X (exclude) command to exclude lines from the display.

*number-of-lines*

Specifies the number of lines excluded.

```
X
```

Specifies a single line to be excluded.

```
Xn
```
Specifies the first of n lines to be excluded.

XX...XX

Specifies the first and last lines of a block of lines to be excluded.

I (insert) Command

I[number-of-lines]

Use the I (insert) command to insert blank lines after the line in which the I command is entered. The I command is not used with the A (after) and B (before) line commands. If no data is typed on an inserted line, the blank inserted line is deleted from the display after the ENTER key is pressed or the RESET, UP, or DOWN primary command is entered.

number-of-lines

Specifies the number of lines to be inserted. The default is 1.

COLS (columns) Command

COLS

The COLS command displays a line with the column markings for you to use as a reference. This line is for reference purposes only. It is not given a line number and is not saved with the text.

The column markings line appears before the line in which you enter the COLS command.

To remove the COLS line from the display, use the D (delete) line command or the RESET primary command.

A (after) Command

A

Use the A (after) line command in conjunction with the C (copy) and M (move) line commands.

B (before) Command

B

Use the B (before) line command in conjunction with the C (copy) and M (move) line commands.

BNDS (bounds) Command

BNDS

The BNDS command displays and allows changes to the current boundary settings. The bounds line is displayed at the line where you entered the command.

Change the current bounds setting by using the < character to define the left bound and the > character to define the right bound.
To remove the bounds line from the display, use the D (delete) line command or the RESET primary command.

Primary Commands 1

Primary commands are entered on the second line of the Edit screen after the word COMMAND. You can enter more than one primary command at a time. Use the following syntax:

command;command

UP and DOWN Commands

The UP (scroll up) and DOWN (scroll down) commands are used to display source lines above or below your current view. The amount you scroll is determined by the Scroll Option setting. The setting can be overridden at any time. To override the existing setting, use the following syntax:

/ UP \ / number-of-lines \< DOWN > < MAX >\ / /

Where:

number-of-lines specifies that you want to scroll up or down a specific number of lines.

MAX specifies that you want to scroll to the first or last full page of text.

Here are the three basic formats you can use:

/ UP \< DOWN >\ /

Use either UP or DOWN alone to scroll the current scroll setting.

/ UP \< DOWN > number-of-lines\ /

Use UP or DOWN with a number to override the current scroll setting.

/ UP \< DOWN > MAX\ /

Use UP or DOWN with MAX to scroll to either the beginning (UP) or the end (DOWN) of the module.

Default UP keys: PF7, PF19

Default DOWN keys: PF8, PF20.

RIGHT and LEFT Commands

The RIGHT (scroll right) and the LEFT (scroll left) commands are used to display source lines to the right or left of your current view. The amount you scroll is determined by the scroll option setting. The setting can be overridden at any time. To override the existing setting, use the following syntax:

/ RIGHT \ / number-of-lines \< LEFT > < MAX >\ / /

where:

number of columns specifies that you want to scroll a specific number of columns to the right or to the left.
MAX specifies that you want to scroll to either the far right or far left of the text.

Here are the three basic formats you can use:

```
/ RIGHT \< LEFT >\ /
```

Use RIGHT or LEFT alone to scroll the current scroll option setting.

```
/ RIGHT \< LEFT > number-of-columns \ /
```

Use RIGHT or LEFT with a number to override the current scroll setting.

```
/ RIGHT \< LEFT > MAX \ /
```

Use RIGHT or LEFT with MAX to scroll to either the far right or far left of the module.

Default RIGHT keys: PF11, PF23

Default LEFT keys: PF10 or PF22.

**LOCATE Command**

**Locate line-number**

Use the LOCATE command to move the display to a specific source line or to the beginning or the end of the JCL.

- **line-number**
  Specifies the number of the line to which you want to move. The line you specify is the top line displayed on the screen.

**Using the LOCATE Command**

To move to a specific line, you specify the line number of the line you want displayed.

To move to the beginning of the JCL, you can specify 0 as the line number, and the first line of the JCL is the top line displayed.

To move to the end of the JCL, you can specify the last line number or any larger number, and the last line of the JCL is displayed on the top line. For example, if the last line of the JCL is numbered 307 and you use 999, line number 307 is the top line displayed.

**How to Use the LOCATE Command**

To scroll to a specific line, you specify the line number of the line you want displayed.

To scroll to the beginning of the module, you can specify 0 as a line number, and the first line of the module will be the top line displayed.

To scroll to the end of the module, you can specify the last line number or any larger number, and the last line of the module will be the top line displayed. For example, if the last line of the module is numbered 307 and you use 999, line number 307 will be the top line displayed.
MEMORY Command

MEMory [ '< STATIC > ]
[ ' DYNAMIC ]

Internal storage is determined by the MEMORY command.

- STATIC
  Specifies to obtain storage one time and track until the end of the session.

- DYNAMIC
  Specifies to obtain new storage and free it each time the EDITOR driver module is called.

FIND and RFIND Commands

Use the FIND command or the RFIND (repeat find) command to search for a string in the module.

The editor begins searching at the position of the cursor when you ENTER the command, and it
searches downward until the string is found. If the cursor is on the Command Line when you ENTER
the command, the editor begins searching at the top line displayed.

The cursor appears at the beginning of the string, if the string is found. Here is the FIND command
syntax:

FIND string

Rules for FIND Command

If the string has embedded blanks, enclose the string in either single or double quotation marks. For
example:

FIND  program name'
FIND  "program name"

- If the string consists of a single asterisk (*), enclose the asterisk in quotation marks. A double
  asterisk does not require any quotation marks. For example:

FIND  *
FIND  **

- Quotation marks are optional for all other strings. If you decide to enclose a string which contains
  a single quotation mark, enclose the string with double quotation marks. If you decide to enclose
  a string which contains double quotation marks, enclose the string with single quotation marks.
  For example:

FIND  "They're"

The RFIND command repeats the last FIND command that was entered. Here is the syntax:

RFIND

Default RFIND keys: PF5, PF17.
If the cursor is on the top line when you enter the RFIND command, the editor will search the entire file. When it reaches the end of the file after several finds, the message line will state BOTTOM OF DATA REACHED. Entering RFIND again will return you to the top of the file. If, however, the string does not exist in the file when you enter the RFIND command, the message line will state NO CHARACTERS "string" FOUND. Entering the RFIND command again has no effect.

To selectively change strings, use the RFIND PF key in conjunction with the RCHANGE PF key. See the next subsection on CHANGE and RCHANGE commands for a description.

**CHANGE and RCHANGE Commands**

Use the CHANGE command or the RCHANGE (repeat change) command to search for and change the next occurrences of a string in the module.

The editor begins searching at the position of the cursor when you ENTER the command, and it searches downward until the string is found. If the cursor is on the Command Line when you ENTER the command, the editor begins searching at the top line displayed.

If the string is found, it is changed to the replacement string. Here is the CHANGE command syntax:

```
/ / / \CHANGE [ALL] < string > < replacement-string > [ALL] \ * \ * / 
```

where:

- **ALL** specifies that the string is to be replaced with the replacement string throughout the module (from beginning to end). ALL can be used in either position shown above. It can be used only once in the CHANGE command.
- **string** specifies the string to be found and replaced.
- **asterisk**, when used as a string value, specifies the string value from the last FIND or CHANGE command entered. When an asterisk is used as a replacement string value, it specifies the replacement string value from the last CHANGE command entered.
- **replacement-string** specifies the string to replace the first string specified.

The rules regarding the CHANGE and RCHANGE commands are on the following pages.

**Rules for CHANGE Command**

If a string has embedded blanks, enclose the string in either single or double quotation marks. For example:

```
CHANGE
program name'
program name'
CHANGE "program name" "program name"
```

- If the string consists of a single asterisk (*), enclose the asterisk in single quotation marks. A double asterisk does not require any quotation marks. For example:

```
CHANGE
*'
comments'
CHANGE **
comments'
```
If a string is the word ALL, enclose the string in quotation marks.

Quotation marks are optional for all other strings. If you enclose a string which contains a single quotation mark, enclose the string with double quotation marks. If you enclose a string which contains double quotation marks, enclose the string with single quotation marks. For example:

```
CHANGE "They're" "He's"
```

The RCHANGE command repeats the last CHANGE command that was entered. The syntax is:

```
RCHANGE
```

Default keys: PF6, PF18.

Using the RCHANGE and RFIND PF keys to Selectively Change Strings

You can use the RFIND PF key in conjunction with the RCHANGE PF key to selectively change strings. Here are two sample sequences:

**First Sample Sequence**

```
CHANGE WORK-NAME-1 WORK-NAME-2
```

ENTER key

- RFIND key
- RCHANGE key
- RFIND key
- RFIND key

In Step 1, you ENTER the CHANGE command to change the next occurrence of WORK-NAME-1 to WORK-NAME-2.

In Step 2, you want to find the next occurrence of WORK-NAME-1, but you are not sure if you will want to change the string. By pressing the RFIND key, the occurrence of WORK-NAME-1, which was specified in the CHANGE command during Step 1, will be found.

In Step 3, you want to change the occurrence of WORK-NAME-1 that was found during Step 2 to WORK-NAME-2. By pressing the RCHANGE key, the occurrence will be changed.

In Step 4, you press the RFIND key to find the next occurrence of WORK-NAME-1. This time you do not want to change the string, so instead of pressing the RCHANGE key, you press the RFIND key again. The next occurrence of WORK-NAME-1 will be found.

**Second Sample Sequence**

```
CHANGE WORK-NAME-1 WORK-NAME-2
```

RFIND key

- RFIND key
RCHANGE key

In Step 1, you want to find the next occurrence of WORK-NAME-1, but you are not sure if you will want to change it to WORK-NAME-2. By keying in the CHANGE command and pressing the RFIND key instead of the ENTER key, the RFIND will be executed. The next occurrence of WORK-NAME-1, which was specified in the CHANGE command, will be found.

In Step 2, you decide that you do not want to change the string that was found during Step 1, so you press the RFIND key. The next occurrence of WORK-NAME-1 will be found.

In Step 3, you want to change the string that was found during Step 2, so you press the RCHANGE key. This changes WORK-NAME-1 to WORK-NAME-2.

RESET Command

RESet

Use the RESET command to clear the display of any line commands, column markers, or extraneous messages.

The default keys are PF9 and PF21.

RESHOW Command

RESHOW

Use the RESHOW command to redisplay the original contents of a screen. This command is only valid when you have typed a screen of data but have not pressed the ENTER key.

⚠️ Note: If you use the RESHOW command, you overlay the current screen with the previous screen.

RFIND Command

RFIND

The RFIND command repeats the last FIND command that was entered.

The RFIND search begins at the position of the cursor. When it reaches the end of the file, it reaches the bottom and the message line states, BOTTOM OF DATA REACHED. Entering RFIND again resumes the search at the top of the file. Then if the string is not found in the file, the message line displays NO CHAR: string Found. Entering the RFIND command has no effect.

To selectively change strings, use the RFIND PF key in conjunction with the RCHANGE PF key.

The default keys are PF5 and PF17.
RIGHT Command

Right [number-of-columns]

The RIGHT (scroll right) command scrolls the current display to the right the specified number of columns. If the number parameter is blank, the scroll options are used.

The default keys are PF11 and PF23.

SAVE Command

SAVE

Use the SAVE command to save the current JCL (if changed). The EDITOR session remains active.

TABB Command

TABB

Use the TABB (tab backward) command to move to the previous tab setting when TABS Mode is ON.

This command is invoked more efficiently if you assign it a PF key value.

TABF Command

TABF

Use the TABF (tab forward) command is used to move to the next tab setting when TABS Mode is ON.

This command is invoked more efficiently if you assign it a PF key value.

TABS Command

TABS [ / ON < OFF ]

The TABS command sets software tabbing. Use the commands TABF (tab forward) and TABB (tab backward) to move a tab setting within the text. To customize tab settings, use the TABS line command.

- **tab-character**
  Specifies any character used to signify a tab.

- **operand**
  Specifies any of the following and their settings:
  - **ADS**- specifies every five positions from 1 through 65.
  - **ASM**- specifies the positions 1, 10, 16, and 36.
  - **COBOL**- specifies the positions 8, 12, 16, and 20.
- **STND-** specifies the positions 1, 10, 16, and 36.

**TIME Command**

```
TIME
```

The `TIME` command displays the time-of-day and the date in the message area of the screen. The `TIME` is given in military hh:mm:ss format. The date is given in standard mm/dd/yy format.

**TOP Command**

```
TOP
```

The `TOP` command displays the first full screen at the top of the source.

**UP Command**

```
UP [ / number-of-lines \ ]
     [     Max       ]
     [   < Half   > ]
     [     Page   ]
```

Use the `UP` (scroll up) command to display source lines above your current view. The amount you scroll is determined by the scroll option setting. You can override the setting at any time.

The default keys are PF7 and PF19.

- **number-of lines**
  Specifies the number of lines to scroll. If this is blank, then scrolling is determined by the scroll option.

- **Max**
  Specifies that you want to scroll to the first screen of text.

- **Page**
  Specifies that you want to scroll a full screen of text.

**CANCEL Command**

```
CANcel
```

Use the `CANCEL` command to cancel all changes made to the JCL since the last `SAVE` and to exit the Edit JCL screen. You are returned to the previous display.

**END/RETRY Command**

The `END/RETRY` command saves the module and returns to the preprocessor for another syntax conversion attempt. This command saves the module, as currently modified, by updating the module in the dictionary. The syntax is:

```
END/RETRY
```

Default keys: PF2, PF3, PF14, PF15.
The END/RETRY function is performed only if the module has actually been changed. If the module has not been changed, END/RETRY simply refreshes the screen.

**NULLS Command**

```
Nulls [ < ON > ]
\ / OFF /
```

Use the NULLS command to turn the NULLS Mode on and off. ON is the default. In the NULLS ON Mode, null characters replace all but the first blank in a line. If the line is completely blank, null characters are not substituted.

To use the keyboard insert mode key to insert characters, turn NULLS Mode ON so that edit automatically inserts trailing nulls in each data line in the display.

Normally, each data line is one field on the display. However, by using the TABS Mode, 3270 tab characters can be created in selected columns and is a way to break up a line into several fields. NULLS replaces trailing blanks in each field.

If edit places the cursor into a field, only blanks that follow the cursor on the line are changed to NULLS. If a character is deleted with the DELETE key, all of the characters in the field are shifted left one position and a null character is inserted into the last position in the field. If the ERASE EOF key is pressed, null characters fill the field on which the cursor is located from the cursor to the end of the field.

**PROFILE Command**

```
PROFile
```

Use the PROFILE command to display the environmental parameters under which your Edit session is operating.

The PROFILE identifiers that are displayed correspond to the primary commands. When you change parameters that are unique to your profile, the changes are saved to the CA IDMS/DC user sign-on.

Use the RESET command to clear the display of any line commands, column markers, or extraneous messages.

**RCHANGE Command**

```
RCHANGE
```

The RCHANGE command repeats the last CHANGE command that was executed.

The EDITOR begins searching at the first line of the display. When it reaches the bottom of data, the message “BOTTOM OF DATA REACHED” appears in the top line of the screen.

The default keys are PF6 and PF18.

**Using the RCHANGE and RFIND PF Keys to Selectively Change Strings**
You can use the RFIND PF key in conjunction with the RCHANGE PF key to selectively change strings. For example, consider the following sequences:

*First Sample RFIND-RCHANGE Sequence*

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>P CHANGE a Work-rt Name-1 1 Work-Name-2</td>
<td>In Part 1, you enter the CHANGE command to change the next occurrence of Work-Name-1 to Work-Name-2.</td>
</tr>
<tr>
<td>P RFIND key a rt</td>
<td>In Part 2, you want to find the next occurrence of Work-Name-1, but you are not sure if you want to change the string. By pressing the RFIND key, the next occurrence of Work-Name-1, which was specified in the CHANGE command during Part 1 is found.</td>
</tr>
<tr>
<td>P RCHANGE a key rt 3</td>
<td>In Part 3, you want to change the occurrence of Work-Name-1 that was found during Part 2 to Work-Name-2. By pressing the RCHANGE key, the occurrence is changed.</td>
</tr>
<tr>
<td>P RFIND key a rt RFIND key 4</td>
<td>In Part 4, you press the RFIND key to find the next occurrence of Work-Name-1. This time you do not want to change the string, so instead of pressing the RCHANGE key, you press the RFIND key again. The next occurrence of Work-Name-1 is found.</td>
</tr>
</tbody>
</table>

*Second Sample RFIND-RCHANGE Sequence*

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>P CHANGE a Work-rt Name-1 1 Work-Name-2 RFIND key</td>
<td>In Part 1, you want to find the next occurrence of Work-Name-1, but if you are not sure if you want to change it to Work-Name-2. If you key in the CHANGE command and press the RFIND key instead of the ENTER key, the RFIND is executed. The next occurrence of Work-Name-1 that was specified in the CHANGE command is found.</td>
</tr>
<tr>
<td>P RFIND key a rt 2</td>
<td>In Part 2, you decide that you do not want to change the string that was found during Part 1, press the RFIND key. The next occurrence of Work-Name-1 is found.</td>
</tr>
<tr>
<td>P RCHANGE a key rt 3</td>
<td>In Part 3, you want to change the string that was found during Part 2, press the RCHANGE key. This changes Work-Name-1 to Work-Name-2.</td>
</tr>
</tbody>
</table>
CAPS Command

CAPS [< ON►►>]

Use the CAPS command to turn the CAPS Mode on and off. With the CAPS ON, all new alphabetic data is translated into uppercase. With the CAPS OFF, the data remains unaffected. Data that is entered with CAPS Off remains in lowercase unless you edit the field.

CHANGE Command

CHANGE [< ALL >] [< string >] [< replacement-string >] [lb[rb]] [< X >]

Use the CHANGE command to search for and change the next occurrence of a string in the JCL.

The EDITOR begins searching at the position of the cursor when you enter the command, and it searches downward until the string is found. If the cursor is on the COMMAND line when you enter the command, the editor begins searching at the top line displayed.

If the string is not found, it is changed to the replacement string.

- **string**
  Specifies the string of characters to find and replace by `replacement-string`.

- **replacement-string**
  Specifies the string of characters used to replace `string`.

- **(asterisk)**
  Specifies the string value from the last FIND or CHANGE command entered.

- **ALL**
  Specifies that all occurrences of a string are to be replaced in scanned lines.

- **lb rb**
  Specifies left and right bounds (column positions) for the find. If specifying just the left bound, the string can be found anywhere within those bounds.

- **X**
  Specifies only excluded lines are to be scanned.

- **NX**
  Specifies only non-excluded lines are to be scanned.

Change Command Rules

- ALL, FIRST, and the 'lb rb' can appear in any order, but the `replacement-string` must follow `string`.

If a string has embedded blanks, enclose the string in single or double quotes. For example:

```
CHANGE 'program name'
```

If a string has a single asterisk (*), number, ALL, or FIRST, enclose `string` in quotation marks:
CHANGE '*' 'comments'

- If a string has leading quotation mark (single or double) enclose the string in quotation marks of the opposite kind. For example:

  CHANGE "t" t

- If CAPS Mode is OFF, enter the string as it appears in the text and the replacement-string as it should appear in the text. If CAPS Mode is ON, all lowercase characters are translated to uppercase characters.

CURSOR Command

CURsor

The CURSOR command moves the cursor directly to the COMMAND line. It functions in the same way as the home key.

The default keys are PF12 and PF24.

DOWN Command

DOWN [ \ number-of-lines ] \n   / Max\n   < Half  >
   \ Page

Use the DOWN (scroll down) command to display source lines below your current view. The amount you scroll is determined by the Scroll setting. The setting can be overridden at any time.

The default keys are PF8 and PF20.

- Max
  Specifies the last full screen at the bottom of the text.

- Half
  Specifies to scroll down half a screen.

- Page
  Specifies to scroll down a full screen.

EDITOR-ID Command

EDITOR-ID

The EDITOR-ID command displays the release number for the version of the EDITOR invoked. The release is displayed in message format.

END Command

END

Use the END command to save the current JCL (if changed) and return to the CA IDMS Extractor session.
ENTER Command

ENTER

The ENTER command redisplays the current screen with any changes made.

The default key is ENTER.

EXCLUDE Command

EXCLUDE

The EXCLUDE command limits your display to specific lines within the text being edited. Display the excluded lines with the RESET command.

- **first line**
  Specifies that the first line number is to be excluded from the display.

- **last-line**
  Specifies that the last line in the block of lines is to be excluded from the display. If this field is left blank, the default is the last line in the text.

- **ALL**
  Specifies that all lines in the text are excluded from the display.

FIND Command

FIND

Use the FIND command to search for a string in the JCL. The EDITOR begins searching at the position of the cursor when you enter the command. It searches downward until the string is found. If the cursor is on the COMMAND line when you enter the command, the EDITOR begins searching at the top line displayed.

The operands of this command can appear in any order.

- **ALL**
  Specifies that all occurrences of a string are to be found in scanned lines.

- **FIRST**
  Specifies that the first occurrence of a string is to be found.

- **string**
  Specifies the string is to be found.

- ***(asterisk)**
  Specifies the string value from the last FIND command entered.

- **lb rb**
  Specifies the left and right bounds (column positions) for the find. If specifying just the left bound, the string to be found must begin in that same column. If specifying both left and right bounds, the string to be found can appear anywhere within those bounds.
• X
  Specifies only excluded lines are to be scanned.

• NX
  Specifies only non-excluded lines are to be scanned.

FIRST Command

FIRST

The FIRST command displays the first screen of the JCL.

KEYS Command

KEYS

Use the KEYS command to view or change EDITOR PF key settings.

LAST Command

LAST

The LAST command presents the last screen of the JCL.

LEFT Command

LEFT [number-of-columns]

The LEFT command scrolls the current display to the left the specified number of columns. If the number parameter is blank, the Scroll Options are used.

BOTTOM Command

BOTTOM

The BOTTOM command displays the last full screen at the bottom of the JCL.

BOUNDS Command

BOUNDS [lb rb]

The BOUNDS command sets the left and right column bounds. These bounds are saved in the edit profile. In addition, the BOUNDS command alters the action of the FIND, CHANGE, Line Shifts, and other commands that are column-sensitive.

You must specify left and right bounds, or neither, and the left bound must be smaller that the right bound. If bounds are specified incorrectly or without an operand, the default is the data set minimum.

lb

Specifies the left bound.
rb

Specifies the right bound.

**Scroll Options 1**

```
Page
Half
Csr
number-of-lines
```

- **Page**
  Specifies that a whole screen is to be scrolled whenever an UP, DOWN, RIGHT, or LEFT command is used.

- **Half**
  Specifies that a half screen is to be scrolled whenever an UP, DOWN, RIGHT, or LEFT command is used.

- **Csr**
  Specifies that the line with the cursor on it is to become: the bottom line displayed whenever the UP command is used, the top line whenever the DOWN command is used, the left-most column whenever the RIGHT command is used, or the right-most column whenever the LEFT command is used.

- **number-of-lines**
  Specifies that this number of lines are to be scrolled whenever an UP, DOWN, RIGHT, or LEFT command is used.
  Use scroll options to specify how much of the screen is scrolled when you use an UP, DOWN, RIGHT, or LEFT primary command (or corresponding PF key) by itself.
  At the far-right side of the second line on the Edit screen, the word SCROLL appears followed by one of the scroll options. To change the current setting, enter one of the other options over the current setting. The scroll option you set remains in effect until you enter a different setting.

**Text Manipulation Line Commands**

The text manipulation line commands are used when entering or altering text data. These commands are especially useful when used in combinations. For example, use TS (text split), enter a word or phrase, and then use TF (text flow) to reformat the paragraph.

**Text Split Command**

```
TS number-of-lines
```

The TS line command splits the text at the cursor so that you can insert text. The text following the cursor is moved to the left margin of the paragraph and an additional line is inserted. The default is 1.

- **number-of-lines**
  Specifies the number of lines to be inserted between the split line.
Text Flow Command

**TF rb**

The TF command starts processing at the current line and flows text upward to the end of a paragraph. The end of a paragraph may be indicated by a:

- Blank line
- Change in indentation
- Special characters

Temporary lines such as COLS or BNDS are deleted before text is flowed.

A single blank separates existing text from the words that are flowed upward from a lower line. When the end of a sentence is detected, two blanks are inserted.

- **rb**
  - Specifies the right bound for the text.

See the online documentation for examples of Text Flow parameters.

Text Entry Command

**TE number-of-lines**

The TE line command formats the screen with an unnumbered open text entry area that may be used without regard for line overflow. The cursor is positioned at the beginning of the first line and the remainder of the screen is blank. After you type the data and press the ENTER key, the text is flowed into a paragraph format.

If you type a number after the TE command, open the text entry area provided for only that number of lines.

See the online documentation for examples of TE command examples.

- **number-of-lines**
  - Specifies the number of blank lines requested.

Destructive Line Shift Command

```plaintext
{[number-of-columns]}
)[number-of-columns] {([([number-of-columns]. .{([ ])[number-of-columns]. .})}
```

Destructive line shift moves the text a specified number of columns to the right-) or left-(. When the shift causes text to exceed the bound position, that text is discarded.
CA IDMS - 19.0

See the online documentation for line shift examples.

**number-of-columns**

Specifies the number of columns to shift. The default is 1.

(  )

Specifies that the line be shifted to the left or right one column.

(n  )

Specifies that the line be shifted to the left or right \( n \) columns.

((  ))

Specifies the first and last lines of a block to be shifted to the left or right one column.

((n  )))

Specifies the first line of a block to be shifted to the left or right \( n \) columns. Use a (( or )) to mark the last line of the block.

**Protective Line Shift Command**

**Contents**

- Bill-of-Material (BOM) (see page 111)
- CA IDMS Extractor Database (see page 111)
- Communication File (see page 111)
- Copy Utility (see page 111)
- Database Entry Point (see page 111)
- Database Extract Component (see page 111)
- Database Load Component (see page 111)
- Database Path (see page 111)
- Deselection (see page 111)
- Documentation Component (see page 111)
- Entry Index (see page 111)
- Entry Record (see page 112)
- Extract File (see page 112)
- Field Level Selection Criteria (see page 112)
- Hierarchical Index Selection (see page 112)
- Hierarchical Record Selection (see page 112)
- JCL Editor Component (see page 112)
- JCL Member Print Utility (see page 112)
- JCL Submission Component (see page 112)
Protective line shift moves the text a specified number of columns to the right- or left-<. When the shift causes the text to exceed the bound position, that text is retained and the shift operation is not completed.

See the online documentation for protective line shift examples.

`number-of-columns`

Specifies the number of columns to shift. The default is 1.

`<`

Specifies that the line be shifted to the left or right one column.

`>`

`<n`

`>n`

Specifies that the line be shifted to the left or right `n` columns.

`<<`

`>>`

Specifies the first and last lines of a block to be shifted to the left or right one column.

`<<n`

`>>n`

Specifies the first line of a block to be shifted to the left or right `n` columns. Use a `<<` or `>>` to mark the last line of the block.
Bill-of-Material (BOM)

CA IDMS Extractor defines a BOM structure as one record type owning another record type through two sets, one of which has a manual connection, the other an automatic.

CA IDMS Extractor Database

The database created when CA IDMS Extractor is installed. It contains the Selection Criteria Specifications and JCL members that you save.

Communication File

File used to pass information between the CA IDMS Extractor Batch Components.

Copy Utility

CA IDMS Extractor allows you to copy saved JCL members and Selection Criteria Specifications from other CA IDMS Extractor users.

Database Entry Point

Defines the beginning of a database path. An entry point can be either a record or an index that is used to retrieve a record.

Database Extract Component

The Batch Component that accesses the source database to extract record and set information.

Database Load Component

The Batch Component that loads the target database with information extracted from the source database.

Database Path

Tells the Extract Component how to walk through your source database and look for records to be extracted that are then loaded onto your target database.

Deselection

To remove all selection criteria from the record.

Documentation Component

The Online Component that provides information about using CA IDMS Extractor screens, command syntax, PF keys, the Transfer Facility, and the Online Message Facility.

Entry Index

A type of database entry point that defines the beginning of a database path. If you tell CA IDMS Extractor to use an entry index, records are retrieved without performing an area sweep. Extraction with an entry index usually takes less time than the same path definition with an Entry Record.
Entry Record

A type of database entry point that defines the beginning of a database path. When you tell CA IDMS Extractor to use an entry record, the Extract Component sweeps the area in which the record resides in order to retrieve the record. If you specify CALC keys in the Field Level Selection Criteria for the record, however, CA IDMS Extractor performs an OBTAIN CALC to retrieve the record.

Extract File

Contains information extracted from the source database by the Database Extract Component.

Field Level Selection Criteria

The field values and conditions that you want CA IDMS Extractor to test for when determining if a record should be extracted from the source database. For example, you can tell CA IDMS Extractor to extract a DEPT record if the DEPT-NAME field contains the value 'HISTORY'.

Hierarchical Index Selection

All records indexed by the named index are selected. The selected records are handled exactly like a hierarchical record selection.

Hierarchical Record Selection

Begins with the named record that CA IDMS Extractor marks as the entry record. All records that are members of sets owned by the entry record, do not already participate in a path, and are not indexed by an entry index, are selected in the path. All selected records that are owners of sets are subject to the same selection process as the entry record.

JCL Editor Component

The online component used to create and modify JCL members.

JCL Member Print Utility

The batch component that allows you to print a JCL member you created and saved using the JCL Editor Component.

JCL Submission Component

The online component used to submit JCL to your batch machine, and optionally, to insert parameters for the Batch Extract and Load Components of CA IDMS Extractor.

JCL Utilities Component

The online component used to copy, delete, print, and rename JCL members.

Network Index/Record Selection

Function exactly like a hierarchical selection. In addition, all owners of sets in which a selected record participates as a member are also selected.
A hierarchical selection selects all records going down in the hierarchy, whereas a network selection selects all records in the network.

**Record Level Selection Criteria**

The information you specify with the online Selection Criteria Specification Component that CA IDMS Extractor uses at EXTRACT time to determine if a record in the source database should be extracted.

**Recursive Data Structure**

A structure is considered to be recursive if a record type is retrieved by more than one set type in the same path. A Bill-of-Materials (BOM) structure is an example of a recursive data structure.

**Selection Criteria Specification**

The instructions that tell CA IDMS Extractor what records to extract from the source database.

**Selection Criteria Specification Component**

The online component used to describe the existing source database, describe the database path(s), describe Selection Criteria Specifications, and save the Specifications under a name of your choice.

**Source Database**

An existing database from which you extract records in order to create a new test database.

**Specification Name**

A name you give to the Selection Criteria Specification which is saved in the CA IDMS Extractor database.

**Specification Print Utility**

The batch component that allows you to print Selection Criteria Specifications that you created and saved.

**Specification Utilities Component**

The online component used to copy, delete, print, and rename your Selection Criteria Specifications.

**Syntax File**

The file used to pass appropriate CA IDMS DDDL syntax between the CA IDMS Extractor batch components, and the CA IDMS DDDL utility.

**Target Database**

A new or existing database that CA IDMS Extractor loads with records extracted from a source database.

**User Profile Component**

The online component used to tailor PF key assignments to suit your needs.
Workfile

A temporary file used by the Database Load Component. Information is saved in the Workfile between sort exits.

Online Session

This section presents an overview of a typical CA IDMS Extractor online session. It describes how to sign on and sign off CA IDMS Extractor and introduces the CA IDMS Extractor screens.

Signing on to CA IDMS Extractor

Follow the steps below to sign on to CA IDMS Extractor (unless your DBA or security administrator has implemented a different procedure):

1. Access your online CA IDMS/DC system.
2. At the CA IDMS/DC system prompt, type the task code assigned to the CA IDMS Extractor system and press the ENTER key. Obtain the task code, which is assigned at the time of installation, from your DBA. The default is **DBX**.

The CA IDMS Extractor Main Menu screen is displayed.

- Online Documentation (see page 114)
- Transfer Facility (see page 115)
- Typical Session Activities (see page 116)
- A Sample Selection Criteria Specification Session (see page 117)
- CA IDMS Extractor Logical System Overview (see page 133)
- Additional Screens (see page 134)

Online Documentation

At any point in a CA IDMS Extractor session, you can use the HELP command or the Help PF key to access the online documentation, which includes:

- Information about the CA IDMS Extractor screens
- Complete details about each of the commands, options, and keys used to edit a JCL member
- The CA IDMS Extractor online message facility.

To access the online documentation:

- Select option T at the Main Menu screen
- Enter the HELP command in the command field of any CA IDMS Extractor screen
CA IDMS Extractor allows you to view message text online.

To access the Message Index screen:

1. Access the online documentation

2. Type M in the OPTION field of the first screen of any CA IDMS Extractor online documentation module.

3. Press the ENTER key.

The Message Index screen is displayed. To view the message text:

1. Type the message number in the INDEX field of the Message Index screen.

2. Press the ENTER key.

The text for that message is displayed.

You can print the text for all CA IDMS Extractor messages using the CA Online Documentation Print Utility. See Error! Reference source not found. page=no. for detailed information on using this utility.

Transfer Facility

The Transfer Facility allows you to move between major components of CA IDMS Extractor without returning to the Main Menu screen. An equal sign followed by an alphanumeric string (=3.1) or the option command name (=SUTIL) transfers you to the function you want. The available transfer commands are shown below.
Transfer Facility Values and Commands

<table>
<thead>
<tr>
<th>System Level</th>
<th>Assigned Value</th>
<th>Option Command</th>
</tr>
</thead>
<tbody>
<tr>
<td>CA IDMS/DC</td>
<td>=X</td>
<td>=MENU</td>
</tr>
<tr>
<td>Main Menu</td>
<td>=0</td>
<td>=MENU</td>
</tr>
<tr>
<td>Specify Database Extract Specification</td>
<td>=1</td>
<td>=SPECIFY</td>
</tr>
<tr>
<td>Edit JCL</td>
<td>=2</td>
<td>=JCL</td>
</tr>
<tr>
<td>Edit JCL Member List</td>
<td>=2.1</td>
<td></td>
</tr>
<tr>
<td>Utilities Menu</td>
<td>=3</td>
<td>=UTIL</td>
</tr>
<tr>
<td>Specification Utilities</td>
<td>=3.1</td>
<td>=SUTIL</td>
</tr>
<tr>
<td>Specification Utilities List</td>
<td>=3.1.1</td>
<td></td>
</tr>
<tr>
<td>JCL Utilities</td>
<td>=3.2</td>
<td>=JUTIL</td>
</tr>
<tr>
<td>JCL Utilities List</td>
<td>=3.2.1</td>
<td></td>
</tr>
<tr>
<td>Copy Utility</td>
<td>=3.3</td>
<td>=COPY</td>
</tr>
<tr>
<td>Submit JCL for Execution</td>
<td>=4</td>
<td>=SUBMIT</td>
</tr>
</tbody>
</table>

Typical Session Activities

There are three categories of activity that you might perform during a typical CA IDMS Extractor session. Each of these activities are detailed below:

- Naming the source and target subschemas
- Creating a Selection Criteria Specification
- Running the Batch Components of CA IDMS Extractor.

This section introduces these activities and shows how to access the CA IDMS Extractor screens where these activities can be performed. Detailed field descriptions and information about how to input data are found in the online documentation.

Naming the Source and Target Subschemas

You must name the source and target subschemas to be used by the Extract and Load Components of CA IDMS Extractor. Do this naming with the Specify Database Build Specification screen, which is displayed when you select option 1 from the Main Menu screen.
Creating a Selection Criteria Specification

After you have named the source subschema, you must specify the selection criteria CA IDMS Extractor will use to extract the target database. This may include:

- Selecting particular areas/records/fields/sets
- Limiting the selection of a record type to a specific FROM/TO page range
- Selecting a record by field value(s)
- Selecting a record by direct DBKEY
- Selecting a group of records within a specified page range, beginning with the first record occurrence within the range
- Selecting a group of records within a specified page range beginning with the \textit{nth} occurrence within the range
- Skipping a specified number of records before selecting the next record occurrence
- Specifying the number of levels to be extracted in a bill-of-materials structure
- Specifying whether you want to extract all owners of extracted recursive records
- Limiting the total number of records selected by record type
- Limiting the number of records selected in each set occurrence

Running the Batch Components

After you have specified the selection criteria, you need to run the Batch Components of CA IDMS Extractor. To do this:

1. Create a JCL stream. Sample JCL is provided with CA IDMS Extractor. Edit this JCL to meet the requirements at your site.
2. Submit the JCL for execution.

See the section "Operations" for a detailed description of the JCL.

A Sample Selection Criteria Specification Session

The remainder of this section takes you through a sample session that includes the following steps:

1. Sign on to CA IDMS Extractor.
2. Specify the source and target subschemas.

3. Define the database path.

4. Limit the number of records displayed.

5. Select the entry record.

6. Define Record Level Selection Criteria for the DEPT record.

7. Select sets for the Extract Component.

8. Define Record Level Selection Criteria for the TEACHER record.

9. Select additional record/sets.

10. Return to the Database Entry Point Selection Menu screen.


12. Edit JCL.

13. Submit JCL for execution.

14. Sign off from CA IDMS Extractor.

The data structure diagram for the STUDENT-TEACHER database used in this sample session is shown below.

Information specific to the sample session is shown in each step.
Data Structure Diagram for the Sample Session

Example Data Structure Diagram

### Step 1. Sign On to CA IDMS Extractor

To sign on to CA IDMS Extractor:

1. Access your online CA IDMS/DC system.
2. At the CA IDMS/DC system prompt, type the task code assigned by your system administrator to CA IDMS Extractor and press the ENTER key. The default is DBX.

The CA IDMS Extractor Main Menu screen is displayed.

The user ID (or, if you did not sign on to CA IDMS/DC, the LTERM ID) is displayed in the upper right corner of the screen. This ID is the ID under which all specifications and JCL members are saved.

For the sample session

You have signed on to CA IDMS/DC prior to invoking CA IDMS Extractor. The user ID in this example is JSMITH. Your specifications and JCL members are saved under the user ID JSMITH.

Main Menu Screen

---

Step 2. Specify the Source and Target Subschemas

The Specify Database Extract Specification screen is shown below.

To access this screen:

1. Type 1 in the OPTION field of the CA IDMS Extractor Main Menu screen.

2. Press the ENTER key.

The Specify Database Extract Specification screen is displayed.

Specify Database Extract Specification Screen

---
Because the specification shown in the Name field is a new specification, you must also enter the names of the source and target subschemas. CA IDMS Extractor allows the same subschema to be named as the source subschema and the target subschema.

For the sample session

The data structure diagram for the STUDENT-TEACHER database shown in Data Structure Diagram for the Sample Session. For this sample session, the target database is to consist of DEPT records that own TEACHER records, selected from an existing source database. The other selection criteria follow:

- Define a path that has the DEPT record as an Entry Record
- Extract the first DEPT, and every fifth DEPT record after the first
- Limit the number of extracted DEPT records to 40
- Limit the extracted records to a subset of its from/to page range
- Extract every second TEACHER record in the DEPT-TEACHER set
- Place no limit on the number of TEACHER records.

The name of the specification is DEPT-TEACHER and both source and target databases reside in the primary dictionary for the CV under which you are running.

1. Type **DEPT-TEACHER** in the Specification Name field.
2. Use the TAB key to move to the Source Subschema field and type **TESTSUB1**.
3. Use the TAB key to move to the Source Subschema Database Name field and type **STUDTCHR**.
4. Use the TAB key to move to the Target Subschema field and type **LOADSUB1**.
5. Use the TAB key to move to the Target Subschema Database Name field and type **STUDTCHR**.
6. Press the ENTER key.

The Database Entry Point Selection Menu screen is displayed.
Rules for Source and Target Subschemas

All non-system-owned, integrated index records and sets in the source subschema must also be in the target database. The records defined in the source and target subschemas can reside in different areas and can have different page ranges. Additionally, the source and target subschemas must:

- Have record types that are the same length.
- Contain the same record/set relationships. That is, an owner or member of a set in the source database must be an owner or member of the same set type in the target database.
- Be accessible to the online component so that the above rules can be validated. Either subschema can reside in a dictionary load area or in a load/core-image library.

**Note:** The Compare Subschemas field allows CA IDMS Extractor to bypass comparison of the source and target subschemas. This allows the batch Database Extract Component of CA IDMS Extractor to access a database whose subschema is not accessible by the online Selection Criteria Specification component.

Step 3. Define the Database Path

Use the Database Entry Point Selection Menu screen, shown below, to begin the definition of a database path. A database path tells the Extract Component how to walk through your source database. A path must begin with either an Entry Record or an Entry Index.

- If you want to have only one Entry Record or Index, select either Record or Index and enter the name of the record or index on this screen.
- If you want to have more than one Entry Record or Entry Index or cannot recall the names of the records or indexes in the source subschema, select either Area, Record, or Index and leave the associated name field blank. The names of all areas, records, or system-owned integrated indexes in the source subschema are displayed, and you can select from them.

Database Entry Point Selection Menu Screen

Enter S to Show a list of records residing in the area, or S to Select a Record or Index as an Entry point, or H to Select a Hierarchy of records starting with the named Record or Index as an Entry point (this option is not valid with Area), or N to Select a Network of records starting with the named Record or Index as an Entry point (this option is not valid with Area), or D to Deselect a previously selected (*S) item, press ENTER.

Note: Leave the name field blank to obtain a list of names.
For the sample session

For a list of all areas in the database:

1. Type S in the selection field to the left of AREA.
2. Press the ENTER key.

The Area Selection List screen is displayed.

Note: Because we are defining only one path for the sample session Selection Criteria Specification, beginning with the DEPT record, we could have selected Record and filled in the Record Name field with DEPT. However, to illustrate the Area and Record Entry Point Selection List screens, we have selected Area.

Step 4. Limit the Number of Records Displayed

Use the Area Selection List screen to limit the number of records displayed on the Record Entry Point Selection List screen. Select the areas for which you want resident records displayed. An area name is displayed only if it contains at least one user-defined record. Areas containing only integrated index control records are not displayed.

For the sample session

Because the DEPT record is the only Entry Record and it resides in the DEPT-REGION:

1. Type S in the selection field to the left of DEPT-REGION.
2. Press the ENTER key.
Step 5. Select the Entry Record

The Record Entry Point Selection List screen is displayed.

The Record Entry Point Selection List screen is shown below. A list of all records in the area or areas selected on the Area Entry Point Selection screen is displayed. Because the DEPT-REGION was the only area selected, only records residing in the DEPT-REGION are displayed.

**Record Entry Point Selection List Screen**

<table>
<thead>
<tr>
<th>CAIDMS/DBX Rnn.nn Record Entry Point Selection List hh:mm mm/dd/yy</th>
</tr>
</thead>
<tbody>
<tr>
<td>COMMAND ===&gt; SCROLL ===&gt; PAGE USVMREC</td>
</tr>
</tbody>
</table>

Enter S to Select an Entry Record,
H to Select a Hierarchy of records starting with the selected Record as an Entry Record,
N to Select a Network of records starting with the selected Record as an Entry Record,
D to Deselect a previously selected (*S) Record,
blank to not show a previously selected path, press ENTER

s DEPT PREREQ SUBJECT TEACHER **END**

For the sample session

You want DEPT to be the only Entry Record.

1. Type $ in the selection field to the left of the DEPT record.
2. Press the ENTER key.

The Record Level Selection Criteria screen is displayed for the DEPT record.

Step 6. Define Record Level Selection Criteria--DEPT Record

You must define Record Level Selection Criteria for each Entry Record selected or record indexed by an Entry Index and all records selected in subsequent sets within the path definition. This is the criteria the Database Extract Component uses. The Record Level Selection Criteria screen for the DEPT record is shown below.

CA IDMS Extractor fills in the current Entry Record or Index Name; the current Path Record Name; its from/to page numbers; how the record is accessed; and the area, record, or set name. The record can be accessed by either an area sweep, an Entry Index, or a set walk. In this case, the DEPT record is accessed by an area sweep because it is an Entry Record.

There is also an indication on the screen as to whether the Selection Criteria Specification is complete or not complete. For the example, the Specification is **not** complete.
Record Level Selection Criteria Screen--DEPT Record

For the sample session

You want to limit the selection of DEPT records to a subset of its from/to page range, have the first DEPT and every fifth DEPT record after the first selected, and select no more than 40 DEPT records.

1. Type S to the left of the Limit Page Range field.
2. Type 88003 in the From Page field and press the EOF key.
3. Type 88008 in the To Page field and press the EOF key.
4. Type S to the left of the Limit Records of This Type field.
5. Type 40 in the Record Count field and press the EOF key.
6. Type S to the left of the Skip Between Records field.
7. Type 4 in the Skip Count field and press the EOF key.
8. Type S to the left of the First Within Range field.
9. Press the ENTER key.

The Path Record Set Selection List screen is displayed.

Step 7. Select Sets for the Extract Component

The Path-Record Set Selection List screen is shown below. Use this screen to select the sets that you want the Extract Component to walk from the current path record. All sets owned by the current record and all sets in which the current record participates as a member are displayed, except a set already selected in a path of this specification. The field in front of the set name indicates whether the current record is an owner or member of that set.
CA IDMS Extractor fills in the current record name and shows how the record is accessed. The DEPT record participates in three sets: DEPT-TEACHER, DEPT-ACADEMIC, and DEPT-GENERAL. The DEPT record is an owner of all of these sets and does not participate as a member in any sets. It is being accessed by an area sweep.

**Note:** A set can only be selected once in a specification. If you need to access the same set in more than one way, you must define multiple Selection Criteria Specifications. Remember that you can run the CA IDMS Extractor Database Load Component on a non-empty target database.

### Path-Record Set Selection List Screen--DEPT Record

<table>
<thead>
<tr>
<th>Owner/Mbr</th>
<th>Set Name</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Owner</td>
<td>DEPT-TEACHER</td>
<td></td>
</tr>
<tr>
<td>Owner</td>
<td>DEPT-ACADEMIC</td>
<td></td>
</tr>
<tr>
<td>Owner</td>
<td>DEPT-GENERAL</td>
<td><strong>END</strong></td>
</tr>
</tbody>
</table>

For the sample session

You want to select the DEPT-TEACHER set.

1. Type **S** in the selection field to the left of the DEPT-TEACHER set.
2. Press the ENTER key.

The Record Level Selection Criteria screen is displayed again for you to specify the Record Level Selection Criteria for the next record in the path definition.

**Step 8. Define Record Level Selection Criteria--TEACHER Record**

Because the DEPT-TEACHER set was included in the path definition, you must define Record Level Selection Criteria for the TEACHER record. The Record Level Selection Criteria screen for the TEACHER record is shown below.
In this case, DEPT is the Entry Record. The current path record for which Record Level Selection Criteria is being defined is the TEACHER record. The TEACHER record is accessed by walking from the owner (SET-WALK) in the DEPT-TEACHER set.

There is an indication on the screen as to whether this Selection Criteria Specification is complete or not complete. For the example, the specification is **not** complete.

### A302101.Record Level Selection Criteria Screen--TEACHER Record

<table>
<thead>
<tr>
<th>CAIDMS/DBX Rnn.nn</th>
<th>Record Level Selection Criteria</th>
<th>hh:mm mm/dd/yy</th>
</tr>
</thead>
<tbody>
<tr>
<td>COMMAND</td>
<td>USVMRLS</td>
<td></td>
</tr>
<tr>
<td>Entry Record:</td>
<td>DEPT</td>
<td></td>
</tr>
<tr>
<td>Path Record:</td>
<td>TEACHER</td>
<td></td>
</tr>
<tr>
<td>Accessed By:</td>
<td>SET WALK of area DEPT-TEACHER set, walked own to mb</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Enter S to Select, D or blank to Deselect selection criteria, press ENTER. Note: The specification is currently not complete.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- Direct DBKEY: Page ===> 0000000000 Line ===> 0000
- Limit Page Range: From Page ===> 0000088001 To Page ===> 0000088015
- Limit Records of This Type: Total Record Count ===> 0000000040 Per Set Occurrence ===> 0000000000
- s Skip Between Records: Skip Count ===> 0000000002
- First Within Range
- Nth Within Range: Which Occurrence ===> 0000000000
- Field Level Selection Criteria: Display Criteria? ===> Y (Y/N)

### For the sample session

You want to specify that a TEACHER record should be skipped in between selections:

1. Type **S** to the left of the SKIP BETWEEN RECORDS field.
2. TAB to the SKIP COUNT field, type **2**, and press the EOF key.
3. Press the ENTER key.

The Path Record Set Selection screen is displayed.

If you want to **all** records of this type displayed:

1. Type **A** to the left of the LIMIT RECORDS OF THIS TYPE field.
2. Press the ENTER key.

### Step 9. Select Additional Record/Sets

The names of all sets owned by the TEACHER record and all sets in which the TEACHER record participates as a member are displayed on the Path Record Set Selection List screen. If you want to have more records/sets in the path, select the appropriate sets at this time.
Note: After a path set is selected, CA IDMS Extractor displays the Record Level Selection Criteria screen for all associated owner and member records of selected paths. You must specify Record Level Selection Criteria for all records in the path.

Path-Record Set Selection List Screen--TEACHER Record

For the sample session

You only want to select the DEPT-TEACHER set. Therefore, the path definition is complete.

1. Press PF3. The Record Entry Point Selection screen is displayed.
2. Press PF3. The Database Entry Point Selection Menu screen is displayed.
Step 10. Return to the Database Entry Point Selection Menu Screen

The Database Entry Point Selection Menu is shown below. If you want to define another path, or modify the path just defined, select the appropriate area, record, or index and continue with the definition.

Database Entry Point Selection Menu Screen 1

```
CAIDMS/DBX Rnn.nn Database Entry Point Selection Menu hh:mm mm/dd/yy
COMMAND ====> USVMENT

Enter S to Show a list of records residing in the area, or
S to Select a Record or Index as an Entry point, or
H to Select a Hierarchy of records starting with the named Record or
Index as an Entry point (this option is not valid with Area), or
N to Select a Network of records starting with the named Record or
Index as an Entry point (this option is not valid with Area), or
D to Deselect a previously selected (*S) item, press ENTER
Note: Leave the name field blank to obtain a list of names.

s Area
  Area Name ===> (blank for Area Name list)

Record
  Record Name ===> (blank for Record Name list)

Index
  Index Name ===> (blank for Index Name list)

An Entry Index must be a System Owned Integrated Index
```

For the sample session

The Selection Specification Criteria are complete.

Press PF3 to end the specification process. The specification is saved on the CA IDMS Extractor database.

The Specify Database Extract Specification screen is displayed.

Step 11. Return to the Specify Database Extract Specification Screen

CA IDMS Extractor returns you to the Specify Database Extract Specification screen shown below. You can edit the specification again, create a new specification, or modify another existing specification. For specifications that you have saved, CA IDMS Extractor allows you to modify the values that you specified, add new path definitions, or delete old specifications.
Specify Database Extract Specification Screen 1

Specify Database Extract Specification Screen 1

- CAIDMS/DBX Rnn.nn Specify Database Extract Specification hh:mm mm/dd/yy
- Command ===> USVMSPC
- Enter Spec Name and, if a new spec, Subschema and DB Names, press ENTER
- Specification Name ===> dept-teacher (blank for Specification list)
- Source Subschema ===> testsub1 DMCL ===> CVDMCL
- Dictionary Name ===> Database Name ===> (or Segment)
- Database Node ===> Target Subschema ===> loadsub1 DMCL ===> CVDMCL
- Dictionary Name ===> Database Name ===> (or Segment)
- Database Node ===> Compare Subschemas ===> Y (Y/N)

For the sample session

1. Press PF3.

The CA IDMS Extractor Main Menu screen is displayed.

Step 12. Edit JCL

You need to create JCL that executes the Extract and Load Components for the selection criteria that you just specified. The Edit JCL Entry screen is shown below.

To access this screen:

1. Type 2 in the OPTION field on the CA IDMS Extractor Main Menu screen.
2. Press the ENTER key. The Edit JCL screen is displayed.

The CA EDITOR is invoked from this screen, allowing you to edit the JCL member.

You should not specify the PROCESS parameter statement in the JCL member because CA IDMS Extractor creates it for you when the JCL is submitted for execution. You can use one JCL member for several specifications.

Edit JCL Screen

- CA IDMS/DBX Rnn.nn Edit JCL
- Command ===> USVMEDJ
- Enter JCL Member Name to Edit, press ENTER
- JCL Member Name ===> (blank for Member Name list)
For the sample session

Create a new JCL member called STUDENT-DATABASE:

Type **STUDENT-DATABASE** in the JCL Member Name field.

1. Press the ENTER key. The CA EDITOR is invoked.

2. Create the needed JCL statements (shown in **Error! Reference source not found.** page=no.).

**Note:** Model execution JCL may have been uploaded to the CA IDMS Extractor database during installation of CA IDMS Extractor. If it was, the member -MODEL-JCL can be edited; renamed to STUDENT-DATABASE and edited; or copied to STUDENT-DATABASE and edited. Use the JCL Utilities screen (option 3.2) to rename or copy a JCL member.

**EDITOR Screen**

```
EDIT STUDENTDATABASE          COLUMNS 001 072
COMMAND ===>                  SCROLL ===> CSR
****** *** TOP OF DATA *************** CA IDMS/DBX ***
```

**Step 13. Submit JCL for Execution**

Use the Submit JCL for Execution screen shown below to submit the JCL you created in the prior step.

To access this screen:

1. Type **4** in the OPTION field on the CA IDMS Extractor Main Menu screen.

2. Press the ENTER key. The Submit JCL for Execution screen is displayed.
CA IDMS Extractor creates the correct PROCESS parameter statement for you and bases the parameter statement on the Specification that you enter in the Specification Name field. The parameter statement is placed after the first SYSIPT statement in your JCL.

Submit JCL for Execution Screen

CAIDMS/DBX Rnn.nn  Submit JCL for Execution & cont.
hh:mm mm/dd/yy
COMMAND ===> USVMSUB
Enter JCL Member Name and Specification Name to be submitted, press ENTER.
Note: You need to fill in the rest of the fields if you are having DBX insert a parameter statement into the JCL stream.

JCL Member
Name ===> STUDENT-DATABASE (blank for Member Name list)

Specification
Name ===> DEPT-TEACHER (blank for Specification list)

InsertParms into JCL? ===> Y (Y/N) Compare Subschemas? ===> Y (Y/N)

Transient SSC Name ===> ExtractUser Exit ===> 
Create SIGNON stmt? ===> Y (Y/N) Dictionary Password ===> password
SIGNON User ID ===> userid

New Source SSC Name ===> DMCL Name ===> 
Dict Name/Node => / DB Name/Node => /

For the sample session

Press the ENTER key, accepting all of the default values and actions. The Submit JCL for Execution screen is redisplayed with the message DBX0039I JCL SUBMITTED displayed under the COMMAND field.

Step 14. Sign Off From CA IDMS Extractor

To sign off CA IDMS Extractor:

- Return to the CA IDMS Extractor Main Menu screen, type X in the OPTION field, and press the ENTER key
- Type =X in the COMMAND/OPTION field of any screen, or press the PF key associated with the =X command.

You return to CA IDMS/DC system prompt.

If you were editing a JCL member or specifying a Selection Criteria Specification when either the X or =X command is entered, your JCL member or Selection Criteria Specification is saved prior to exiting to CA IDMS/DC.

Signing off ends your CA IDMS Extractor session. Any changes you have made to your PF key assignments are saved.
How Specifications and JCL Members are Saved

If you are signed onto CA IDMS/DC when you execute CA IDMS Extractor, your specifications and JCL members are saved on the CA IDMS Extractor database under your user ID. If you are not signed onto CA IDMS/DC when you execute CA IDMS Extractor, your specifications and JCL members are saved on the CA IDMS Extractor database under your logical terminal identifier (LTERM ID) of the terminal you are using.

If you enter CA IDMS Extractor again and have not signed on to CA IDMS/DC, you are not able to access your specifications saved in the previous session.

If you enter CA IDMS Extractor again, either after signing on to CA IDMS/DC, or after your terminal is given a different LTERM ID by CA IDMS/DC, you are also not able to access your specifications saved in the previous session.

You should sign on to CA IDMS/DC before using CA IDMS Extractor.

CA IDMS Extractor Logical System Overview

CA IDMS Extractor is menu-driven. The diagram below shows a logical overview of the system. To perform some of the functions, several screens may be used. Each screen and its associated fields are described in detail in the online documentation that you can access from any CA IDMS Extractor screen by using the HELP command or the appropriate PF key.
Logical System Overview

Additional Screens

The CA IDMS Extractor screens listed below were not covered in the sample CA IDMS Extractor session:

- Specify Specification List
- Begin Edit Record Selection List
- Index Entry Point Selection List
- Area Deselection
- Entry Record/Index Deselection

*You can invoke these functions from any screen
Specify Specification List Screen

The Specify Specification List screen, shown below, is displayed when the Specification Name field on the Specify Database Extract Specification screen is left blank. The names of all specifications that you have previously saved are displayed on this screen.

The COMPLETE field indicates whether you have specified Record Level Selection Criteria for ALL selected records/sets in all paths in the specification. If N(o) is shown, you must finish the specification before using it.

To select a specification for viewing or modification:

1. Type S in the selection field to the left of the specification name you want to select.

2. Press the ENTER key.

Specify Specification List Screen 1

<table>
<thead>
<tr>
<th>Specification Name</th>
<th>Created</th>
<th>Last Modified</th>
<th>Size</th>
<th>Complete</th>
</tr>
</thead>
<tbody>
<tr>
<td>DEPARTMENT-TEACHER</td>
<td>mm/dd/yy</td>
<td>mm/dd/yy</td>
<td>11</td>
<td>Y</td>
</tr>
</tbody>
</table>

**END**
Begin Edit Record Selection List Screen

The Begin Edit Record Selection List screen, shown below, is displayed when you specify Y in the Begin viewing/editing in the middle of a path definition field on the Specify Database Extract Specification screen. This field is only displayed when viewing a previously saved specification.

Use this screen to bypass the viewing of all the Record Level Selection Criteria and Path-Record Set Selection List screens that would appear if you were to view the path definition beginning with the Entry Record or Index. A list of all records and the sets in which the record participates in, and which are included in your path definition, are displayed on this screen.

The RLSC field indicates whether Record Level Selection Criteria have been specified for the indicated record/set.

The Recursive field indicates whether or not the record in the set is recursive.

To select a record to begin viewing or editing:

1. Type S in the selection field to the left of the record name you want to select.

2. Press the ENTER key.

Index Entry Point Selection List Screen

The Index Entry Point Selection List screen, shown below, is displayed when you select Index on the Database Entry Point Selection Menu screen and leave the Index Name field blank.

To select indexes that are to be entry points into the source database (that is, indexes that define the beginning of a path definition):

1. Type S in the selection field to the left of the index you want to select as an entry point.

2. Press the ENTER key.
**Index Entry Point Selection List Screen 1**

```
CAIDMS/DBX Rnn.nn  Index Entry Point Selection List  hh:mm mm/dd/yy
COMMAND ===> SCROLL ===> PAGE USVMIDX

Enter S to Select an Entry Index,
  H to Select a Hierarchy of records starting with the selected Index as
  an Entry Index,
  N to Select a Network of records starting with the selected Index as
  an Entry Index,
  D to Deselect a previously selected (*S) Index,
  blank to not show a previously selected path, press ENTER

S  IX-SUBJ-LNAME
**END**
```

**Area Deselection Screen**

The Area Deselection screen, shown below, is displayed when you deselect an entry area on the Database Entry Point Selection Menu screen.

Deselecting an entry area implies that ALL entry records residing in the area are deselected. Deselecting an entry record implies that the ENTIRE path definition will be deleted.

Each previously selected area is shown with an *S to the left of its name. To deselect an area, type a D to the left of the name you want deselected and press the ENTER key.

When the Area Deselection screen appears, you have two processing options:

- Press the ENTER key to continue with the deselection or
- Type the END command or press the End PF key to cancel the deselection.

**Area Deselection Screen 1**

```
CAIDMS/DBX Rnn.nn  Area/Record/Index/Set Deselection  hh:mm mm/dd/yy
COMMAND ===> USVMDSL

*****************************************************************************
**                     WARNING  WARNING  WARNING                              **
**                     **                                                  **
**                     **                                                  **
**                     You are requesting to DESELECT a previously selected AREA **
**                     **                                                  **
**                     **                                                  **
**                     **                                                  **
**                     **                                                  **
**                     **                                                  **
**                     **                                                  **
**                     **                                                  **
**                     **                                                  **
**                     **                                                  **
**                     **                                                  **
**                     **                                                  **
**                     **                                                  **
**                     **                                                  **
**                     **                                                  **
**                     **                                                  **
**                     **                                                  **
**                     **                                                  **
**                     **                                                  **
**                     **                                                  **
**                     **                                                  **
**                     **                                                  **
**                     **                                                  **
**                     **                                                  **
**                     **                                                  **
**                     **                                                  **
**                     **                                                  **
**                     **                                                  **
**                     **                                                  **
**                     **                                                  **
**                     **                                                  **
**                     **                                                  **
*****************************************************************************

```
Entry Record/Index Deselection Screen

The Entry Record/Index Deselection screen, shown below, is displayed when you deselect an Entry Record or Entry Index on the Database Entry Point Selection Menu screen.

Deselecting an Entry Record or Entry Index implies that the ENTIRE path definition is deleted.

Each previously selected Entry Record or Entry Index is shown with an *S to the left of its name. To deselect the Entry Record or Entry Index, type a D to the left of the name you want deselected and press the ENTER key.

When the Entry Record/Index Deselection screen appears, you have two processing options:

- Press the ENTER key to continue with the deselection or
- Type the END command or press the End PF key to cancel the deselection.

Path Set Deselection Screen

The Path Set Deselection screen is displayed when you deselect a path set on the Path-Record Set Selection List screen. Deselecting a path implies that all sets selected in the path after the deselected set are also deselected. However, the deselected process stops at recursive records. Sets selected in the path after recursive records are not deselected. This allows you to flip-flop the way sets are walked in recursive structures.

Deselecting a path set can have one of two effects:

- **If the path set does not participate in a recursive structure**, deselecting a path set implies that ALL sets selected in the path AFTER the deselected set are also deselected.
If the owner-member record of this set participates in a recursive structure and is therefore accessed by other sets in the path, you have one of the following options:

- Deselect THIS set only (Option T)
- Deselect this set and ALL selected sets in the path AFTER this set (Option A).

Thus, you can deselect only one set or flip-flop the way a recursive bill-of-materials structure is walked.

### Field Level Selection Criteria Screen

The Field Level Selection Criteria screen, shown below, is displayed when the Field Level Selection Criteria field is selected on the Record Level Selection Criteria screen. This screen is used to select field level criteria for the specified path record.

To define field level selection criteria:

1. Type $S$ in the selection field to the left of one of the operands you want to select or type $S$ in the selection field to the left of one of the range options you want to select.

2. Press the ENTER key.
Enter the criterion number you wish to view/edit, or
S to Select one (1) of the operators to be applied against the data you
specify on subsequent Record Element screen(s), or
END command to end viewing/editing field level criteria, press ENTER

Extract a record from the source database if its data is
EQ (=) S NE ( =) LT (<) LE (<=) GT (> ) GE (>=)

Within Not Within

the data specified on the following Record Element screen, OR

the range of data specified on the following two Record Element screens.

Despecify Character for Record Element screen ===> 

Record/Element Review Screen

The Record/Element Review screen, shown below, is displayed after selecting an operand on the
Field Level Selection Criteria screen. This screen is used to specify values that CA IDMS Extractor uses
to determine which records to extract from the source database.

Example

To have CA IDMS Extractor select records where the DEPT-ROOM-NO is not equal to zero:

1. Type 0 for all the field values for the DEPT-ROOM-NO field and press the EOF key.
   When values are entered in multiple fields, each field must satisfy the condition or database
   record is not extracted. In this example, DEPT-ROOM-NO(1) must be not=0 and DEPT-ROOM-NO(2)
   must be not=0, etc. for the database record to be extracted.

2. Type the END command or press the End PF key to save the criteria and return to the Field
   Level Selection Criteria screen.
   At this time, another criteria can be specified by selecting another operator and pressing the
   ENTER key again. In this situation, the database record would be extracted if its data satisfied
   either Criteria 1 or Criteria 2.

Record/Element Review Screen 1

```plaintext
GSI Rnn.nn   Record/Element Review   h
COMMAND ==>   GSIRECCO
TDB0077I MODIFY FIELD SELECTION DATA FOR "NOT_EQUAL" (NE) CONDITION
RECORD: DEPT V 6
        DICT:  
        DSPEC: \ LINE 0001 OF 009
        02 DEPT-NAME......................... A
        02 DEPT-HEAD.......................... A
        02 DEPT-ROOM.......................... G
        05 DEPT-ROOM-NO(0001)............... P 000
        05 DEPT-ROOM-NO(0002)............... P 000
        05 DEPT-ROOM-NO(0003)............... P 000
        05 DEPT-ROOM-NO(0004)............... P 000
        05 DEPT-ROOM-NO(0005)............... P 000
        05 DEPT-ROOM-NO(0006)............... P 000
```
Record/Element Display & Modification Commands

The following commands are active at the Record/Element Review screen:

<table>
<thead>
<tr>
<th>Command</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>DISPLAY</td>
<td>Redisplay the record after using the SET command.</td>
</tr>
<tr>
<td>EXIT</td>
<td>Leave the record display and return to the CA IDMS Extractor session.</td>
</tr>
<tr>
<td>INITIALIZE</td>
<td>Ensure that all record element descriptions are available and initialize elements to null values.</td>
</tr>
<tr>
<td>QUIT</td>
<td>Leave record display, terminate the CA IDMS Extractor session, and return to the CA IDMS/DC system prompt.</td>
</tr>
<tr>
<td>SET HEX / NATIVE</td>
<td>Change the display format to/from hexadecimal format.</td>
</tr>
<tr>
<td>SET AUTOHEX</td>
<td>Change the display format of any element whose data content does not match its picture and/or usage to hexadecimal.</td>
</tr>
<tr>
<td>SET LOWERCASE</td>
<td>Change the display format of any element to lowercase.</td>
</tr>
</tbody>
</table>

Record/Element Display & Modification Commands Syntax Summary

DISPLAY [CONTINUE] EXIT INITIALIZE record-name QUIT / \ [SET] AutoHEX < ON > \ OFF / / \ [SET] element < HEX > \ NATIVE / / \ [SET] LowerCASE < ON > \ OFF /

DISPLAY Command

DISPLAY [CONTINUE]

Use the DISPLAY command to redisplay the record under review or modification. You must enter this command in order to enact a SET AUTOHEX, SET HEX/NATIVE, or SET LOWERCASE command.

Rule for the DISPLAY Command

Record display may exhaust available CA IDMS Extractor storage before all fields (or occurrences) are formatted. CA IDMS Extractor responds by displaying a warning message to inform you of this condition. Use the DISPLAY CONTINUE command to begin formatting at the point in the record where storage is exhausted.

EXIT Command

exit

Use the EXIT command to leave the record display and return to the CA IDMS Extractor session.

INITIALIZE Command

INITialize record-name

Use the INITIALIZE command at the Record/Element Review screen to ensure that all record element descriptions are available and to initialize elements to null values appropriate to usage: numeric fields are initialized to zero and others are initialized to spaces.
QUIT Command

Quit

Use the QUIT command to leave the record display, terminate the CA IDMS Extractor session, and return to the CA IDMS/DC system prompt.

SET AUTOHEX Command

/ \ [SET] AutoHEX < ON > \ OFF /

Use the SET AUTOHEX command to automatically change the record/element display format of any element whose data contents does not match its PICTURE and/or USAGE to hexadecimal. Items with valid data are not affected.

If ON or OFF is not specified, the option is toggled.

If AUTOHEX is specified while a record is being displayed, enter the DISPLAY command to change the fields on the current display.

SET HEX/NATIVE Command

/ \ [SET] element < HEX > \ NATIVE /

Use the SET HEX/NATIVE command to change the display format of a specific element to/from hexadecimal format.

The command does not cause the record, group, or element to be displayed. The new mode takes effect only when you enter the DISPLAY command or when a new occurrence of the record is obtained from the database.

When you specify NATIVE at the element level, CA IDMS Extractor displays the level number and usages of the specified element in the message area.

SET LOWERCASE Command

/ \ [SET] LowerCASE < ON > \ OFF /

Use the SET LOWERCASE command to change the display format of any element to lowercase.

If ON or OFF is not specified, the option is toggled.

If LOWERCASE is specified while a record is being displayed, enter the display command to change the fields on the current display.

Field Level Criteria Deselection Screen

The Field Level Criteria Deselection screen, shown below, is displayed when you want to deselect ALL field level selection criteria for a selected record. To deselect the field level criteria, place a D to the left of the Field Level Selection Criteria field on the Record Level Selection Criteria screen and press the ENTER key.

When the Field Level Criteria Deselection screen appears, you have two processing options:
Press the ENTER key to continue with the deselection or

Type the END command or press the End PF key to cancel the deselection.

Field Level Criteria Deselection Screen 1

Confirm Subschema Record Date Screen

The Confirm Subschema Record Date screen, shown below, is displayed when you are viewing or editing an existing path definition and want to look at Field Level Selection Criteria for a record. If the dictionary has been updated and any element within that record has changed since the last time the specification was saved, this screen is displayed.

The Confirm Subschema Record Date screen warns you that the dictionary has been updated since the last time you specified Field Level Selection Criteria for that record. If elements were added to or deleted from the dictionary, or the lengths of any element changed, you will not be able to use the Field Level Selection Criteria you specified. The name of the record and the date at which the record was updated in the dictionary are displayed.

When the Confirm Subschema Record Date screen appears, you have two processing options:

- Press the ENTER key to continue with the deletion of the Field Level Selection Criteria or
- Type the END command or press the End PF key to continue using the Field Level Selection Criteria.

Confirm Subschema Record Date Screen 1
Field Mismatch Screen

The Field Mismatch screen, shown below, is displayed when you are viewing or editing an existing path definition and you want to look at the Field Level Selection Criteria for a record. This screen is displayed after the Confirm Subschema Record Date screen.

The Field Mismatch screen informs you that the dictionary has been updated since the last time you specified Field Level Selection Criteria for the record. The data lengths or displacements of the fields defined in the dictionary no longer match the data lengths or displacements of the fields in your Field Level Selection Criteria.

When the Field Mismatch screen appears, you have two processing options:

- Press the ENTER key to continue with the deletion of the Field Level Selection Criteria.
- Type the END command or press the End PF key to return to the Field Level Selection Criteria screen.

Field Mismatch Screen 1
Utilities Menu Screen

The Utilities Menu screen, shown below, is displayed when option 3 (utilities) is specified on the CA IDMS Extractor Main Menu. The Utilities Menu screen gives you the following options:

**Deleting, Printing, or Renaming Selection Criteria Specifications**

Type 1 in the OPTION field.

Press the ENTER key. The Specification Utilities screen is displayed.

**Deleting, Printing, or Renaming JCL Members**

Type 2 in the OPTION field.

Press the ENTER key. The JCL Utilities screen is displayed.

**Copying Selection Criteria Specifications or JCL Members**

Type 3 in the OPTION field.

Press the ENTER key. The Copy Utility screen is displayed.

Utilities Menu Info

<table>
<thead>
<tr>
<th>CAIDMS/DBX Rnn.nn</th>
<th>Utilities Menu</th>
</tr>
</thead>
<tbody>
<tr>
<td>OPTION</td>
<td>====&gt; 1</td>
</tr>
</tbody>
</table>

Enter Option, press ENTER

1. Specification - Delete, Print, or Rename Selection Criteria Specifications
2. JCL Member - Delete, Print, or Rename JCL Members
3. Copy - Copy Selection Criteria Specifications or JCL Members

Specification Utilities Screen

The Specification Utilities screen, shown below, is displayed when option 1 is specified on the Utilities Menu screen. The Specification Utilities screen allows you to delete, rename, or print a specification.

**Deleting a Selection Specification**

Type D in the OPTION field.

Type the name of the specification you want to delete in the Name field.

Press the ENTER key.
Renaming a Selection Criteria Specification

Type R in the OPTION field.

Type the name of the specification you want to rename in the Name field.

Type a new name for the specification in the Newname field.

Press the ENTER key.

Printing a Selection Criteria Specification

The -MODEL-PRINT-SPEC member must first be copied and renamed to PRINT-SPEC then edited to tailor it for your environment. This JCL is submitted to perform the actual printing of the specification.

1. Type P in the OPTION field.

2. Type the name of the specification you want to print in the Name field.

3. Press the ENTER key.

Specification Utilities Screen 1

<table>
<thead>
<tr>
<th>Specification Utilities Menu</th>
<th>10-Jan-2018</th>
<th>146/240</th>
</tr>
</thead>
<tbody>
<tr>
<td>CAIDMS/DBX Rnn.nn</td>
<td>Specification Utilities</td>
<td>hh:mm mm/dd/yy</td>
</tr>
<tr>
<td>OPTION ====&gt; D</td>
<td>USVMSUT</td>
<td></td>
</tr>
<tr>
<td>blank - Display Specification List</td>
<td></td>
<td></td>
</tr>
<tr>
<td>D - Delete Spec</td>
<td>R - Rename Spec (From &quot;Name&quot; to &quot;Newname&quot;)</td>
<td></td>
</tr>
<tr>
<td>P - Print Spec</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Enter Option and optional Specification Name(s), press ENTER</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

JCL Utilities Screen

The JCL Utilities screen, shown below, is displayed when option 2 is specified on the Utilities Menu screen. Use the JCL Utilities screen to delete, rename, or print a JCL member.

Deleting a JCL Member

Type D in the OPTION field.

Type the name of the JCL member you want to delete in the Name field.

Press the ENTER key.
Renaming a JCL Member

Type R in the OPTION field.

Type the name of the JCL member you want to rename in the Name field.

Type a new name for the JCL member in the Newname field.

Press the ENTER key.

Printing a JCL Member

The -MODEL-PRINT-JCL member must first be copied and renamed to PRINT-JCL and then edited to tailor it for your environment. This JCL is submitted to perform the actual printing of the JCL member.

1. Type P in the OPTION field.
2. Type the name of the JCL member you want to print in the Name field.
3. Press the ENTER key.

JCL Utilities Screen 1

Copy Utility Screen

The Copy Utility screen, shown below, is displayed when option 3 is specified on the Utilities Menu screen. Use the Copy Utility screen to copy a Selection Criteria Specification or JCL member from any CA IDMS Extractor user ID, including yourself.

Copying a Selection Criteria Specification

Type 1 in the OPTION field.

Type the user ID or LTERM ID you want to copy from in the Copy from User-id field.

Type the name of the Selection Criteria Specification in the Name field.

Type a name for the copied Selection Criteria Specification in the Newname field.
Press the ENTER key.

Copying a JCL Member

Type 2 in the OPTION field.

Type the user ID or LTERM ID you want to copy from in the Copy from User-id field.

Type the name of the JCL member in the Name field.

Type a name for the copied JCL member in the Newname field.

Press the ENTER key.

Copy Utility Screen Info

<table>
<thead>
<tr>
<th>CAIDMS/DBX Rnn.nn Copy Utility &amp; OPTION</th>
<th>USVMCPY</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 - Copy a Selection Criteria Specification</td>
<td></td>
</tr>
<tr>
<td>2 - Copy a JCL Member</td>
<td></td>
</tr>
</tbody>
</table>

Enter Option, From User-id, and optional Name(s), press ENTER

Copy from User-id ===> userid

Specification/JCL

Name ===> DEPT-TEACHER (blank for member list)

Newname ==> NEW-DEPT-TEACHER

Operations

This section covers CA IDMS Extractor operations. It describes the CA IDMS Extractor system requirements, storage requirements, and recovery procedures.

Before running CA IDMS Extractor, consider these operational factors:

- All CA IDMS conventions regarding locking of the source and target databases are used. To ensure accurate extraction of data from the source database while CA IDMS Extractor is executing, you may want to prevent other jobs from updating the source database.

- You may either load a new database (that is, a database that was just formatted) or update an existing database. If you update an existing database, unexpected problems may arise if an attempt is made to store or connect a duplicate record unless the store or connect is prevented by a DUPLICATES NOT ALLOWED clause.

- The target subschema must reside in a dictionary load area. The Database Load Component of CA IDMS Extractor requires all user-defined and non-system owned integrated index sets have a set CONNECT option of MANUAL and a set ORDER of NEXT or SORTED. Further, if a SORTED set allows duplicates, and the source database was created with duplicates FIRST, the target database must be created with DUPLICATES LAST. To assist in complying with these requirements, a transient (temporary) subschema is created from the specified target so that you do not have to create a special target schema/subschema for CA IDMS Extractor use. The transient subschema is...
added to the target dictionary, used by the Database Load Component, and subsequently deleted from the target dictionary. Optionally, the transient subschema may be linked into a load library that is defined as a STEPLIB for the Database Load Component.

- CA IDMS Extractor Requirements (see page 149)
- CA IDMS Journal Analyzer Control Statements (see page 157)
- z/OS Operation (see page 157)
- z/VSE Operation (see page 160)
- Sample z/VM EXEC--USJEXEC (see page 164)
- Recovery Procedure (see page 167)
- Executing CA IDMS Extractor (see page 167)
- PROCESS Parameter (see page 204)
- Performing the EXTRACT and LOAD Steps--USVEXEC (see page 207)
- Printing JCL--USVPJCL (see page 222)
- Printing a Selection Criteria Specification--USVPSC (see page 226)
- Printing Online Documentation--USVPRINT (see page 230)
- Online Documentation Modules (see page 231)
- Customizing the CA IDMS Extractor Environment--USVTPARM (see page 232)
- Writing a User Exit Module--CUVUSRXA (see page 234)
- Syntax Notation (see page 237)

CA IDMS Extractor Requirements

The system requirements and storage requirements for executing both the Online and Batch Components of CA IDMS Extractor are outlined below.

System Requirements

The system requirements of CA IDMS Extractor are as follows:

CA CIS Requirements for CA IDMS

The following CA CIS (CA Common Infrastructure Services) components must be installed prior to the installation of CA IDMS:

⚠️ **Note:** If you are installing only the CA IDMS CMS Option in order to access a z/OS or z/VSE back end environment, it is not necessary to install any of the Common Services.

- CAISSF Standard Security Facility
  - Required by CA IDMS. security implementation
  - Required during installation as well as runtime
CA IDMS - 19.0

- CAICCI Common Communications Interface
  - Required if installing CA IDBGs.

The CA CIS software is installed using CA Activator. If this is the first time you have installed CA CIS, the initial installation process only offloads the CA Activator software; it will not install any of the actual CA CIS applications.

CA CIS expects all other software packages on the z/VM platform to also use CA Activator for their installation. Each package would then 'request' CA Activator to tell CA CIS which of its products were required.

CA IDMS. does not use CA Activator for its installation. In order to simulate this 'request', an EXEC, CAE5ACT, is provided on the installation file which is downloaded at the beginning of the installation process. See Complete Installation of CA CIS Software (https://docops.ca.com/display/IDMS19/Complete+Installation+of+CA+CIS+Software) for information on when to run this EXEC.

⚠️ **Note:** Access to the CA CIS software is required to use the CA IDMS. software. You must be linked to the CA CIS software installation minidisks. If this is not done, you may receive S-806 (or other) abends when CA IDMS. makes calls to the CA CIS software and the system is unable to locate the required CA CIS modules, such as CAS9SEC.

### CA IDMS System Requirements

The following list contains the CA IDMS. system requirements:

- DASD requirements
  - 472 cylinders (default configuration)
  - More or less depending on products installed
  - Each site may tailor database disk sizes but should not make smaller

- Tape drive requirement
  - 3480 cartridge tape drive

⚠️ **Note:** Use of a physical tape is optional if you use the Electronic file delivery option.

- Any release of COBOL and TXTLIB (optional)

⚠️ **Note:** TXTLIB is required only for the installation of the demonstration database.
CA IDMS - 19.0

- Assembler
- Separate user ID for CA IDMS. CV machine
  - 16M+ region
    (default SYSGEN requires 20M)
  - MAXCONN should be at least 50M for database access
  - IUCV authorization for ALLOW and ANY

Environments Supported

These lists contain the IBM mainframe computer systems and IBM operating systems supported by the CA IDMS installation.

Mainframe systems

- S/390
- zSeries 900 enterprise server
- Or any mainframe computer system that is plug-compatible with the above

Operating systems

- z/VM Release V5R3 or higher

Minidisk Space Requirements

Product Install Minidisk

The product install minidisk requires approximately 200-225 cylinders depending on the products selected. The following files reside on the install disk:

- IDMSLIB MACLIB
- DBALIB LOADLIB
- IDMSLIB LOADLIB
- TEXT modules
- SYSLIN members
- Install EXECs
- APARs and APAR control files

⚠️ **Note:** If you plan to install CA IDTOOL on the same minidisk as CA IDMS, it should be taken into account in your initial allocation.
Database Minidisks

Minidisk parameters are customized for ALL files:

Page size - 1024, 2048, 4096, or FBA

• File name - used for FILEDEFS or dynamic allocation
• Starting page number
• Number of pages

Total cylinders required (using defaults) if all products installed - 272 cylinders

⚠️ Note: If you change the number of pages, the installation verifies there are no page overlaps before creating your DMCL. If page overlaps are reported, you must correct them. This includes:

• Required Database Files (see page 152)
• Demo Database Files (only if installing the demonstration database) (see page 153)
• ASF Database Files (ASF Option only) (see page 153)
• SQL Database Files (SQL Option only) (see page 154)

Required Database Files

This table contains space requirements for required database files. You may change:

• Minidisk addresses
• Page ranges
• And/or the number of pages

for all files during the install process as dictated by your site requirements.

<table>
<thead>
<tr>
<th>File name</th>
<th>Disk address</th>
<th>Page size</th>
<th>Starting page number</th>
<th>Number pages (3380)</th>
<th>Number pages (3390)</th>
<th>Number cylinders</th>
</tr>
</thead>
<tbody>
<tr>
<td>DCCAT</td>
<td>500</td>
<td>4096</td>
<td>1</td>
<td>592</td>
<td>1432</td>
<td>4</td>
</tr>
<tr>
<td>DCCATX</td>
<td>501</td>
<td>4096</td>
<td>801</td>
<td>142</td>
<td>172</td>
<td>1</td>
</tr>
<tr>
<td>DCCATL</td>
<td>502</td>
<td>4096</td>
<td>1001</td>
<td>142</td>
<td>172</td>
<td>1</td>
</tr>
<tr>
<td>DCDML</td>
<td>503</td>
<td>4096</td>
<td>2001</td>
<td>1040</td>
<td>1250</td>
<td>7</td>
</tr>
<tr>
<td>DCLUD</td>
<td>504</td>
<td>4096</td>
<td>3501</td>
<td>142</td>
<td>172</td>
<td>1</td>
</tr>
</tbody>
</table>
Demo Database Files (only if installing the demonstration database)

The following table contains minidisk space requirements for the Commonweather database files.

<table>
<thead>
<tr>
<th>File name</th>
<th>Disk address</th>
<th>Page size</th>
<th>Starting page number</th>
<th>Number pages (3380)</th>
<th>Number pages (3390)</th>
<th>Number cylinders</th>
</tr>
</thead>
<tbody>
<tr>
<td>EMPDE</td>
<td>511</td>
<td>4096</td>
<td>75001</td>
<td>142</td>
<td>172</td>
<td>1</td>
</tr>
<tr>
<td>INSDEMO</td>
<td>512</td>
<td>4096</td>
<td>75201</td>
<td>142</td>
<td>172</td>
<td>1</td>
</tr>
<tr>
<td>ORGDEMO</td>
<td>513</td>
<td>4096</td>
<td>75401</td>
<td>142</td>
<td>172</td>
<td>1</td>
</tr>
</tbody>
</table>

Subtotal : 3

ASF Database Files (ASF Option only)

The following table contains minidisk space requirements for ASF database files.

⚠️ **Note:** Allocation of new ASF files is optional. ASF users may choose to use existing files and only install the ASF software.

<table>
<thead>
<tr>
<th>File name</th>
<th>Disk address</th>
<th>Page size</th>
<th>Starting page number</th>
<th>Number pages (3380)</th>
<th>Number pages (3390)</th>
<th>Number cylinders</th>
</tr>
</thead>
<tbody>
<tr>
<td>ASFDML</td>
<td>518</td>
<td>4096</td>
<td>80001</td>
<td>2089</td>
<td>2509</td>
<td>14</td>
</tr>
<tr>
<td>ASFLOD</td>
<td>519</td>
<td>4096</td>
<td>83001</td>
<td>1040</td>
<td>1250</td>
<td>7</td>
</tr>
</tbody>
</table>
SQL Database Files (SQL Option only)

The following table contains minidisk space requirements for SQL database files.

<table>
<thead>
<tr>
<th>File name</th>
<th>Disk address</th>
<th>Page size</th>
<th>Starting page number</th>
<th>Number pages (3380)</th>
<th>Number pages (3390)</th>
<th>Number cylinders</th>
</tr>
</thead>
<tbody>
<tr>
<td>SQLDD</td>
<td>50E</td>
<td>4096</td>
<td>20001</td>
<td>2089</td>
<td>2509</td>
<td>14</td>
</tr>
<tr>
<td>SQLLOD</td>
<td>50F</td>
<td>4096</td>
<td>28001</td>
<td>592</td>
<td>712</td>
<td>4</td>
</tr>
<tr>
<td>SQLXDD</td>
<td>510</td>
<td>4096</td>
<td>27001</td>
<td>592</td>
<td>712</td>
<td>4</td>
</tr>
<tr>
<td>EMPLDE</td>
<td>MO 514</td>
<td>4096</td>
<td>77001</td>
<td>142</td>
<td>172</td>
<td>1</td>
</tr>
<tr>
<td>INFODE</td>
<td>MO 515</td>
<td>4096</td>
<td>77201</td>
<td>142</td>
<td>172</td>
<td>1</td>
</tr>
<tr>
<td>INDXDE</td>
<td>MO 516</td>
<td>4096</td>
<td>77401</td>
<td>142</td>
<td>172</td>
<td>1</td>
</tr>
<tr>
<td>PROJDE</td>
<td>MO 517</td>
<td>4096</td>
<td>77601</td>
<td>142</td>
<td>172</td>
<td>1</td>
</tr>
<tr>
<td>Subtotal:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>26</td>
</tr>
</tbody>
</table>

Journal Database Files

A minimum of two journal files are required. The following table contains minidisk space requirements for journal database files.

<table>
<thead>
<tr>
<th>File name</th>
<th>Disk address</th>
<th>Page size</th>
<th>Starting page number</th>
<th>Number pages (3380)</th>
<th>Number pages (3390)</th>
<th>Number cylinders</th>
</tr>
</thead>
<tbody>
<tr>
<td>J1JRNL</td>
<td>51C</td>
<td>2048</td>
<td>N/A</td>
<td>5112</td>
<td>5964</td>
<td>19</td>
</tr>
<tr>
<td>J2JRNL</td>
<td>51D</td>
<td>2048</td>
<td>N/A</td>
<td>5112</td>
<td>5964</td>
<td>19</td>
</tr>
<tr>
<td>J3JRNL</td>
<td>51E</td>
<td>2048</td>
<td>N/A</td>
<td>5112</td>
<td>5964</td>
<td>19</td>
</tr>
<tr>
<td>J4JRNL</td>
<td>51F</td>
<td>2048</td>
<td>N/A</td>
<td>5112</td>
<td>5964</td>
<td>19</td>
</tr>
<tr>
<td>Subtotal:</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>76</td>
</tr>
</tbody>
</table>

Operating Environment

CA IDMS Extractor supports the following product releases:

- CA IDMS Release 16.0
Terminal Type

The online components of CA IDMS Extractor can be used from any 3270-compatible terminal, model 2, 3, 4, or 5.

Security

Normal CA IDMS/DC security applies when using the Online Components of CA IDMS Extractor. You can access only selection criteria specifications and JCL members that you create during the current session, or that you have previously created and saved. Your ability to access to other users' Selection Criteria Specifications and JCL members is controlled using the CA IDMS Extractor customization macro described in this section.

Online Storage Requirements

CA IDMS Extractor requires program storage, working storage, scratch storage, and queue storage.

Program Storage

All CA IDMS Extractor modules are reentrant. Therefore, one copy supports multiple users. All modules, except modules related to the following functions, are pseudo-conversational:

- Recovery
- Area/Record/Index/Set Deselection
- Online Documentation

The total program storage requirement for the CA IDMS Extractor online modules is approximately 750K. In addition, the Edit JCL function, which uses the CA EDITOR, requires 120K of program storage.

Working Storage

Each user that is currently executing a command uses between 10K and 20K of working storage, depending upon the function being performed. In addition, each active user shares the program storage, which is noted above.

Scratch Storage

Scratch records are used during the session to store information about the executing task. Approximately 4K of scratch area are used per task. The EDIT JCL function uses an additional 1K of scratch area for every 10 lines of source code.

Queue Storage

CA IDMS Extractor does not use CA IDMS/DC queue storage.
Batch Storage Requirements

CA IDMS Extractor batch storage requirements depend on the following factors:

- The size of the source and target subschemas.
- The size of the largest database record included in a database path.
- The sort storage requirements.
- Whether the components are run under a central version (CV) or in local mode.

The tables in the Batch Storage Requirements--EXTRACT Step (see page 156) and Batch Storage Requirements--LOAD Step (see page 156) sections show the amount of batch storage that is required for the Database Extract and Database Load Components, respectively, when executing the components under a CV. If the components are to run in local mode, additional storage is required for:

- CA IDMS runtime system modules
- Journal and database buffers.

### Batch Storage Requirements--EXTRACT Step

<table>
<thead>
<tr>
<th>Category</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Database Extract Component</td>
<td>250K - size of the largest module loaded into virtual storage at one time for the Batch Database Extract Component.</td>
</tr>
<tr>
<td>File Buffers</td>
<td>Size of the operating system buffers for the files used by the Database Extract Component.</td>
</tr>
<tr>
<td>Pseudo Subschema</td>
<td>Approximately 3 times the size of the source or target subschemas.</td>
</tr>
<tr>
<td>Extract Record</td>
<td>Size of the largest database record being extracted plus 28 bytes, or 256 bytes, whichever is greater.</td>
</tr>
<tr>
<td>Set Stack Area</td>
<td>8K - size of area for saving set information when walking sets.</td>
</tr>
</tbody>
</table>

### Batch Storage Requirements--LOAD Step

<table>
<thead>
<tr>
<th>Category</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Database Load Component</td>
<td>100K - size of the largest modules loaded into virtual storage at one time for the batch Database Load Component.</td>
</tr>
<tr>
<td>File Buffers</td>
<td>Size of the operating system buffers for the files used by the Database Load Component.</td>
</tr>
<tr>
<td>Pseudo Subschema</td>
<td>Approximately 3 times the size of the target subschema.</td>
</tr>
<tr>
<td>Extract Record</td>
<td>Size of the largest database record being extracted plus 28 bytes, or 256 bytes, whichever is greater.</td>
</tr>
<tr>
<td>Sort Storage</td>
<td>Use you installation's default sort storage.</td>
</tr>
</tbody>
</table>
CA IDMS Journal Analyzer Control Statements

CA IDMS Journal Analyzer produces reports and displays in a large variety of combinations. Chapter 5 ([https://docops.ca.com/display/IDMS19/Examples](https://docops.ca.com/display/IDMS19/Examples)) shows sets of parameter statements that can be used in various situations.

The sample parameter statements provided below are recommended for an evaluation of CA IDMS Journal Analyzer. They can be used in any environment--z/OS, z/VSE, or z/VM.

P=ALL, CONT=N, DMCL=DBMSDMCL, F=SPARSE, RHDC=N, NON=Y
R=ACTV, ALL=Y, START=0101820000, I=15
R=PROG, ALL=Y, LEV=SUM, HILITES=Y
R=CHRONO, ALL=Y
R=ABORTC, ALL=Y
R=HI-SUM, LEV=PROG
R=HI-ONL
R=HI-BPU
R=RANK, RANKH=HIGH, RANKW=#RU
R=RANK, RANKH=HIGH, RANKW=DURATION

z/OS Operation

The operation of CA IDMS Journal Analyzer in a z/OS environment can be accomplished in a single job step. CA IDMS Journal Analyzer processes display images up to 32K in length.

Execution JCL (z/OS)

The following example shows the execution JCL for CA IDMS Journal Analyzer, where variables (lowercase, bold) are explained in the key opposite the JCL:

```plaintext
//JNLA       EXEC    PGM=USJMAIN,REGION=K
//STEPLIB     DD      DSN=idms.dba.loadlib,DISP=SHR
//        DD      DSN=idms.custom.loadlib,DISP=SHR
//        DD      DSN=idms.cagjload,DISP=SHR
//SYSIDMS     DD      *
//SYSOUT     DD      ECHO=ON
//JOURNAL   = OFF
//LOCAL     = on-or-off
//        DD      TESTD=OFF
//SYSOUT     DD      BN=on-or-off
//DBN       dd      DBN=dbname//dictdb dd dsn=your.idms.dictionary,disp=shr
//        DD      SYSOUT=a//JASRTMSG
//        DD      UT=a
//SYSLSTR     DD      DBC=BLKSIZE=1330
//        DD      DBC=BLKSIZE=1330
//SYSLSTD     DD      DBN=your.idms.dictionary,disp=shr
//        DD      DBN=dbname//dictdb dd dsn=your.idms.dictionary,disp=shr
//        DD      SYSOUT=r
//SORTWK01    DD      DBC=BLKSIZE=1330
//        DD      DBC=BLKSIZE=1330
//SORTWK02    DD      DBC=BLKSIZE=1330
//        DD      DBC=BLKSIZE=1330
//SORTWK03    DD      DBC=BLKSIZE=1330
//        DD      DBC=BLKSIZE=1330
//SORTWK04    DD      DBC=BLKSIZE=1330
//        DD      DBC=BLKSIZE=1330
//SORTWK05    DD      DBC=BLKSIZE=1330
//        DD      DBC=BLKSIZE=1330
```

10-Jan-2018 157/240
The variables in the execution JCL are:

- \#
  Defines the number of kilobytes of storage to be made available. The region size is between 1200K and 1800K, depending on CA IDMS Journal Analyzer's mode of accessing IDMS (Local Mode or CV), the BUFNOs specified for the various files, and the SIZE option of your sort.

- idms.dba.loadlib (required for Local Mode)
  Defines the dataset name of the library that contains the DMCL used by your dictionary subschema, IDMSNWKA. Omit if running through CV.

- idms.custom.loadlib
  Defines the dataset name of the library that contains customized load modules.

- idms.cagjload
  Defines the dataset name of the library that contains CA IDMS Journal Analyzer modules and IDMSINTB.

- dbname
  Defines the DBNAME needed to access the appropriate DDLDM area of the CA IDMS dictionary for the DMCL name used.

- on-or-off
  Specify LOCAL=ON if you want Display processing to access the CA IDMS dictionary in LOCAL mode. Specify LOCAL=OFF for Central Version (CV) mode. Specify TESTDBN=ON if you want Display processing to access a secondary dictionary. Specify TESTDBN=OFF for the primary dictionary.

- dmclname
  Defines the DMCLNAME to be used when Display processing accesses the CA IDMS dictionary.

- dictdb dd dsn=your.idms.dictionary,disp=shr (required for Local Mode)
  Specify the correct DDNAME and dataset name of your CA IDMS dictionary's DDLDM area. Omit this statement if running through CV, or if the correct information is supplied in the DMCL.

- a
  An appropriate SYSOUT class for your installation.
- **sort-message**
  To print sort messages, specify **SYSOUT=a**. To suppress printing of sort messages, specify **DUMMY**.

- **r**
  An appropriate SYSOUT class for Journal Reports. This should be different from **a**. If **SYSLSTR** is assigned to a file rather than to **SYSOUT**, specify **DISP=MOD** to ensure that Management Reports do not overlay other reports.

- **d**
  An appropriate SYSOUT CLASS FOR Journal Displays. This should be different from **a**.

- **sort-space**
  A space specification appropriate to the number of records to be sorted. If **PROCESS=EXTRACTS**, provide space to sort the entire extract file. If **PROCESS=DISPLAYS**, provide space to sort the entire display file. If **PROCESS=ALL**, provide space to sort the larger of the display file or the extract file. If **PROCESS=REPORTS**, provide space to sort the extracted records for a single report interval.

- **your.archive.journal**
  Defines the dataset name of your archive journal file. This file should come from the **FIX ARCHIVE** utility, or from a batch job that ran in Local Mode. Do not use the output from **IDMSJFIX**. (You may specify **NULLFILE** if **PROCESS=REPORTS**.)

- **blksize=archive-blksize**
  Defines the current block size of your archive journal file. (This parameter may be omitted if your file has standard labels.)

- **unit=tape,label=(,sl),vol=ser=serial-number**
  Defines the proper unit, label, and volume-serial information for your archive journal file. (These parameters may be omitted for a cataloged file.)

- **dsn=your.extract.file**
  Defines the dataset name of the extract file. This parameter may be omitted, and a temporary file created, if **PROCESS=DISPLAY**, or if **PROCESS=ALL** and you do not intend to run **PROCESS=REPORTS** at a later time. If **PROCESS=REPORTS**, do not concatenate extract files. Multiple extract files must be sorted or merged on the first 46 bytes of the record before being input to CA IDMS Journal Analyzer.

- **disk**
  Defines the device type of the extract, display, and ranking work files. (If this is a tape device, the following SPACE parameter should be omitted.)

- **extract-space**
  If **PROCESS=ALL** or **PROCESS=EXTRACTS**, CA IDMS Journal Analyzer extracts archive records for each report type that is not suppressed. All reports use the **BGIN**, **AREA**, **COMT**, **ABRT**, and **ENDJ** records present for each run-unit. **DBAREA** and **DBRECD** reports also use the **BFOR** and **AFTR** records present for each run-unit. The amount of space depends on the number of run-units in the archive file, the number of areas readied, and the number of commits issued by those run-units (and if **DBAREA** and **DBRECD** reports are not suppressed, the number of database records processed by each run-unit).
• **disp=extract-disp**
  Defines a disposition for the extract file. If PROCESS=EXTRACTS, specify DISP=(NEW,CATLG). If PROCESS=REPORTS, specify DISP=OLD. (This parameter should be omitted if the extract dataset name was omitted.)

• **display-space**
  If journal displays are specified, CA IDMS Journal Analyzer writes a record to the display file for each BFOR/AFTR record that meets selection criteria for DISPLAY=DBKEY or for DISPLAY=RECORD and for each BFOR/AFTR record for every run-unit that meets selection criteria for DISPLAY=PROGRAM. All ABRT records are also written for the displays specified. The amount of space depends on the number and size of these BFOR/AFTR/ABRT records.

• **display-blksize**
  Specifies the BLKSIZE for the DISPLAY file. Specify a value that contains at least one occurrence of your largest DISPLAY record plus a 4-byte block descriptor word. (A DISPLAY record contains a 4-byte record descriptor word, an 84-byte header, and a database record image.)

• **ranking-space**
  CA IDMS Journal Analyzer writes a record to the ranking file for every run-unit that meets the selection criteria of each REPORT=RANK specification. The amount of space depends on the number of rank reports specified and the number of run-units that satisfy the selection criteria.

• **sortlib dd dsn=sysl.sortlib,disp=shr**
  This statement is optional. Its specification is dependent on your installation.

• **sortcntl dd ***
  This statement is optional (see below). Consult the programmer’s guide of your sort utility to determine the DDNAME of this file.

• **your-sort-control-statements**
  These statements are optional. Include them to override the options of your sort utility. (CA IDMS Journal Analyzer should be run with sort options that specify printing of critical messages, setting a return code of 16 if there is an unsuccessful sort, and sorting into EBCDIC sequence.) You may also want to specify a size different from the default. If you specify maximum size, further ensure that you have the sort use all but 256K (that is, SIZE=MAX-256) so that a BIND RUN-UNIT can be processed while sorting. Consult the programmer’s guide of your sort utility to determine the format of these statements.

• **your-journal-analyzer-control-statements**
  Specify a PROCESS statement and other statements for the reports and displays that you want generated. You can find more information in Chapter 5 (https://docops.ca.com/display/IDMS19/Examples).

---

**z/VSE Operation**

The z/VSE operation of CA IDMS Journal Analyzer can be accomplished in a single job step. Sample JCL is contained in TOOLJCL library member USJEXEC.S. The JCL and key are shown below.
z/VSE File Assignments

Even if you use a storage management tool such as CA-DYNAM, CA IDMS Journal Analyzer requires an ASSGN statement for every file except SORTWKnn. This ASSGN is required because CA IDMS Journal Analyzer has its own device-independent support which dynamically builds a DTF based on the device type indicated by the ASSGN. Unless the ASSIGN specifies VSAM or BDAM, the file may be defined with either DLBL or TLBL.

Execution JCL (z/VSE)

```jcl
// OPTION PARTDUMP, LOG
* **** CREATE A SYSIDMS PARAMETER FILE ****
// UPSI 1
// DLBL anyname,'work.file.sysidms',0,SD
// EXTENT SYS041,sysids-extent// ASSGN SYS041,DISK,VOL=volser,SHR
// EXEC DITTO
$$DITTO CSQ FILEOUT=anyname ECHO=ON
JOURNAL=OFF
LOCAL=ON-OR-OFF
TESTDBN=ON-OR-OFF
DMCL=dmlcename DBN=dbname/*
/* CARD AND PRINT FILES
// ASSGN SYS016,SYSIPT
// ASSGN SYS017,SYSLST
// ASSGN SYS018,printer
// ASSGN SYS019,printer
/*
/* IDMS FILES
// UPSI upsi-byte// ASSGN SYS005,DISK// DLBL DICTDB,'idms-dictionary',DA
// EXTENT SYS005,dictionary-extent// ASSGN SYS009,ign/* SORTWORK FILES FOLLOW
// ASSGN SYS001,DISK,VOL=volser,SHR
// DLBL SORTWK1,'sort.work.1',0,SD
// EXTENT SYS001,sort-extent// ASSGN SYS002,DISK,VOL=volser,SHR
// DLBL SORTWK2,'sort.work.2',0,SD
// EXTENT SYS002,sort-extent// ASSGN SYS003,DISK,VOL=volser,SHR
// DLBL SORTWK3,'sort.work.3',0,SD
// EXTENT SYS003,sort-extent// ASSGN SYS004,DISK,VOL=volser,SHR
// DLBL SORTWK4,'sort.work.4',0,SD
// EXTENT SYS004,sort-extent/*
/* Journal Analyzer WORK FILES
// ASSGN SYS012,DISK,VOL=volser,SHR
// DLBL EXTRACT,'jnla.extract',0,SD EXTRACT FILE - OUTPUT
// EXTENT SYS012,extract-extent// DLBL EXTRAC1,'jnla.extract',0,
SD EXTRACT FILE - INPUT
// EXTENT SYS012
// ASSGN SYS013,DISK,VOL=volser,SHR
// DLBL DISPLAY,'jnla.display',0,SD DISPLAY FILE - OUTPUT
// EXTENT SYS013,display-extent// DLBL DISPLAY1,'jnla.display',0,
SD DISPLAY FILE - INPUT
// EXTENT SYS013
// ASSGN SYS014,DISK,VOL=volser,SHR
// DLBL RANKING,'jnla.ranking',0,SD RANKING FILE - OUTPUT
// EXTENT SYS014,ranking-extent// DLBL RANKIN1,'jnla.ranking',0,
SD RANKING FILE - INPUT
// EXTENT SYS014
/* ARCHIVE JOURNAL FILE
// ASSGN SYS011,TAPE// TLBL ARCHIVE,'archive.journal',,volser ARCHIVE JOURNAL FILE
/* PRIVATE CORE IMAGE LIBRARY(S)
// DLBL CILIB1,'your-journal-corelib' Journal Analyzer LIBRARY
```
An appropriate unit address/device class for printing reports and displays. This should be different than SYSLST.

Any suitable name for your SYSIDMS file. The name chosen must be identical to the fileout value in the DITTO copy step.

Specify extent information for the SYSIDMS file. One track is adequate.

The UPSI byte settings appropriate to the IDMSOPTI module that you have linked with CA IDMS Journal Analyzer, and appropriate to whether you will be accessing CA IDMS in local mode or central version.

Specify the proper ASSGN information for logical unit SYS005, your CA IDMS dictionary, if running in local mode. Omit this statement if running through central version.

Required for local mode processing. Specify the filename and file ID of your local CA IDMS dictionary's DDL DML area. Omit this statement if running through central version.

Specify the proper extent information for logical unit SYS005, your IDMS dictionary, if running in local mode. Omit this statement if running through central version.

Specify an ASSGN for logical unit SYS009, your CA IDMS local journal file, if running in local mode. Since CA IDMS Journal Analyzer accesses your CA IDMS dictionary in retrieval mode only, it is necessary to journal that activity. Omit this statement if running through central version.

Specify the file ID of your sortwork files. Note that CA IDMS Journal Analyzer should be run with sort options that specify printing of critical messages, not abending if there is an unsuccessful sort and sorting into EBCDIC sequence.

The volume serial number of the disk or tape to be used for the file.
- **jnla.extract**
  Specify the file ID of the extract file. This statement may specify 'IGN' if PROCESS=DISPLAY. The retention period for the extract file depends on your use of PROCESS options. Specify a retention period that will allow you to fully generate desired reports from the extract file. If you never intend to specify PROCESS=REPORTS, your retention period may define this as a temporary file. If you specify PROCESS=REPORTS, this file must already exist.

- **extract-extent**
  Specify extent information adequate to hold all of the records extracted from the archive journal file. If PROCESS=ALL, or PROCESS=EXTRACTS, is specified, CA IDMS Journal Analyzer extracts archive records for each report type that is not suppressed. All reports use the BGIN, AREA, COMT, ABRT, and ENDJ records present for each run-unit. DBAREA and DBRECD reports also use the BFOR and AFTR records present for each run-unit. The amount of space will depend on the number of run-units on the archive file, the number of areas readied, and the number of COMMITs issued by those run-units (and if DBAREA and DBRECD reports are not suppressed, the number of database records processed by each run-unit).

- **jnla.display**
  Specify the file ID of the display file. This statement may specify 'IGN' if you do not request journal displays. The display file is created each time you request journal displays, so the retention period may define this as a temporary file.

- **display-extent**
  Specify extent information adequate to hold all of the records selected from the archive journal file. If journal displays are requested, CA IDMS Journal Analyzer writes a record to the display file for each BFOR/AFTR record that meets selection criteria for DISPLAY=DBKEY or for DISPLAY=RECORD and for each BFOR/AFTR record for every run-unit that needs selection criteria for DISPLAY=PROGRAM. Each ABRT record will also be written for each display requested. The amount of space will depend on the number and size of these BFOR/AFTR/ABRT records.

- **jnla.ranking**
  Specify the file ID of the ranking file. This statement may specify 'IGN' if you do not request rank reports. The ranking file is created each time you request rank reports, so the retention period may define this as a temporary file.

- **ranking-extent**
  Specify extent information adequate to hold all of the records selected from the extract file if REPORT=RANK was specified. CA IDMS Journal Analyzer writes a record to the ranking file for every run-unit that meets the selection criteria for each REPORT=RANK request. The amount of space will depend on the number of requests for rank reports and the number of run-units that satisfy the selection criteria. Since the ranking file is created each time you request REPORT=RANK, the retention period may define this as a temporary file.

- **archive.journal**
  Specify the file ID of your archive journal file. This file should come from ARCHIVE JOURNAL utility, or from a batch job that ran in local mode. Do not use the output from IDMSJFIX. This statement may specify 'IGN' if PROCESS=REPORTS.

- **tape**
  Specify an appropriate unit address for your archive journal file. This must be a tape device. **your-journal-corelib** -- Specify the file ID of the core image library containing your IDMSINTB module.
The amount of your partition to reserve for program storage. The remainder of the partition is used for GETVIS. #k designates additional program storage to be made available for CA IDMS Journal Analyzer programs (about 800K) and for your sort. The total partition requirements will be between 1300K and 2100K.

**LOCAL ON/OFF** -- Specify LOCAL=ON if you want journal displays processing to access the CA IDMS dictionary in local mode. Specify LOCAL=OFF for CV mode.

**dmclname**
Specify the name of the DMCL that should be used when accessing the dictionary for information about records represented by the BFOR and AFTR images of the CA IDMS archive journal being processed.

**dbname**
Specify the DBNAME needed to access the appropriate DDLDM area of the dictionary for the DMCLNAME used.

**your.idms.dictionary**
Required for local mode when the correct data set name is not supplied in the DMCL. Specify the correct DDNAME and data set name for the DDLDM area of your CA IDMS dictionary.

**your-journal-analyzer-control-statements**
Specify a PROCESS statement and other statements as desired.

---

**Sample z/VM EXEC--USJEXEC**

A sample EXEC for executing CA IDMS Journal Analyzer is shown below. Variables (lowercase) are explained in the key below the EXEC.

```plaintext
Note: The PROCESS=DISPLAYS option should not be used in a z/VM/CMS environment. The Journal must be archived using the CA IDMS utility with a file mode of x4 to indicate z/OS file-type simulation, and a DCB of: (RECFM F LRECL 4096.

/* */
TRACE OFF; SIGNAL ON ERROR
/* */
CA LOADLIB FN = 'yourlib'
IDMS LOADLIB FN = 'idmslib'
SORTLIB FN = 'sortlib'
JNLA_ARCHIVE FN = 'your.journal.archive'
JNLA_ARCHIVE FT = 'filetype'
JNLA_ARCHIVE FM = '*:'
/* */
/* Link and access the Minidisks containing the required librarie(s) */
/* and database file(s). */
/* */
'CP SPPOOL PRINTER NOCONT CLOSE'
'CP SPPOOL PRINTER TO * NOHOLD CONT OFF DIST OFF'
'GLOBAL LOADLIB ' CA LOADLIB FN IDMS LOADLIB FN
GLOBAL TXTLIB ' SORTLIB FN
/* */
/* Dictionary and Journal file definitions. */
/* */
```

10-Jan-2018 164/240
/* When running in local mode remove the comments from around the */
/* next 4 statements. */

/*FILEDEF DICTDB  DISK  dictdb_fn  dictdb_ft  dictdb_fm  ( dcb extent'
 'FILEDEF SYSJRNL  DUMMY'
 'FILEDEF J1JRNL  DUMMY'
 'FILEDEF J2JRNL  DUMMY'
*/
/* Files needed for all runs. */

/*FILEDEF 3ASRTMSG  PRINTER'
 'FILEDEF SYSLST  PRINTER'
 'FILEDEF SYSUDUMP  PRINTER'
 'FILEDEF SYSOUT  PRINTER'
 'FILEDEF SYSLSTD DISK  JNLA  SYSLSTD.fm'
 'FILEDEF SYSLSTR DISK  JNLA  SYSLSTR.fm'
 'FILEDEF AUDIT  DISK  JNLA  AUDIT.fm'
 'FILEDEF EXTRACT DISK  JNLA  EXTRACT.fm'
 'FILEDEF DISPLAY DISK  JNLA  DISPLAY fm  (BLKSIZE  dddd'
 'FILEDEF RANKING DISK  JNLA  RANKING fm  (BLKSIZE  rrrr'
/* Archive Journal Disk File */

/*FILEDEF SYSIDMS DISK  SYSIDMS  INPUT  A'
/* You must create a file 'SYSIDMS  INPUT  A' containing the SYSIDMS */
/* parameters you use to specify your runtime environment. */
/*FILEDEF SYSIPT DISK  USJEXEC  SYSIPT  A'
/* Insert FILEDEF statements for SORT work space as required by */
/* your SORT product. */

SIGNAL OFF ERROR
SAY 'STARTING EXECUTION OF CA IDMS Journal Analyzer'
'EXECOS OSRUN USJMAIN'
USJEXEC RC = RC
'CP  SPOOL  PRINTER  NOCONT'
'CP  CLOSE  PRINTER  NAME  JNLA  LISTING'
'CP  SPOOL  PRINTER  OFF'
SAY 'USJEXEC FINISHED WITH A RETURN CODE OF' USJEXEC_RC
'GLOBAL  LOADLIB'
'GLOBAL  TXTLIB'
'FILEDEF  *  CLEAR'
EXIT USJEXEC_RC
/*ERROR:

ERROR_RC = RC
TRACE OFF; SIGNAL OFF ERROR
SAY 'NON-ZERO RETURN CODE ENCOUNTERED IN EXEC AT LINE' SIGL
'CP  SPOOL  PRINTER  NOCONT'
'CP  CLOSE  PRINTER  NAME  JNLA  LISTING'
'CP  SPOOL  PRINTER  OFF'
'GLOBAL  LOADLIB'
'GLOBAL  TXTLIB'
'FILEDEF  *  CLEAR'
EXIT ERROR_RC
*/
- **yourlib**
The file name of the load library into which you downloaded CA IDMS Journal Analyzer.

- **idmslib**
The file name of the load library containing your CA IDMS SUBSCHEMA and DMCL modules.

- **sortlib**
The file name of the text library containing your sort modules.

- **your.journal.archive**
The file name of your journal archive file. This file should come from the ARCHIVE JOURNAL utility, or from a batch job that ran in local mode. Do not use the output from the FIX ARCHIVE utility. If you request any Journal Displays, you should pre-sort the archive file so that each record is sequenced on RUNUNIT-ID and ARC-SEQUENCE-NBR. USJSORT may be used for this purpose.

- **filetype**
The file type of your journal archive file.

- **dictdb_fn**
The file name of your CA IDMS dictionary file.

- **dictdb_ft**
The file type of your CA IDMS dictionary file.

- **dictdb_fm**
The file mode of your CA IDMS dictionary file.

- **fm**
The file mode of the relevant file.

- **fn SYSCTL fm**
The file name, file type, and file mode of the SYSCTL file for the Central Version (CV) used during batch processing.

- **JNLA EXTRACT fm**
The EXTRACT file. CA IDMS Journal Analyzer writes extracts of Archive records for each report type that is not suppressed. All reports use the BGIN, AREA, COMT, ABRT, and ENDJ records present for each run-unit. The amount of space needed depends on the number of run-units in the Archive file, the number of Areas readied, and the number of Commits issued by those run-units.

- **JNLA DISPLAY fm**
The DISPLAY file. If Journal Displays are requested, CA IDMS Journal Analyzer writes a record to the Display File for each BFOR/AFTR record that meets the selection criteria for DISPLAY=DBKEY or for DISPLAY=RECORD, and for each BFOR/AFTR record for every run-unit that meets the selection criteria for DISPLAY=PROGRAM. Each ABRT record will also be written for each DISPLAY requested. The amount of space needed depends on the number and size of these records.
The RANKING file. CA IDMS Journal Analyzer writes a record to the Ranking File for every run-unit that meets the selection criteria of each REPORT=RANK request. The amount of space needed depends on the number of requests for RANK reports and the number of run-units that satisfy the selection criteria.

The file name, file type, and file mode of the Journal Display file. This should be different from PRINTER.

The file name, file type, and file mode of the Journal Reports file. This should be different from PRINTER.

The information required for your CA IDMS dictionary file.

The BLKSIZE for the Ranking File. Must be a multiple of 60 (LRECL).

The BLKSIZE for the DISPLAY file. Specify a value that will contain at least one occurrence of your largest DISPLAY record plus a 4-byte block descriptor word. (A DISPLAY record contains a 4-byte record descriptor word, an 84-byte header, and a database record image.)

Recovery Procedure

During a CA IDMS Extractor session, information is stored in the scratch area. While editing a JCL member, the member you are editing is also stored in the scratch area. In the event of an abend, you can recover the changes you have made since your last save only if the scratch area is left intact. If CA IDMS Extractor abends, the scratch area is unaffected and recovery is possible unless you were in the middle of an online Selection Criteria Specification session. If your system goes down, the scratch area is lost.

When an abend does occur, you are returned to the CA IDMS/DC prompt and a message is displayed. When you invoke CA IDMS Extractor again, you are returned to the point in the session when the abend occurred.

Executing CA IDMS Extractor

Once a Selection Criteria Specification is created, you must run the Database Extract Component to extract data from the source database and the Database Load Component to load the extracted data onto the target database.

To run the extract and load batch components, the model JCL and key for:

- z/OS-Target or Distribution source library member USVEXEC
CA IDMS - 19.0

- z/VSE- SAMPJCL library member USVEXEC.S
- z/VM- USVEXEC EXEC

To run the JCL Member Print Utility, the model JCL and key for:
- z/OS-Target or Distribution source library member USVPJCL
- z/VSE- SAMPJCL library member USVPJCL.S
- z/VM- USVPJCL EXEC

To run the Specification Print Utility, the model JCL and key for:
- z/OS-Target or Distribution source library member USVPSPC
- z/VSE- SAMPJCL library member USVPSPC.S
- z/VM- USVPSPC EXEC

The sample JCL and keys for these components and utilities are shown below.

Sample JCL

When CA IDMS Extractor is installed, sample JCL is loaded onto the CA IDMS Extractor database by USVUJCL. This sample JCL is associated with the user ID of DBXADMIN. When you access the Edit JCL Component, if you do not already have a copy of the sample JCL, it is copied to your user ID. You must edit the JCL, using the JCL Editor Component, to tailor it to your environment. You can copy the JCL member first, using the JCL Utilities Component, to a name of your choice so that the original model JCL is not modified.

Extract Information from the Source Dictionary - z/VSE

Contents

- Export Syntax Files--z/VSE (see page 171)
- Import the Extract Tape -- z/VSE (see page 174)
- Upload Syntax Files -- z/VSE (see page 177)
- Delete Migrated Source from Source Dictionary -- z/VSE (see page 184)
- Allocate the VSAM Work File--z/VM (see page 187)
- Extract Information from the Source Dictionary -- z/VM (see page 188)
- Export Syntax Files--z/VM (see page 190)
- Import the Extract Tape--z/VM (see page 192)
- Upload Syntax Files -- z/VM (see page 194)
- Delete Migrated Source from Source Dictionary - z/VM (see page 201)

The JCL in TOOUCJCL library member USMXTRCT.S (z/VSE) allocates the work files, allocates the syntax files, extracts information from the source dictionary, compares the extract to the object dictionary, reports on the comparison, and produces the syntax files.
----- Dictionary Migrator STEP ----- 'USMXTRCT' SAMPLE JCL ----- 

// UPSI 1 
// OPTION LOG, PARTDUMP 
// DLBL anyname,'work.file.SYSIDMS',0,SD 
// EXTENT SYS041, volser,, 948,1 
// ASSGN SYS041.DISK, VOL=volser, SHR 
// EXEC DITTO 
$DITTO CSQ FILEOUT=anyname 
* SYSIDMS parameters. 
* For Local Mode specify: 
* DMCL=dmcl-name, LOCAL=ON, JOURNAL=OFF 
FILENAME=SYSIPT, FILETYPE=D, BLKSIZE=80, DEVADDR=SYSIPT 
* 
* For CV runs specify: 
* DMCL=dmcl-name, LOCAL=OFF, JOURNAL=OFF 
FILENAME=SYSIPT, FILETYPE=D, BLKSIZE=80, DEVADDR=SYSIPT 
*/ 

* OPTION PARTDUMP 
* **** z/VSE library where CA IDMS Dictionary Migrator resides 
// DLBL DBMS,'your.corelib' 
// EXTENT ,volser 
* **** z/VSE library where CA IDMS/DB executable phases reside 
// DLBL DBMS,'idms.corelib' 
// EXTENT ,volser 
* 
// LIBDEF PHASE, SEARCH=(dbms.sublib, idms.sublib) 
* 
***** PARAMETER INPUT FOR DICT. MIGRATOR AND IDMS UTILITIES ***** 
* 
// ASSGN SYS011, SYSRDR SYNTAX FILE 
// ASSGN SYS010, SYSRCH HOLD SYSPCH ASSGN 
// DLBL IJSYSPH,'work.file.SYSIPT',0,SD IJSYSPH IS USED TO 
// EXTENT SYSPCH, volser,, 00100,050 CREATE VTOC ENTRY 
ASSGN SYSPCH, DISK, VOL=volser, SHR FOR SYSPCH ASSGN 
CLOSE SYSPCH, SYS010 
// DLBL IJSYSIN,'work.file.SYSIPT',0,SD SAME FILE-ID AND 
// EXTENT SYSPCH, volser,, 00100,050 TRACKS AS IJSYSPH 
ASSGN SYSPCH, DISK, VOL=volser, SHR 
// DLBL SYSPCH,'work.file.SYSIPT',0,SD SAME FILE-ID AND 
// EXTENT SYSPCH, volser,, 00100,050 TRACKS AS IJSYSPH 
ASSGN SYSPCH, DISK, VOL=volser, SHR 
* 
* OUTPUT FROM IDMS UTILITIES AND INPUT TO Dictionary Migrator *** 
* 
// ASSGN SYSPCH, IGN 
// DLBL SYSPCH,'work.file.SYSPCH',0,SD 
// EXTENT SYSPCH, volser,, 00150,200 SAME FILE-ID AND 
// DLBL IDMSPCH,'work.file.SYSPCH',0,SD 
// EXTENT SYSPCH, volser,, 00150,200 TRACKS AS SYSPCH 
// ASSGN SYSPCH, DISK, VOL=volser, SHR 
* 
*************** REPORT FILE *********************** 
* 
// ASSGN SYS013, SYSLST 
* 
************ Dictionary Migrator WORK FILES *********************** 
* 
// DLBL EXTRACT,'work.file.EXTRACT',0,SD 
// EXTENT SYS015, volser,, 00350,200 
// ASSGN SYS015, DISK, VOL=volser, SHR 
// DLBL VSAMEXT,'vsam.file.VSAMEXT',0,VSAM 
// EXTENT SYS016, volser 
// ASSGN SYS016, DISK, VOL=volser, SHR 
// DLBL SELECT,'work.file.SELECT',0,SD 
// EXTENT SYS014, volser,, 00550,003
/*
 * * SYNTAX FILES FROM DICT MIGRATOR FOR IDMS UTILITIES **********
 *
 * DLBL RHDCDEL,'SYNTAX.FILE.RHDCDEL',0,SD MAP & PANEL DELETES
 // EXTENT SYS019,volser,,00760,005
 // ASSGN SYS019,DISK,VOL=volser,SHR
 // DLBL SCHMDEL,'SYNTAX.FILE.SCHMDEL',0,SD SCHEMA DELETES
 // EXTENT SYS020,volser,,00765,005
 // ASSGN SYS020,DISK,VOL=volser,SHR
 // DLBL SUBSDEL,'SYNTAX.FILE.SUBSDEL',0,SD SUBSCHEMA DELETES
 // EXTENT SYS021,volser,,00770,005
 // ASSGN SYS021,DISK,VOL=volser,SHR
 // DLBL DDDLDEL,'SYNTAX.FILE.DDDLDEL',0,SD ENTITY DELETES
 // EXTENT SYS022,volser,,00775,100
 // ASSGN SYS022,DISK,VOL=volser,SHR
 // DLBL DDDLUPD,'SYNTAX.FILE.DDDLUPD',0,SD ENTITY ADD/MODIFIES
 // EXTENT SYS023,volser,,00785,100
 // ASSGN SYS023,DISK,VOL=volser,SHR
 // DLBL BCFUPD,'SYNTAX.FILE.BCFUPD',0,SD SQL UPDATES
 // EXTENT SYS040,volser,,00950,005
 // ASSGN SYS040,DISK,VOL=volser,SHR
 // DLBL TABLLOD,'SYNTAX.FILE.TABLLOD',0,SD TABLE LOAD MODULES
 // EXTENT SYS024,volser,,00885,005
 // ASSGN SYS024,DISK,VOL=volser,SHR
 // DLBL SCHMUPD,'SYNTAX.FILE.SCHMUPD',0,SD ADD SCHEMAS
 // EXTENT SYS026,volser,,00895,005
 // ASSGN SYS026,DISK,VOL=volser,SHR
 // DLBL SUBSUPD,'SYNTAX.FILE.SUBSUPD',0,SD ADD SUBSCHEMA
 // EXTENT SYS028,volser,,00905,005
 // ASSGN SYS028,DISK,VOL=volser,SHR
 // DLBL SUBSLOD,'SYNTAX.FILE.SUBSLOD',0,SD REGENERATE SUBSCHEMAS
 // EXTENT SYS029,volser,,00910,005
 // ASSGN SYS029,DISK,VOL=volser,SHR
 // DLBL RHDCUPD,'SYNTAX.FILE.RHDCUPD',0,SD ADD PANELS & MAPS
 // EXTENT SYS030,volser,,00915,005
 // ASSGN SYS030,DISK,VOL=volser,SHR
 // DLBL RHDCLOD,'SYNTAX.FILE.RHDCLOD',0,SD RECOMPILE MAPS
 // EXTENT SYS031,volser,,00920,005
 // ASSGN SYS031,DISK,VOL=volser,SHR
 // DLBL ADSOBN,'SYNTAX.FILE.ADSOBN',0,SD REGENERATE DIALOGS
 // EXTENT SYS032,volser,,00930,005
 // ASSGN SYS032,DISK,VOL=volser,SHR
 // DLBL DDLPGM,'SYNTAX.FILE.DDDLPGM',0,SD ADD PROGRAM
 // EXTENT SYS033,volser,,00925,005
 // ASSGN SYS033,DISK,VOL=volser,SHR
 // DLBL ADSBTAT,'SYNTAX.FILE.ADSBTAT',0,SD REGENERATE ADSA
 // EXTENT SYS037,volser,,00942,003
 // ASSGN SYS037,DISK,VOL=volser,SHR
 // DLBL USERUPD,'SYNTAX.FILE.USERUPD',0,SD PASSWORD MIGRATION
 // EXTENT SYS038,volser,,00945,003
 // ASSGN SYS038,DISK,VOL=volser,SHR
 */

* ******* SYNTAX FILES FROM DICT MIGRATOR FOR IDMS UTILITIES *******

* ******* SYNTAX FILES FROM DICT MIGRATOR FOR IDMS UTILITIES *******

* ********** SYNTAX FILES FROM DICT MIGRATOR FOR IDMS UTILITIES **********

* PROCESS,
your.corelib -- The file id of the z/VSE library into which CA IDMS Dictionary Migrator was installed.

idms.corelib -- The file id of the z/VSE library which contains the CA IDMS utility executable phases.

volser -- The volume serial number or generic assignment of the disk volume on which the library, as specified in the previous statement, resides.

.sublib -- The name of the sublibrary of the z/VSE library specified by the previous filename.

vsam.file -- It is recommended that you retain the DLBL name VSAMEXT within the vsam file name. Replace vsam.file with a name suitable to your site.

work.file -- It is recommended that you retain the corresponding DLBL name within each file-id. Replace work.file with names suitable to your site.

syntax.file -- It is recommended that you retain the corresponding DLBL name within each file-id. Replace syntax.file with names suitable to your site.

anyname -- Any suitable name for your SYSIDMS file. Note that the name chosen must be identical to the FILEOUT value in the DITTO copy step.

Export Syntax Files--z/VSE

The JCL in TOOLJCL library member USMEXPRT.S (z/VSE) allocates the work files, allocates the syntax files, extracts information from the source dictionary and puts it onto disk, and produces syntax files on disk.

* ------ Dictionary Migrator STEP ------ 'USMEXPRT' SAMPLE JCL ------
* * **** CREATE A SYSIDMS PARAMETER FILE *****
// UPSI 1
// OPTION LOG,PARDUMP
// DLBL anyname,'work.file.SYSIDMS',0,SD
// EXTENT SYS041,volser,,,948,1
// ASSGN SYS041,DISK,VOL=volser,SHR
// EXEC DITTO
$$DITTO CSQ FILEOUT=anyname
* SYSIDMS parameters.
* For Local Mode specify:
* DMCL=dmcl-name,LOCAL=ON,JOURNAL=OFF
FILENAME=SYSIPT,FILETYPE=D,BLKSIZE=80,DEVADDR=SYSIPT
* For CV runs specify:
* DMCL=dmcl-name,LOCAL=OFF,JOURNAL=OFF
FILENAME=SYSIPT,FILETYPE=D,BLKSIZE=80,DEVADDR=SYSIPT

// OPTION PARTDUMP
* **** z/VSE library where CA IDMS Dictionary Migrator resides
// DLBL DBMS,'your.corelib'
// EXTENT ,volser
* **** z/VSE library where CA IDMS/DB executable phases reside
// DLBL IDMS,'idms.corelib'
// EXTENT ,volser
* // LIBDEF PHASE,SEARCH=(dbms.sublib,idms.sublib)

* ******** PARAMETER INPUT FOR DICT. MIGRATOR AND IDMS UTILITIES ******
// ASSGN SYS011,SYSRDR SYNTAX FILE
// ASSGN SYS010,SYSCH HOLD SYSCH ASSGN
// DLBL IJSYSPCH,'work.file.sysipt',0,SD IJSYSPCH IS USED TO
// EXTENT SYSCH,volser,,00100,050 CREATE VTOC ENTRY
// ASSGN SYSCH,DISK,VOL=volser,SHR FOR SYSIPT ASSGN
// CLOSE SYSCH,SYS010
// DLBL IJSYSIN,'work.file.sysipt',0,SD SAME FILE-ID AND
// EXTENT SYSIPT,volser,,00100,050 TRACKS AS IJSYSPCH
// ASSGN SYSIPT,DISK,VOL=volser,SHR PERMANENT ASSIGNMENT
// DLBL SYSIPT,'work.file.sysipt',0,SD SAME FILE-ID AND
// EXTENT SYS012,volser,,00100,050 TRACKS AS IJSYSPCH
// ASSGN SYS012,DISK,VOL=volser,SHR

* **** OUTPUT FROM IDMS UTILITIES AND INPUT TO Dictionary Migrator ***
// ASSGN SYS013,SYSLST

* ************** Dictionary Migrator WORK FILES **********************
* ********** Export Extract FILE ON TAPE FOR INPUT TO IMPORT STAGE ******
// DLBL WORKFIL,'work.file.WORKFIL',0,SD
// EXTENT SYS034,DISK,VOL=volser,SHR
// DLBL WORKFL2,'work.file.WORKFL2',0,SD
// EXTENT SYS039,DISK,VOL=volser,SHR

* ******** Syntax Files from Dict Migrator for IDMS Utilities ********
// DLBL RHDCDEL,'syntax.file.RHDCDEL',0,SD MAP & PANEL DELETES
// EXTENT SYS019,volser,,00760,005
// ASSGN SYS019,DISK,VOL=volser,SHR
CA IDMS - 19.0

// DLBL SCHMDEL,'syntax.file.SCHMDEL',0,SD
// EXTENT SYS020,volser,,00765,005
// ASSGN SYS020,DISK,VOL=volser,SHR
// DLBL SUBSDEL,'syntax.file.SUBSDEL',0,SD
// EXTENT SYS021,volser,,00770,005
// ASSGN SYS021,DISK,VOL=volser,SHR
// DLBL DDDLDEL,'syntax.file.DDDLDEL',0,SD
// EXTENT SYS022,volser,,00775,010
// ASSGN SYS022,DISK,VOL=volser,SHR
// DLBL DDDLUPD,'syntax.file.DDDLUPD',0,SD
// EXTENT SYS023,volser,,00785,100
// ASSGN SYS023,DISK,VOL=volser,SHR
// DLBL BCFUPD,'syntax.file.BCFUPD',0,SD
// EXTENT SYS040,volser,,00950,003
// ASSGN SYS040,DISK,VOL=volser,SHR
// DLBL TABLLOD,'syntax.file.TABLLOD',0,SD
// EXTENT SYS024,volser,,00885,005
// ASSGN SYS024,DISK,VOL=volser,SHR
// DLBL DDDLLOD,'syntax.file.DDDLLOD',0,SD
// EXTENT SYS025,volser,,00890,005
// ASSGN SYS025,DISK,VOL=volser,SHR
// DLBL SCHMUPD,'syntax.file.SCHMUPD',0,SD
// EXTENT SYS026,volser,,00895,005
// ASSGN SYS026,DISK,VOL=volser,SHR
// DLBL SUBSUPD,'syntax.file.SUBSUPD',0,SD
// EXTENT SYS028,volser,,00905,005
// ASSGN SYS028,DISK,VOL=volser,SHR
// DLBL SUBSLOD,'syntax.file.SUBSLOD',0,SD
// EXTENT SYS029,volser,,00910,005
// ASSGN SYS029,DISK,VOL=volser,SHR
// DLBL RHDCUPD,'syntax.file.RHDCUPD',0,SD
// EXTENT SYS030,volser,,00915,005
// ASSGN SYS030,DISK,VOL=volser,SHR
// DLBL RHDCLOD,'syntax.file.RHDCLOD',0,SD
// EXTENT SYS031,volser,,00920,005
// ASSGN SYS031,DISK,VOL=volser,SHR
// DLBL ADSOBGN,'syntax.file.ADSOBGN',0,SD
// EXTENT SYS032,volser,,00930,005
// ASSGN SYS032,DISK,VOL=volser,SHR
// DLBL DDLPGM,'syntax.file.DDLPGM',0,SD
// EXTENT SYS033,volser,,00925,005
// ASSGN SYS033,DISK,VOL=volser,SHR
// DLBL ADSBTAT,'syntax.file.ADSBTAT',0,SD
// EXTENT SYS037,volser,,00942,003
// ASSGN SYS037,DISK,VOL=volser,SHR
// DLBL USERUPD,'syntax.file.USERUPD',0,SD
// EXTENT SYS038,volser,,00945,003
// ASSGN SYS038,DISK,VOL=volser,SHR

* *************** Dictionary Migrator SYNTAX *******************
PROCESS,
RUN=EXPORT,

*** IDENTIFY SOURCE DICTIONARY
* IDSOURCE=(USER-ID,PASSWORD),
  DICT=?????????,
* THE OBJECT DICTIONARY WILL NOT BE ACCESSED IN THIS STEP.
* VALUES USED FOR IDOBJECT AND OBJDICT WILL BE PASSED TO THE
  UPLOAD STEP VIA THE SYNTAX FILES CREATED IN THIS STEP.
* IDOBJECT=(USER-ID,PASSWORD),
  OBJDICT=????????.

10-Jan-2018  173/240
EXTRACT,.............
.
.
EXTRACT,.............
/*
/&
// JOB RESET SYSIPT
CLOSE SYSIPT,SYSRDR
/&

- **your.corelib** -- The file id of the z/VSE library into which CA IDMS Dictionary Migrator was installed.

- **idms.corelib** -- The file id of the z/VSE library which contains the CA IDMS utility executable phases.

- **volser** -- The volume serial number or generic assignment of the disk volume on which the library, as specified in the previous statement, resides.

- **.sublib** -- The name of the sublibrary of the z/VSE library specified by the previous filename.

- **work.file** -- It is recommended that you retain the corresponding DLBL name within each file-id. Replace *work.file* with names suitable to your site.

- **syntax.file** -- It is recommended that you retain the corresponding DLBL name within each file-id. Replace *syntax.file* with names suitable to your site.

- **anyname** -- Any suitable name for your SYSIDMS file. Note that the name chosen must be identical to the FILEOUT value in the DITTO copy step.

**Import the Extract Tape -- z/VSE**

The JCL in TOOLJCL library member USMIMPRT.S (z/VSE) imports the extract tape, compares extract to object dictionary, and reports on the comparison.

**Note:** Suggested space allocations for all work files are contained in the Work Files Table in the section "Operations."

```plaintext
*----- Dictionary Migrator STEP ----- 'USMIMPRT' SAMPLE JCL -----*
* **** CREATE A SYSIDMS PARAMETER FILE *****
// UPSI 1
// OPTION LOG, PARTDUMP
// DLBL anyname,'work.file.SYSIDMS',0,SD
// EXTENT SYS041,volser,,,948,1
// ASSGN SYS041,DISK,VOL=volser,SHR
// EXEC DITTO
$$DITTO CSQ FILEOUT=anyname
* SYSIDMS parameters.
* For Local Mode specify :
* DMCL=dmcl-name,LOCAL=ON,JOURNAL=OFF
FILENAME=SYSIPT,FILETYPE=D,BLKSIZE=80,DEVADDR=SYSIPT
* For CV runs specify :
```
DMCL=dmcl-name, LOCAL=OFF, JOURNAL=OFF
FILENAME=SYSIPT, FILETYPE=D, BLKSIZE=80, DEVADDR=SYSIPT

// OPTION PARTDUMP
* ***** z/VSE library where CA IDMS Dictionary Migrator resides
// DLBL DBMS,'your.corelib'
// EXTENT ,volser
* ***** z/VSE library where CA IDMS/DB executable phases reside
// DLBL IDMS,'your.corelib'
// EXTENT ,volser
* // LIBDEF PHASE, SEARCH=(dbms.sublib, idms.sublib)
* // PARAMETER INPUT FOR DICT. MIGRATOR AND IDMS UTILITIES
* // ASSGN SYS011, SYSRDR
// DLBL IJSYSPH,'work.file.sysipt',0,SD
// EXTENT IJSYSPH, volser,,00100,050
// ASSGN IJSYSPH, DISK, VOL=volser
// CLOSE IJSYSPH, SYS010
* // DLBL IJSYSIN,'work.file.sysipt',0,SD
// EXTENT IJSYSIN, volser,,00100,050
// ASSGN IJSYSIN, DISK, VOL=volser
* // DLBL SYSIPT,'work.file.sysipt',0,SD
// EXTENT SYSIPT, volser,,00100,050
// ASSGN SYSIPT, DISK, VOL=volser
* // ASSIGN SYSPCH, IGN
// DLBL SYSPCH,'work.file.SYSPCH',0,SD
// EXTENT SYSPCH, volser,,00150,200
// ASSGN SYSPCH, DISK, VOL=volser
* // ASSIGN SYS012, SYSLST
* // DLBL EXTRACT,'WORK.FILE.EXTRACT'
// EXTENT EXTRACT, volser,,00350,200
// ASSGN EXTRACT, DISK, VOL=volser
* // DLBL VSAMEXT, 'WORK.FILE.VSAMEXT',0,VSAM
// EXTENT VSAMEXT, volser
// ASSGN VSAMEXT, DISK, VOL=volser
* // DLBL SELECT, 'WORK.FILE.SELECT',0,SD
// EXTENT SELECT, volser,,00550,003
// ASSGN SELECT, DISK, VOL=volser
* // DLBL VALDRPT, 'WORK.FILE.VALDRPT',0,SD
// EXTENT VALDRPT, volser,,00553,007
// ASSGN VALDRPT, DISK, VOL=volser
* // DLBL WORKFIL, 'WORK.FILE.WORKFIL',0,SD
// EXTENT WORKFIL, volser,,00560,100
// ASSGN WORKFIL, DISK, VOL=volser
* // DLBL WORKFL2, 'WORK.FILE.WORKFL2',0,SD
// EXTENT WORKFL2, volser,,00660,100
// ASSGN WORKFL2, DISK, VOL=volser
* // ASSIGN SYS013, SYSLST
* // DLBL RHDCDEL,'syntax.file.RHDCDEL',0,SD
// EXTENT RHDCDEL, volser,,00760,005
// ASSGN RHDCDEL, DISK, VOL=volser
* // DLBL SCHMDEL, 'syntax.file.SCHMDEL',0,SD
// EXTENT SCHMDEL, volser,,00765,005
// ASSGN SCHMDEL, DISK, VOL=volser
your.corelib -- The file id of the z/VSE library into which CA IDMS Dictionary Migrator was installed.
idms.corelib -- The file id of the z/VSE library which contains the CA IDMS utility executable phases.

volser -- The volume serial number or generic assignment of the disk volume on which the library, as specified in the previous statement, resides.

.sublib -- The name of the sublibrary of the z/VSE library specified by the previous filename.

work.file -- It is recommended that you retain the corresponding DLBL name within each file-id. Replace work.file with names suitable to your site.

syntax.file -- It is recommended that you retain the corresponding DLBL name within each file-id. Replace syntax.file with names suitable to your site.

anyname -- Any suitable name for your SYSIDMS file. Note that the name chosen must be identical to the FILEOUT value in the DITTO copy step.

Upload Syntax Files -- z/VSE

The JCL in TOOLJCL library member USMLOAD1.S (z/VSE) is used to upload syntax files to the object dictionary using the CA IDMS utilities, or using the CA user password upload utility supplied by CA. There can be from 1 to 16 steps involved in the upload, depending on your site and type of migration.

For more information on uploading syntax files, see the section "Operations."

Note that if you want to delete entities from the source dictionary after migration, make a second non-CHANGEONLY run using the same parameters as the real migration, specify the same dictionary for both source and object.

* --- USE IDMS TO LOAD OBJECT DICT ----- 'USMLOAD1' SAMPLE JCL ---
* // JOB RHDCDEL
* *-------------------------------------------------------------------*
* RHDCDEL - DELETE PANELS AND MAPS
* *-------------------------------------------------------------------*
* **** CREATE A SYSIDMS PARAMETER FILE *****
* **** $DITTO (S) FILEOUT=anyname
* SYSIDMS parameters.
* For Local Mode specify :
* DMCL=dmcl-name,LOCAL=ON,JOURNAL=OFF
FILENAME=SYSIPT,FILETYPE=D,BLKSIZE=80,DEVADDR=SYSIPT
* For CV runs specify :
* DMCL=dmcl-name,LOCAL=OFF,JOURNAL=OFF
FILENAME=SYSIPT,FILETYPE=D,BLKSIZE=80,DEVADDR=SYSIPT
* // ups 00000000
* **** z/VSE library where CA IDMS Dictionary Migrator resided
// DLBL DBMS,'your.corelib'
* // EXTENT ,volser
* **** z/VSE library where CA IDMS/DB executable phases reside
// DLBL IDMS,'idms.corelib'
// EXTENT ,volser
CA IDMS - 19.0

* // LIBDEF PHASE,SEARCH=(dbms.sublib,idms.sublib)
* // DLBL IJSYSIN,'syntax.file.RHDCDEL',0,SD
// EXTENT SYSIPT,volser
  ASSGN SYSIPT,DISK,VOL=volser,SHR
// DLBL SYSIDMS,'work.file.SYSIDMS'
// EXTENT SYS041,volser
// ASSGN SYS041,DISK,VOL=volser,SHR
// EXEC RHDCMAP1
*
/&
// JOB RESET SYSIPT
  CLOSE SYSIPT,SYSRDR
/&
// JOB SCHMDEL
*-------------------------------------------------------------------*
*SCHMDEL - DELETE SCHEMA, AND SUBSCHEMA SOURCE
*-------------------------------------------------------------------*
// upsi 00000000
* **** z/VSE library where CA IDMS Dictionary Migrator resided
// DLBL DBMS,'your.corelib'
// EXTENT ,volser
*
* **** z/VSE library where CA IDMS/DB executable phases reside
// DLBL IDMS,'idms.corelib'
// EXTENT ,volser
*
// LIBDEF PHASE,SEARCH=(dbms.sublib,idms.sublib)
*
// DLBL IJSYSIN,'syntax.file.SCHMDEL',0,SD
// EXTENT SYSIPT,volser
  ASSGN SYSIPT,DISK,VOL=volser,SHR
// DLBL SYSIDMS,'work.file.SYSIDMS'
// EXTENT SYS041,volser
// ASSGN SYS041,DISK,VOL=volser,SHR
// EXEC IDMSCHEM
*
/&
// JOB RESET SYSIPT
  CLOSE SYSIPT,SYSRDR
/&
// JOB SUBSDEL
*-------------------------------------------------------------------*
*SUBSDEL - DELETE SUBSCHEMA
*-------------------------------------------------------------------*
// upsi 00000000
* **** z/VSE library where CA IDMS Dictionary Migrator resided
// DLBL DBMS,'your.corelib'
// EXTENT ,volser
*
* **** z/VSE library where CA IDMS/DB executable phases reside
// DLBL IDMS,'idms.corelib'
// EXTENT ,volser
*
// LIBDEF PHASE,SEARCH=(dbms.sublib,idms.sublib)
*
// DLBL IJSYSIN,'syntax.file.SUBSDEL',0,SD
// EXTENT SYSIPT,volser
  ASSGN SYSIPT,DISK,VOL=volser,SHR
// DLBL SYSIDMS,'work.file.SYSIDMS'
// EXTENT SYS041,volser
// ASSGN SYS041,DISK,VOL=volser,SHR
// EXEC IDMSUBSC
*
/&
// JOB RESET SYSIPT
  CLOSE SYSIPT,SYSRDR
/&
// JOB DDDLDEL
*-------------------------------------------------------------------*
* DDDLDEL - DELETE OTHER ENTITIES                             *
*-------------------------------------------------------------------*
// upsi 00000000
**** z/VSE library where CA IDMS Dictionary Migrator resided
// DLBL  DBMS,'your.corelib'
// EXTENT ,volser
*
**** z/VSE library where CA IDMS/DB executable phases reside
// DLBL  IDMS,'idms.corelib'
// EXTENT ,volser
* // LIBDEF PHASE, SEARCH=(dbms.sublib, idms.sublib) *
* // DLBL  IJSYSIN,'syntax.file.DDDLDEL',0,SD *
* // EXTENT SYSIPT, volser *
* // ASSGN SYSIPT, DISK, VOL=volser, SHR *
// DLBL  SYSIDMS,'work.file.SYSIDMS'
// EXTENT SYS041, volser
// ASSGN SYS041, DISK, VOL=volser, SHR
* // EXEC IDMSDDDL *
* /
* // JOB RESET SYSIPT
CLOSE SYSIPT, SYSRDR
* /
* // JOB DDDLUPD
*-------------------------------------------------------------------*
* DDDLUPD - ADD/MODIFY CLASS, ATTRIBUTE, SYSTEM, RECORD, ELEMENT, *
* MESSAGE, MODULE, AND TABLE ENTITIES                             *
*-------------------------------------------------------------------*
// upsi 00000000
**** z/VSE library where CA IDMS Dictionary Migrator resided
// DLBL  DBMS,'your.corelib'
// EXTENT ,volser
*
**** z/VSE library where CA IDMS/DB executable phases reside
// DLBL  IDMS,'idms.corelib'
// EXTENT ,volser
* // LIBDEF PHASE, SEARCH=(dbms.sublib, idms.sublib) *
* // DLBL  IJSYSIN,'syntax.file.DDDLUPD',0,SD *
* // EXTENT SYSIPT, volser *
* // ASSGN SYSIPT, DISK, VOL=volser, SHR *
// DLBL  SYSIDMS,'work.file.SYSIDMS'
// EXTENT SYS041, volser
// ASSGN SYS041, DISK, VOL=volser, SHR
* // EXEC IDMSDDDL *
* /
* // JOB RESET SYSIPT
CLOSE SYSIPT, SYSRDR
* /
* // JOB TABLLOD
*-------------------------------------------------------------------*
* TABLLOD - GENERATE TABLE LOAD MODULES                             *
*-------------------------------------------------------------------*
// upsi 00000000
**** z/VSE library where CA IDMS Dictionary Migrator resided
// DLBL  DBMS,'your.corelib'
// EXTENT ,volser
*
**** z/VSE library where CA IDMS/DB executable phases reside
// DLBL  IDMS,'idms.corelib'
// EXTENT ,volser
* // LIBDEF PHASE, SEARCH=(dbms.sublib, idms.sublib)
// DLBL  IJSYSIN,'syntax.file.TABLLOD',0,SD
// EXTENT SYSIPT,volser
ASSGN SYSIPT,DISK,VOL=volser,SHR
// DLBL  SYSIDMS,'work.file.SYSIDMS'
// EXTENT SYS041,volser
ASSGN SYS041,DISK,VOL=volser,SHR
// EXEC IDMSDDDL
*
&
// JOB RESET SYSIPT
CLOSE SYSIPT,SYSRDR
&
// JOB DDDLLOD
*-----------------------------------------------*
*DDDLLOD - ADD/MODIFY LOAD MODULES FOR SUBSCHEMA, MAP, MAP EDIT *
* TABLE, AND DIALOG *
*-----------------------------------------------*
// upsi 00000000
* ***** z/VSE library where CA IDMS Dictionary Migrator resided
// DLBL  DBMS,'your.corelib'
// EXTENT ,volser
*
* ***** z/VSE library where CA IDMS/DB executable phases reside
// DLBL  IDMS,'idms.corelib'
// EXTENT ,volser
*
// LIBDEF PHASE,SEARCH=(dbms.sublib,idms.sublib)
* 
// DLBL  IJSYSIN,'syntax.file.DDDLLOD',0,SD
// EXTENT SYSIPT,volser
ASSGN SYSIPT,DISK,VOL=volser,SHR
// DLBL  SYSIDMS,'work.file.SYSIDMS'
// EXTENT SYS041,volser
ASSGN SYS041,DISK,VOL=volser,SHR
// EXEC IDMSDDDL
&
// JOB RESET SYSIPT
CLOSE SYSIPT,SYSRDR
&
// JOB SCHMUPD
*-----------------------------------------------*
*SCHMUPD - ADD SCHEMA SOURCE *
*-----------------------------------------------*
// upsi 00000000
* ***** z/VSE library where CA IDMS Dictionary Migrator resided
// DLBL  DBMS,'your.corelib'
// EXTENT ,volser
*
* ***** z/VSE library where CA IDMS/DB executable phases reside
// DLBL  IDMS,'idms.corelib'
// EXTENT ,volser
*
// LIBDEF PHASE,SEARCH=(dbms.sublib,idms.sublib)
* 
// DLBL  IJSYSIN,'syntax.file.SCHMUPD',0,SD
// EXTENT SYSIPT,volser
ASSGN SYSIPT,DISK,VOL=volser,SHR
// DLBL  SYSIDMS,'work.file.SYSIDMS'
// EXTENT SYS041,volser
ASSGN SYS041,DISK,VOL=volser,SHR
// EXEC IDMSCHEM
*
&
// JOB RESET SYSIPT
CLOSE SYSIPT,SYSRDR
&
// JOB SUBSUPD
*-----------------------------------------------*
*SUBSUPD - ADD SUBSCHEMA SOURCE *
*-----------------------------------------------*
// upsi 00000000
* **** z/VSE library where CA IDMS Dictionary Migrator resided
  // DLBL DBMS,'your.corelib'
  // EXTENT ,volser
*  
* **** z/VSE library where CA IDMS/DB executable phases reside
  // DLBL IDMS,'idms.corelib'
  // EXTENT ,volser
*  
  // LIBDEF PHASE,SEARCH=(dbms.sublib,idms.sublib)
*  
  // DLBL IJSYSIN,'syntax.file.SUBSUPD',0,SD
  // EXTENT SYSIPT,volser
  ASSGN SYSIPT,DISK,VOL=volser,SHR
  // DLBL SYSIDMS,'work.file.SYSIDMS'
  // EXTENT SYS041,volser
  ASSGN SYS041,DISK,VOL=volser,SHR
  // EXEC IDMSUBSC
*  
  */&
  // JOB RESET SYSIPT
  CLOSE SYSIPT,SYSRDR
  */&
  // JOB SUBSLOD
  *-------------------------------------------------------------------*
  *SUBSLOD - GENERATE SUBSCHEMA LOAD MODULE
  *-------------------------------------------------------------------*
  *  
  *  
  // DLBL IJSYSIN,'syntax.file.SUBSLOD',0,SD
  // EXTENT SYSIPT,volser
  ASSGN SYSIPT,DISK,VOL=volser,SHR
  // DLBL SYSIDMS,'work.file.SYSIDMS'
  // EXTENT SYS041,volser
  ASSGN SYS041,DISK,VOL=volser,SHR
  // EXEC IDMSUBSC
  */&
  // JOB RESET SYSIPT
  CLOSE SYSIPT,SYSRDR
  */&
  // JOB RHDCUPD
  *-------------------------------------------------------------------*
  *RHDCUPD - ADD/MODIFY PANEL AND MAP SYNTAX
  *-------------------------------------------------------------------*
  *  
  *  
  // DLBL IJSYSIN,'syntax.file.RHDCUPD',0,SD
  // EXTENT SYSIPT,volser
  ASSGN SYSIPT,DISK,VOL=volser,SHR
  // DLBL SYSIDMS,'work.file.SYSIDMS'
  // EXTENT SYS041,volser
  ASSGN SYS041,DISK,VOL=volser,SHR
  // EXEC RHDCMAP1
* /&
* JOB RESET SYSIPT
  CLOSE SYSIPT,SYSRDR
/&
* JOB RHDCLOD
*-------------------------------------------------------------------*
*RHDCLOD - COMPILE MAP
*-------------------------------------------------------------------*
// upsi 00000000
* ****** z/VSE library where CA IDMS Dictionary Migrator resided
  // DLBL  DBMS,'your.corelib'
  // EXTENT ,volser
* ****** z/VSE library where CA IDMS/DB executable phases reside
  // DLBL  IDMS,'idms.corelib'
  // EXTENT ,volser
* // LIBDEF PHASE,SEARCH=(dbms.sublib,idms.sublib)
* // DLBL  IJSYSIN,'syntax.file.RHDCLOD',0,SD
  // EXTENT SYSIPT,volser
    ASSGN SYSIPT,DISK,VOL=volser,SHR
  // DLBL  SYSIDMS,'work.file.SYSIDMS'
    // EXTENT SYSO41,volser
    ASSGN SYSO41,DISK,VOL=volser,SHR
    // EXEC RHDCMPUT
* /&
* JOB RESET SYSIPT
  CLOSE SYSIPT,SYSRDR
/&
* JOB ADSOBGN
*-------------------------------------------------------------------*
*ADSOBGN - GENERATE ADS DIALOG
*-------------------------------------------------------------------*
// upsi 00000000
* ****** z/VSE library where CA IDMS Dictionary Migrator resided
  // DLBL  DBMS,'your.corelib'
  // EXTENT ,volser
* ****** z/VSE library where CA IDMS/DB executable phases reside
  // DLBL  IDMS,'idms.corelib'
  // EXTENT ,volser
* // LIBDEF PHASE,SEARCH=(dbms.sublib,idms.sublib)
* // DLBL  IJSYSIN,'syntax.file.ADSOBGN',0,SD
  // EXTENT SYSIPT,volser
    ASSGN SYSIPT,DISK,VOL=volser,SHR
  // DLBL  SYSIDMS,'work.file.SYSIDMS'
    // EXTENT SYSO41,volser
    ASSGN SYSO41,DISK,VOL=volser,SHR
    // EXEC ADSOBCOM
* /&
* JOB RESET SYSIPT
  CLOSE SYSIPT,SYSRDR
/&
* JOB DDDLPGM
*-------------------------------------------------------------------*
*DDDLPGM - ADD/MODIFY PROGRAM SYNTAX
*-------------------------------------------------------------------*
// upsi 00000000
* ****** z/VSE library where CA IDMS Dictionary Migrator resided
  // DLBL  DBMS,'your.corelib'
  // EXTENT ,volser
* ****** z/VSE library where CA IDMS/DB executable phases reside
CA IDMS - 19.0

// DLBL IDMS,'idms.corelib'
// EXTENT ,volser
* 
// LIBDEF PHASE,SEARCH=(dbms.sublib,idms.sublib)
* 
// DLBL IJSYSIN,'syntax.file.DDLPGM',0,SD
// EXTENT SYSIPT,volser
    // ASSGN SYSIPT,DISK,_VOL=volser,SHR
// DLBL SYSIDMS,'work.file.SYSIDMS'
// EXTENT SYS041,volser
    // ASSGN SYS041,DISK,_VOL=volser,SHR
// EXEC IDMSDDDL
* 
// JOB RESET SYSIPT
CLOSE SYSIPT,SYSRDR
&
*-------------------------------------------------------------------*
*ADSBTAT - UPDATE TASK APPLICATION TABLE.                        *
*             IF THERE IS APPLICATION WHICH USES THE REVISED ADSA, *
*             EXECUTE THIS STEP.                                  *
*-------------------------------------------------------------------*
// upsi 00000000
* ***** z/VSE library where CA IDMS Dictionary Migrator resided
// DLBL DBMS,'your.corelib'
// EXTENT ,volser
* 
* ***** z/VSE library where CA IDMS/DB executable phases reside
// DLBL IDMS,'idms.corelib'
// EXTENT ,volser
* 
// LIBDEF PHASE,SEARCH=(dbms.sublib,idms.sublib)
* 
// DLBL IJSYSIN,'syntax.file.ADSBTAT',0,SD
// EXTENT SYSIPT,volser
    // ASSGN SYSIPT,DISK,_VOL=volser,SHR
// DLBL SYSIDMS,'work.file.SYSIDMS'
// EXTENT SYS041,volser
    // ASSGN SYS041,DISK,_VOL=volser,SHR
// EXEC ADSOBATAT
* 
*-------------------------------------------------------------------*
* BCFUPD - ADD SQL ENTITY DEFINITIONS                              *
*                                                             *
*-------------------------------------------------------------------*
// upsi 00000000
* ***** z/VSE library where CA IDMS/DB executable phases reside
// DLBL IDMS 'idms.corelib'
// EXTENT ,volser
* 
// LIBDEF PHASE,SEARCH=(idms.sublib)
* 
// DLBL IJSYSIN,'syntax.file.BCFUPD',0,SD
// EXTENT SYSIPT,volser
    // ASSGN SYSIPT,DISK,_VOL=volser,SHR
// DLBL SYSIDMS,'work.file.SYSIDMS'
// EXTENT SYS041,volser
    // ASSGN SYS041,DISK,_VOL=volser,SHR
// EXEC IDMSBCF
&
// JOB RESET SYSIPT
CLOSE SYSIPT,SYSRDR
&
// JOB USMULOD
*-------------------------------------------------------------------*
*USMULOD - MODIFY MIGRATED USER-PASSWORDS. DO NOT RUN THIS STEP     *
*             IS DISCONNECT=USER WAS SPECIFIED.                    *
*-------------------------------------------------------------------*
// upsi 00000000
* ***** z/VSE library where CA IDMS Dictionary Migrator resided
CA IDMS - 19.0

// DLBL   DBMS,'your.corelib'
// EXTENT ,volser

* * * z/VSE library where CA IDMS/DB executable phases reside
// DLBL   IDMS,'idms.corelib'
// EXTENT ,volser

* LIBDEF PHASE,SEARCH=(dbms.sublib,idms.sublib)
* ASSGN SYS013,SYSLST
*
// DLBL   USERUPD,'syntax.file.USERUPD',0,SD
// EXTENT SYS038,volser
ASSGN SYS038,DISK,VOL=volser,SHR
// DLBL   SYSIDMS,'work.file.SYSIDMS'
// EXTENT SYS041,volser
ASSGN SYS041,DISK,VOL=volser,SHR
// EXEC  USMULOD,SIZE=USMULOD
*
/&

- **your.corelib** -- The file id of the z/VSE library into which CA IDMS Dictionary Migrator was installed.

- **idms.corelib** -- The file id of the z/VSE library which contains the CA IDMS utility executable phases.

- **volser** -- The volume serial number or generic assignment of the disk volume on which the library, as specified in the previous statement, resides.

- **upsi** -- Replace 'UPSI 00000000' with the central version UPSI setting appropriate to your site. If you wish to run in local mode, add the JCL appropriate to local mode processing.

- **.sublib** -- The name of the sublibrary of the z/VSE library specified by the previous filename.

- **vsam.file** -- It is recommended that you retain the DLBL name VSAMEXT within the VSAM file name. Replace vsam.file with a name suitable to your site.

- **syntax.file** -- The syntax files were created in a previous step or job. Use the file-id's for these files that were used when the files were created.

- **anyname** -- Any suitable name for your SYSIDMS file. Note that the name chosen must be identical to the FILEOUT value in the DITTO copy step.

- **nnnn** -- The beginning relative track number for SYSIDMS file (1 track required).

**Delete Migrated Source from Source Dictionary -- z/VSE**

The JCL in TOOLJCL library member USMLOAD3.S (z/VSE) is an optional job that, when executed, deletes the migrated source from the source dictionary after migration.

**Note:** If you want to delete entities from the source dictionary after migration, make a second non-CHANGEONLY run using the same parameters as the real migration, specify the same dictionary for both source and object.
* --- DELETE ENTITIES FROM SOURCE DICT - 'USMLOAD3' SAMPLE JCL ---

// JOB RHDCDEL

* RHDCDEL - DELETE PANELS AND MAPS

* **** CREATE A SYSIDMS PARAMETER FILE *****

// UPSI 1
// OPTION LOG,PARTDUMP
// DLBL anyname,'work.file.SYSIDMS',0,SD
// EXTENT SYS041,volser,,,nnnn,1
// ASSGN SYS041,DISK,VOL=volser,SHR
// EXEC DITTO

\$DITTO CSQ FILEOUT=anyname

** SYSIDMS parameters.
* For Local Mode specify :

DMCL=dmcl-name,LOCAL=ON,JOURNAL=OFF
FILENAME=SYSIPT,FILETYPE=D,BLKSIZE=80,DEVADDR=SYSIPT
*
* For CV runs specify :

DMCL=dmcl-name,LOCAL=OFF,JOURNAL=OFF
FILENAME=SYSIPT,FILETYPE=D,BLKSIZE=80,DEVADDR=SYSIPT

/*

// upsi 00000000
* **** z/VSE library where CA IDMS Dictionary Migrator resided
// DLBL DBMS,'your.corelib'
// EXTENT ,volser
*
* **** z/VSE library where CA IDMS/DB executable phases reside
// DLBL IDMS,'idms.corelib'
// EXTENT ,volser
*
// LIBDEF PHASE,SEARCH=(dbms.sublib,idms.sublib)
*
// DLBL IJSYSIN,'syntax.file.RHDCDEL',0,SD
// EXTENT SYSIPT,volser
// ASSGN SYSIPT,DISK,VOL=volser,SHR
// DLBL SYSIDMS,'work.file.SYSIDMS'
// EXTENT SYS041,volser
// ASSGN SYS041,DISK,VOL=volser,SHR
// EXEC RHDCMAP1
*
/&

// JOB RESET SYSIPT
CLOSE SYSIPT,SYSRDR
/&

// JOB SCHMDEL

* SCHMDEL - DELETE SCHEMA, DMCL, AND SUBSCHEMA SOURCE
*
* **** z/VSE library where CA IDMS Dictionary Migrator resided
// DLBL DBMS,'your.corelib'
// EXTENT ,volser
*
* **** z/VSE library where CA IDMS/DB executable phases reside
// DLBL IDMS,'idms.corelib'
// EXTENT ,volser
*
// LIBDEF PHASE,SEARCH=(dbms.sublib,idms.sublib)
*
// DLBL IJSYSIN,'syntax.file.SCHMDEL',0,SD
// EXTENT SYSIPT,volser
// ASSGN SYSIPT,DISK,VOL=volser,SHR
// DLBL SYSIDMS,'work.file.SYSIDMS'
// EXTENT SYS041,volser
// ASSGN SYS041,DISK,VOL=volser,SHR
// EXEC IDMSCHEM
CA IDMS - 19.0

/* JOB RESET SYSIPT
   CLOSE SYSIPT,SYSRDR */
/*
   JOB SUBSDEL
   *-----------------------------------------------*
   *SUBSDEL - DELETE SUBSCHEMA
   *N O T E: THIS IS REQUIRED ONLY IF A SUBSCHEMA WAS MIGRATED
   * WITHOUT ITS SCHEMA
   *-----------------------------------------------*
   * upsi 00000000
   * **** z/VSE library where CA IDMS Dictionary Migrator resided
   * DLBL DBMS,'your.corelib'
   * EXTENT ,volser
   * **** z/VSE library where CA IDMS/DB executable phases reside
   * DLBL IDMS,'idms.corelib'
   * EXTENT ,volser
   * LIBDEF PHASE,SEARCH=(dbms.sublib,idms.sublib)
   * DBL IJSYSIN,'syntax.file.SUBSDEL',0,SD
   * EXTENT SYSIPT,volser
   * ASSGN SYSIPT,DISK,VOL=volser,SHR
   * DBL SYSIDSMS,'work.file.SYSIDSMS'
   * EXTENT SYS041,volser
   * ASSGN SYS041,DISK,VOL=volser,SHR
   * EXEC IDMSSUBSC */
/*
   JOB DDDEL
   *-----------------------------------------------*
   *DDDDLDEL - DELETE OTHER ENTITIES
   *-----------------------------------------------*
   * upsi 00000000
   * **** z/VSE library where CA IDMS Dictionary Migrator resided
   * DLBL DBMS,'your.corelib'
   * EXTENT ,volser
   * **** z/VSE library where CA IDMS/DB executable phases reside
   * DLBL IDMS,'idms.corelib'
   * EXTENT ,volser
   * LIBDEF PHASE,SEARCH=(dbms.sublib,idms.sublib)
   * DBL IJSYSIN,'syntax.file.DDDLDEL',0,SD
   * EXTENT SYSIPT,volser
   * ASSGN SYSIPT,DISK,VOL=volser,SHR
   * DBL SYSIDSMS,'work.file.SYSIDSMS'
   * EXTENT SYS041,volser
   * ASSGN SYS041,DISK,VOL=volser,SHR
   * EXEC IDMSDDDL */
/*
   JOB RESET SYSIPT
   CLOSE SYSIPT,SYSRDR */

- **your.corelib** -- The file id of the z/VSE library into which CA IDMS Dictionary Migrator was installed.
- **idms.corelib** -- The file id of the z/VSE library which contains the CA IDMS utility executable phases.
- **volser** -- The volume serial number or generic assignment of the disk volume on which the library, as specified in the previous statement, resides.

- **upsi** -- Replace 'UPSI 00000000' with the central version UPSI setting appropriate to your site. If you wish to run in local mode, add the JCL appropriate to local mode processing.

- **.sublib** -- The name of the sublibrary of the z/VSE library specified by the previous filename.

- **syntax.file** -- The syntax files were created in a previous step or job. Use the file-id's for these files that were used when the files were created.

- **anyname** -- Any suitable name for your SYSIDMS file.

  **Note:** The name chosen must be identical to the FILEOUT value in the DITTO copy step.

- **nnnn** -- The beginning relative track number for SYSIDMS file (1 track required).

**Allocate the VSAM Work File--z/VM**

The USMVSAM EXEC allocates the VSAM work file.

  **Note:** For more information, see the section "Operations."

```c
/* */
TRACE OFF; SIGNAL ON ERROR
/* */
DELETE -
  (work.file.vsamext) -
CLUSTER
DEFINE CLUSTER (-
  NAME(work.file.vsamext) -
  TRACKS(prialloc seclalloc) -
  INDEXED -
  VOLUMES(volser) -
  RECZ(228 228) -
  KEYS(84 2) -
  usetype -
)
```

- **work.file.vsamext** -- The data set name of the VSAM KSDS work file. It is recommended that the data set name include the work file name VSAMEXT.

- **volser** -- The volume serial of the disk that will contain the VSAM KSDS work file.

- **usetype** -- The appropriate usage type: REUSE or UNIQUE. If you specify UNIQUE, you must delete and define this cluster prior to each execution of CA IDMS Dictionary Migrator>
- **primary and secondary space allocations** for the VSAM KSDS work file. A track allocation of 60 primary tracks and 30 secondary tracks (60 30) on a 3350 disk device should be adequate for this file. You may need to change these allocations for your DASD type and migration needs.

**Extract Information from the Source Dictionary -- z/VM**

The USMXTRCT EXEC allocates the work files, allocates the syntax files, extracts information from the source dictionary, compares the extract to the object dictionary, reports on the comparison, and produces the syntax files.

```plaintext
/* */
TRACE OFF; SIGNAL ON ERROR
/* */
/* USMXTRCT */
/* */
CA_LOADLIB_FN = 'yourlib'
IDMS_LOADLIB_FN = 'idmslib'
SORT_TXTLIB_FN = 'sortlib'
INPUT_PARM_FN = 'USMXTRCT'
INPUT_PARM_FT = 'MIGPARM'
INPUT_PARM_FM = 'A'
/* */
/* Link and access the Minidisks containing the required librarie(s) */
/* */
'CP SPOOL PRINTER NOCONT CLOSE'
'CP SPOOL PRINTER TO * NOHOLD CONT FORM OFF DIST OFF'
'GLOBAL LOADLIB ' CA_LOADLIB_FN IDMS_LOADLIB_FN
'GLOBAL TXTLIB ' SORT_TXTLIB_FN
/* */
/* Create the input parameter file. */
/* */
CALL CREATE_INPUT_PARM_FILE
/* */
/* Product specific files. */
/* */
'FILEDEF SYSPRT PRINTER'
'FILEDEF SYSLST PRINTER'
'FILEDEF SYSOUT PRINTER'
'FILEDEF AUDIT PRINTER'
'FILEDEF SELECT DISK USMXTRCT SELECT a'
'FILEDEF SYSIPT DISK USMXTRCT MIGIN a'
'FILEDEF VALDRPT DISK USMXTRCT VALDRPT a4'
'FILEDEF WORKFL2 DISK USMXTRCT WORKFL2 a4'
'FILE EXTRACT DISK USMXFILE EXTRACT a4 (RECFM FB LRECL 228 BLOCK 228'
'FILEDEF RHDCDEL DISK USMXFILE RHDCDEL a'
'FILEDEF SCHMDEL DISK USMXFILE SCHMDEL a'
'FILEDEF SUBSDEL DISK USMXFILE SUBSDEL a'
'FILEDEF DDDLDEL DISK USMXFILE DDDLDEL a'
'FILEDEF DDDLPUD DISK USMXFILE DDDLPUD a'
'FILEDEF BCFUPD DISK USMXFILE BCFUPD a'
'FILEDEF TABLLOD DISK USMXFILE TABLLOD a'
'FILEDEF DDDLLOD DISK USMXFILE DDDLLOD a'
'FILEDEF SCHMPUD DISK USMXFILE SCHMPUD a'
'FILEDEF SUBSUPD DISK USMXFILE SUBSUPD a'
'FILEDEF SUBSLOD DISK USMXFILE SUBSLOD a'
'FILEDEF RHDCUPD DISK USMXFILE RHDCUPD a'
'FILEDEF RHDCLOD DISK USMXFILE RHDCLOD a'
'FILEDEF ADSOBN DISK USMXFILE ADSOBN a'
'FILEDEF DDDLPGM DISK USMXFILE DDDLPGM a'
'FILEDEF ADSBTAT DISK USMXFILE ADSBTAT a'
'FILEDEF USERUPD DISK USMXFILE USERUPD a'
'FILEDEF MIGPARM DISK: INPUT_PARM_FN INPUT_PARM_FT INPUT_PARM_FM
/* */
'DLBL IJSYSCT vsam fn DSN vsamcat'
'DLBL IJSYSUC vsam fn DSN vsamcat'
```
'DLBL VSAMEXT vsam_fm DSN vsamext (VSAM)
/*
 /* You must create a file 'MIGR SYSIDMS a' containing the SYSIDMS
 /* parameters you use to specify your runtime environment.
 /* If you wish to run in Local Mode specify the following values:
 /* LOCAL=ON,JOURNAL=OFF,DMCL=dmcl-name.
 /* If you wish to run CV an 80 byte blank record will suffice.
 /*
 /* FILEDEF SYSIDMS DISK MIGR SYSIDMS a'
 /* Insert FILEDEF statements for SORT work space as required by
 /* your SORT product.
 /* SIGNAL OFF ERROR
 SAY 'STARTING CA IDMS/Dictionary Migrator RUN'
 'EXECOS OSRUN USMGRTR'
 USMXTRCT_RC = RC
 IF USMXTRCT_RC > 4
 THEN DO
 CALL ERROR
 END
 'CP SPOOL PRINTER NOCONT'
 'CP CLOSE PRINTER NAME USMXTRCT LISTING'
 'CP SPOOL PRINTER OFF'
 SAY 'USMXTRCT FINISHED WITH A RETURN CODE OF' USMXTRCT_RC
 'GLOBAL LOADLIB'
 'GLOBAL TXTLIB'
 'FILEDEF * CLEAR'
 EXIT USMXTRCT_RC
/*++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
CREATE_INPUT_PARM_FILE:
/*++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
SIGNAL OFF ERROR
'ERASE' INPUT_PARM_FN INPUT_PARM_FT INPUT_PARM_FM
SIGNAL ON ERROR
/*
PUSH 'FFILE'
PUSH 'EXTRACT, user extract statements'
PUSH 'DICT=srcdict, IDOBJECT=(userid, passwd), OBJDICT=objdict'
PUSH 'PROCESS, RUN=runtype,'
PUSH 'INPUT'
PUSH 'SET LRECL 80'
PUSH 'SET RECFM F'
'XEDIT' INPUT_PARM_FN INPUT_PARM_FT INPUT_PARM_FM,
'(NOPROFILE NOSCREEN NOMSG'
RETURN
/*
/*++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
ERROR:
/*++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
ERROR_RC = RC
TRACE OFF; SIGNAL OFF ERROR
SAY 'NON-ZERO RETURN CODE ENCOUNTERED IN EXEC AT LINE' SIGL
'CP SPOOL PRINTER NOCONT'
'CP CLOSE PRINTER NAME USMXTRCT LISTING'
'CP SPOOL PRINTER OFF'
'GLOBAL LOADLIB'
'GLOBAL TXTLIB'
'FILEDEF * CLEAR'
EXIT ERROR_RC
/*

• yourlib -- The file name of the load library into which you downloaded CA IDMS Dictionary
  Migrator.

• idmslib -- The file name of the load library containing your CA IDMS SUBSCHEMA and DMCL
  modules.

• sortlib -- The file name of the text library containing your sort modules.
The file mode for the relevant file.

- **vsam_fm** -- The file mode of the accessed minidisk which contains your VSAM catalog(s) and file(s).

- **vsamcat** -- The data set name of your VSAM catalog.

- **vsamext** -- The data set name of the PAGUTIL VSAM file.

**Export Syntax Files--z/VM**

The USMEXPRT EXEC allocates the work files, allocates the syntax files, extracts information from the source dictionary and puts it onto disk, and produces syntax files on disk.

```plaintext
/* */
TRACE OFF; SIGNAL ON ERROR
/* */
/* USMEXPRT */
CA_LOADLIB_FN = 'yourlib'
IDMS_LOADLIB_FN = 'idmslib'
SORT_TXTLIB_FN = 'sortlib'
INPUT_PARM_FN = 'USMEXPRT'
INPUT_PARM_FT = 'MIGPARM'
INPUT_PARM_FM = 'A'
/* */
/* Link and access the Minidisks containing the required library(s) */
'CP SPOOL PRINTER NOCONT CLOSE'
'CP SPOOL PRINTER TO * NOHOLD CONT FORM OFF DIST OFF'
'GLOBAL LOADLIB ' CA_LOADLIB_FN IDMS_LOADLIB_FN
'GLOBAL TXTLIB ' SORT_TXTLIB_FN
/* */
/* Create the input parameter file. */
/* */
CALL CREATE_INPUT_PARM_FILE
/* */
/* Product specific files. */
'FILEDEF SYSPRT PRINTER'
'FILEDEF SYSLST PRINTER'
'FILEDEF SYSOUT PRINTER'
'FILEDEF AUDIT PRINTER'
'FILEDEF SYSIPT DISK USMEXPRT MIGIN a'
'FILEDEF SYSPCH DISK USMEXPRT MIGPCH a'
'FILEDEF SELECT DISK USMEXPRT SELECT a'
'FILEDEF VALDRPT DISK USMEXPRT VALDRPT a'
'FILEDEF WORKFIL DISK USMEXPRT WORKFIL a'
'FILEDEF WORKFL2 DISK USMEXPRT WORKFL2 a'
'FILEDEF WORKFIL2 DISK USMEXPRT WORKFIL2 a'
'FILEDEF EXTRACT DISK USMFILE EXTRACT a4 (RECFM FB LRECL 228 BLOCK 228'
'FILEDEF RHDCDEL DISK USMFILE RHDCDEL a'
'FILEDEF SCHMDEL DISK USMFILE SCHMDEL a'
'FILEDEF SUBSDEL DISK USMFILE SUBSDEL a'
'FILEDEF DDDLDDEL DISK USMFILE DDDLDDEL a'
'FILEDEF DDDLUPD DISK USMFILE DDDLUPD a'
'FILEDEF BCFUPD DISK USMFILE BCFUPD a'
'FILEDEF TABLLOD DISK USMFILE TABLLOD a'
'FILEDEF DDDLLOD DISK USMFILE DDDLLOD a'
'FILEDEF SCHMUPD DISK USMFILE SCHMUPD a'
'FILEDEF SUBSUPD DISK USMFILE SUBSUPD a'
'FILEDEF SUBSLLOD DISK USMFILE SUBSLLOD a'
'FILEDEF RHDCUPD DISK USMFILE RHDCUPD a'
'FILEDEF RHDCLOD DISK USMFILE RHDCLOD a'
'FILEDEF ADSOBGN DISK USMFILE ADSOBGN a'
'FILEDEF DDDLPGM DISK USMFILE DDDLPGM a'
'FILEDEF ADSBTAT DISK USMFILE ADSBTAT a'
'FILEDEF USERUPD DISK USMFILE USERUPD a'
'FILEDEF MIGPARM DISK INPUT_PARM_FN INPUT_PARM_FT INPUT_PARM_FM
/* */
```
/* You must create a file 'SYSIDMS INPUT a' containing the SYSIDMS */
/* parameters you use to specify your runtime environment. */
/* If you wish to run in Local Mode specify the following values : */
/* LOCAL=ON,JOURNAL=OFF,DMCL=dmcl-name */
/* If you wish to run CV an 80 byte blank record will suffice. */
/* FILEDEF SYSIDMS DISK SYSIDMS INPUT a' */
/* Insert FILEDEF statements for SORT work space as required by */
/* your SORT product. */
*/
*/
SAY 'STARTING CA IDMS/Dictionary Migrator EXPORT RUN'
SIGNAL OFF ERROR
'EXECOS OSRUN USMGRTR'
USMEXPRT_RC = RC
IF USMEXPRT_RC > 4 THEN DO
  CALL ERROR
END
'CP SPOOL PRINTER NOCONT'
'CP CLOSE PRINTER NAME USMEXPRT LISTING'
'CP SPOOL PRINTER OFF'
SAY 'USMEXPRT FINISHED WITH A RETURN CODE OF' USMEXPRT_RC
'GLOBAL LOADLIB'
'GLOBAL TXTLIB'
'FILEDEF * CLEAR'
EXIT USMEXPRT_RC
/*++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++ */
CREATE_INPUT_PARM_FILE:
/*++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++ */
SIGNAL OFF ERROR
'ERASE' INPUT_PARM_FN INPUT_PARM_FT INPUT_PARM_FM
SIGNAL ON ERROR
/*
*/
PUSH 'FFILE'
PUSH
PUSH 'EXTRACT,user extract statements'
PUSH 'DICT=srcdict,IDOBJECT=(userid,passwd),OBJDICT=objdict'
PUSH 'PROCESS,RUN=EXPORT,'
PUSH 'INPUT'
PUSH 'SET LRECL 80'
PUSH 'SET REC FM F'
'XEDIT' INPUT_PARM_FN INPUT_PARM_FT INPUT_PARM_FM ,
  '(NOPROFILE NOSCREEN NOMSG'
RETURN
/*++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++ */
ERROR:
/*++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++ */
ERROR_RC = RC
TRACE OFF; SIGNAL OFF ERROR
SAY 'NON-ZERO RETURN CODE ENCOUNTERED IN EXEC AT LINE' SIGL
'CP SPOOL PRINTER NOCONT'
'CP CLOSE PRINTER NAME USMEXPRT LISTING'
'CP SPOOL PRINTER OFF'
'GLOBAL LOADLIB'
'GLOBAL TXTLIB'
'FILEDEF * CLEAR'
EXIT ERROR_RC
/*
*/

- **yourlib** -- The file name of the load library into which you downloaded CA IDMS Dictionary Migrator.

- **idmslib** -- The file name of the load library containing your CA IDMS SUBSCHEMA and DMCL modules.

- **sortlib** -- The file name of the text library containing your sort modules.

- **a** -- The file mode for the relevant file.
Import the Extract Tape--z/VM

The USMIMPRT EXEC imports the extract tape, compares extract to object dictionary, and reports on the comparison.

⚠️ **Note:** Suggested space allocations for all work files are contained in the Work Files Table in the section "Operations."

```plaintext
/* */
TRACE OFF; SIGNAL ON ERROR
/* */
CA_LOADLIB_FN = 'yourlib'
IDMS_LOADLIB_FN = 'idmslib'
SORT_TXTLIB_FN = 'sortlib'
INPUT_PARM_FN = 'USMIMPRT'
INPUT_PARM_FT = 'MIGPARM'
INPUT_PARM_FM = 'A'

/* */
/* Link and access the Minidisks containing the required librarie(s) */
/* */
'CP SPOOL PRINTER NOCONT CLOSE'
'CP SPOOL PRINTER TO * NOHOLD CONT FORM OFF DIST OFF'
'GLOBAL LOADLIB CA_LOADLIB_FN IDMS_LOADLIB_FN'
'GLOBAL TXTLIB SORT_TXTLIB_FN

/* */
/* Create the input parameter file. */
/* */
CALL CREATE_INPUT_PARM_FILE

/* */
/* Product specific files. */
/* */
'FILEDEF SYSPRT PRINTER'
'FILEDEF SYSLST PRINTER'
'FILEDEF SYSPRT PRINTER'
'FILEDEF AUDIT PRINTER'
'FILEDEF SELECT DUMMY'
'FILEDEF SYSIPT DISK USMIMPRT MIGIN a'
'FILEDEF VALDRPT DISK USMIMPRT VALDRPT a'
'FILEDEF WORKFIL DISK USMIMPRT WORKFIL a'
'FILEDEF WORKFL2 DISK USMIMPRT WORKFL2 a'
'FILEDEF EXTRACT DISK USMFILE EXTRACT a'
'FILEDEF RHDCDEL DISK USMFILE RHDCDEL a'
'FILEDEF SCHMDEL DISK USMFILE SCHMDEL a'
'FILEDEF SUBSDEL DISK USMFILE SUBSDEL a'
'FILEDEF DDDLDEL DISK USMFILE DDDLDEL a'
'FILEDEF DDDLUPD DISK USMFILE DDDLUPD a'
'FILEDEF BCFUPD DISK USMFILE BCFUPD a'
'FILEDEF TABLOD DISK USMFILE TABLOD a'
'FILEDEF DDDLOD DISK USMFILE DDDLOD a'
'FILEDEF SCHMOPD DISK USMFILE SCHMOPD a'
'FILEDEF SUBSUPD DISK USMFILE SUBSUPD a'
'FILEDEF SUBSLOD DISK USMFILE SUBSLOD a'
'FILEDEF RHDCUPD DISK USMFILE RHDCUPD a'
'FILEDEF RHDCLOD DISK USMFILE RHDCLOD a'
'FILEDEF ADSOBN DISK USMFILE ADSOBN a'
'FILEDEF DDDLPGM DISK USMFILE DDDLPGM a'
'FILEDEF ADSTAT DISK USMFILE ADSTAT a'
'FILEDEF USERUPD DISK USMFILE USERUPD a'
'FILEDEF MIGPARM DISK' INPUT_PARM_FN INPUT_PARM_FT INPUT_PARM_FM

/* */
/* Product specific VSAM files. */
/* */
'DLBL IJSYSCT vsam_fm DSN vsamcat'
'DLBL IJSYSUC vsam_fm DSN vsamcat'
'DLBL VSAMEXT vsam_fm DSN vsamext (VSAM'
/*          */
/* You must create a file 'SYSIDMS INPUT a' containing the SYSIDMS */
/* parameters you use to specify your runtime environment. */
/* If you wish to run in Local Mode specify the following values : */
/* LOCAL=ON,JOURNAL=OFF,DMCL=dmcl-name */
/* If you wish to run CV an 80 byte blank record will suffice. */
/* */
'FILEDEF SYSIDMS DISK SYSIDMS INPUT a'
/* */
/* Insert FILEDEF statements for SORT work space as required by */
/* your SORT product. */
/* */
SAY 'STARTING CA IDMS/Dictionary Migrator IMPORT'
SIGNAL OFF ERROR
'EXECOS OSRUN USMGRTR'
USMIMPRT_RC = RC
IF USMIMPRT_RC > 4
THEN DO
  CALL ERROR
END
'CP SPOOL PRINTER NOCONT'
'CP CLOSE PRINTER NAME USMIMPRT LISTING'
'SAY 'USMIMPRT FINISHED WITH A RETURN CODE OF' USMIMPRT_RC
'GLOBAL LOADLIB'
'GLOBAL TXTLIB'
'FILEDEF * CLEAR'
EXIT USMIMPRT_RC
/
CREATE_INPUT_PARM_FILE:
/
CREATE_INPUT_PARM_FILE:
/
SIGNAL OFF ERROR
'ERASE' INPUT_PARM_FN INPUT_PARM_FT INPUT_PARM_FM
SIGNAL ON ERROR
/
PUSH 'FFILE'
PUSH
PUSH 'OBJDICT=objdict'
PUSH 'PROCESS,RUN=IMPORT,'
PUSH 'INPUT'
PUSH 'SET LRECL 80'
PUSH 'SET RECFM F'
'XEDIT' INPUT_PARM_FN INPUT_PARM_FT INPUT_PARM_FM
  '(NOPROFILE NOSCREEN NOMSG'
RETURN
/
ERROR:
/
ERROR_RC = RC
TRACE OFF; SIGNAL OFF ERROR
SAY 'NON-ZERO RETURN CODE ENCOUNTERED IN EXEC AT LINE' SIGL
'CP SPOOL PRINTER NOCONT'
'CP CLOSE PRINTER NAME USMIMPRT LISTING'
'CP SPOOL PRINTER OFF'
'GLOBAL LOADLIB'
'GLOBAL TXTLIB'
'FILEDEF * CLEAR'
EXIT ERROR_RC
/

• yourlib -- The file name of the load library into which you downloaded CA IDMS Dictionary Migrator.

• idmslib -- The file name of the load library containing your CA IDMS SUBSCHEMA and DMCL modules.
sortlib -- The file name of the text library containing your sort modules.

a -- The file mode for the relevant file.

vsam_fm -- The file mode of the accessed minidisk which contains your VSAM catalog(s) and file (s).

vsamcat -- The data set name of your VSAM catalog.

vsamext -- The data set name of the PAGUTIL VSAM file.

Upload Syntax Files -- z/VM

The USMLOAD1 EXEC is used to upload syntax files to the object dictionary using CA IDMS utilities, or using the CA user password upload utility supplied by CA. There can be from 1 to 16 steps involved in the upload, depending on your site and type of migration.

More information:

For more information on uploading syntax files, see Syntax Files Table in the section "Operations."

If you want to delete entities from the source dictionary after migration, make a second non-CHANGEONLY run using the same parameters as the real migration, specify the same dictionary for both source and object.

/* */
TRACE OFF; SIGNAL ON ERROR
/* */
/* USMLOAD1 */
/* */
CA_LOADLIB_FN = 'yourlib'
IDMS_LOADLIB_FN = 'idmslib'
/* */
/* Link and access the Minidisks containing the required librarie(s) */
'CP SPOOL PRINTER NOCONT CLOSE'
'CP SPOOL PRINTER TO * NOHOLD CONT FORM OFF DIST OFF'
'GLOBAL LOADLIB ' CA_LOADLIB_FN IDMS_LOADLIB_FN
/* */
MAX_RC = 0
/* */
CALL RHDCDEL
IF RHDCDEL_RC > 0
THEN SAY 'RHDCDEL STEP ENDED WITH RETURN CODE' RHDCDEL_RC
IF RHDCDEL RC > MAX RC
THEN MAX_RC = RHDCDEL_RC
/* */
CALL SCHMDEL
IF SCHMDEL_RC > 0
THEN SAY 'SCHMDEL STEP ENDED WITH RETURN CODE' SCHMDEL_RC
IF SCHMDEL RC > MAX RC
THEN MAX_RC = SCHMDEL_RC
/* */
CALL SUBSDEL
IF SUBSDEL_RC > 0
THEN SAY 'SUBSDEL STEP ENDED WITH RETURN CODE' SUBSDEL_RC
IF SUBSDEL RC > MAX RC
THEN MAX_RC = SUBSDEL_RC
/* */
CALL DDDLDEL
IF DDDLDEL_RC > 0
THEN SAY 'DDDLDEL STEP ENDED WITH RETURN CODE' DDDLDEL_RC
IF DDDLDEL RC > MAX_RC
THEN MAX_RC = DDDDEL_RC
/* */
CALL DDDLUPD
IF DDDLUPD_RC > 0
   THEN SAY 'DDDLUPD STEP ENDED WITH RETURN CODE' DDDLUPD_RC
IF DDDLUPD_RC > MAX_RC
   THEN MAX_RC = DDDLUPD_RC
/* */
CALL TABBLOD
IF TABBLOD_RC > 0
   THEN SAY 'TABBLOD STEP ENDED WITH RETURN CODE' TABBLOD_RC
IF TABBLOD_RC > MAX_RC
   THEN MAX_RC = TABBLOD_RC
/* */
CALL DDDLLLOD
IF DDDLLLOD_RC > 0
   THEN SAY 'DDDLLLOD STEP ENDED WITH RETURN CODE' DDDLLLOD_RC
IF DDDLLLOD_RC > MAX_RC
   THEN MAX_RC = DDDLLLOD_RC
/* */
CALL SCHMUPD
IF SCHMUPD_RC > 0
   THEN SAY 'SCHMUPD STEP ENDED WITH RETURN CODE' SCHMUPD_RC
IF SCHMUPD_RC > MAX_RC
   THEN MAX_RC = SCHMUPD_RC
/* */
CALL SUBSUPD
IF SUBSUPD_RC > 0
   THEN SAY 'SUBSUPD STEP ENDED WITH RETURN CODE' SUBSUPD_RC
IF SUBSUPD_RC > MAX_RC
   THEN MAX_RC = SUBSUPD_RC
/* */
CALL SUBSLOD
IF SUBSLOD_RC > 0
   THEN SAY 'SUBSLOD STEP ENDED WITH RETURN CODE' SUBSLOD_RC
IF SUBSLOD_RC > MAX_RC
   THEN MAX_RC = SUBSLOD_RC
/* */
CALL RHDCUPD
IF RHDCUPD_RC > 0
   THEN SAY 'RHDCUPD STEP ENDED WITH RETURN CODE' RHDCUPD_RC
IF RHDCUPD_RC > MAX_RC
   THEN MAX_RC = RHDCUPD_RC
/* */
CALL RHDCLOD
IF RHDCLOD_RC > 0
   THEN SAY 'RHDCLOD STEP ENDED WITH RETURN CODE' RHDCLOD_RC
IF RHDCLOD_RC > MAX_RC
   THEN MAX_RC = RHDCLOD_RC
/* */
CALL ADSOBN
IF ADSOBN_RC > 0
   THEN SAY 'ADSOBN STEP ENDED WITH RETURN CODE' ADSOBN_RC
IF ADSOBN_RC > MAX_RC
   THEN MAX_RC = ADSOBN_RC
/* */
CALL DDDLPGM
IF DDDLPGM_RC > 0
   THEN SAY 'DDDLPGM STEP ENDED WITH RETURN CODE' DDDLPGM_RC
IF DDDLPGM_RC > MAX_RC
   THEN MAX_RC = DDDLPGM_RC
/* */
CALL ADSBTAT
IF ADSBTAT_RC > 0
   THEN SAY 'ADSBTAT STEP ENDED WITH RETURN CODE' ADSBTAT_RC
IF ADSBTAT_RC > MAX_RC
   THEN MAX_RC = ADSBTAT_RC
/* */
CALL BCFUPD
IF BCFUPD_RC > 0
   THEN SAY 'BCFUPD STEP ENDED WITH RETURN CODE' BCFUPD_RC
IF BCFUPD RC > MAX RC
    THEN MAX RC = BCFUPD RC
/* */
CALL USERUPD
IF USERUPD RC > 0
    THEN SAY 'USERUPD STEP ENDED WITH RETURN CODE' USERUPD RC
IF USERUPD RC > MAX RC
    THEN MAX RC = USERUPD RC
/* */
'GLOBAL LOADLIB'
'FILEDEF * CLEAR'
EXIT MAX RC
/* */
*rhdcdel - delete panels and maps */
RHDCDEL:
/* */
'CP SPOOL PRINTER NOCONT CLOSE'
'CP SPOOL PRINTER TO * NOHOLD CONT FORM OFF DIST OFF'
'FILEDEF SYSLST PRINTER'
'FILEDEF SYSCH DUMMY'
'FILEDEF SYSIPT DISK FILE RHDCDEL a'
'FILEDEF SYSIDMS DISK SYSIDMS INPUT a'
SIGNAL OFF ERROR
SAY 'STARTING DELETION OF MAPS AND PANELS'
'EXECOS OSRUN RHDCMAP1'
RHDCDEL RC = RC
SIGNAL ON ERROR
'CP SPOOL PRINTER NOCONT'
'CP CLOSE PRINTER NAME RHDCDEL LISTING'
'CP SPOOL PRINTER OFF'
'FILEDEF * CLEAR'
RETURN
/* */
*schmdel - delete schema and subschema source */
SCHMDEL:
/* */
'CP SPOOL PRINTER NOCONT'
'CP SPOOL PRINTER TO * NOHOLD CONT FORM OFF DIST OFF'
'FILEDEF SYSLST PRINTER'
'FILEDEF SYSCH DUMMY'
'FILEDEF SYSIPT DISK FILE SCHMDEL a'
'FILEDEF SYSIDMS DISK SYSIDMS INPUT a'
SIGNAL OFF ERROR
SAY 'STARTING DELETION OF SCHEMA AND SUBSCHEMA SOURCE'
'EXECOS OSRUN IDMSCHEM'
IDMSCHEM RC = RC
SIGNAL ON ERROR
'CP SPOOL PRINTER NOCONT'
'CP CLOSE PRINTER NAME IDMSCHEM LISTING'
'CP SPOOL PRINTER OFF'
'FILEDEF * CLEAR'
RETURN
/* */
*subsdel - delete subschema */
/* */
*NOTE: THIS IS ONLY REQUIRED IF A SUBSCHEMA WAS MIGRATED WITHOUT ITS SCHEMA */
* */
SUBSDEL:
/* */
'CP SPOOL PRINTER NOCONT CLOSE'
'CP SPOOL PRINTER TO * NOHOLD CONT FORM OFF DIST OFF'
'FILEDEF SYSLST PRINTER'
'FILEDEF SYSCH DUMMY'
'FILEDEF SYSIPT DISK FILE SUBSDEL a'
'FILEDEF SYSIDMS DISK SYSIDMS INPUT a'
SIGNAL OFF ERROR
SAY 'STARTING DELETION OF SUBSCHEMA SOURCE'
'EXECOS OSRUN IDMSUBSC'
IDMSUBSC RC = RC
SIGNAL ON ERROR
'CP SPOOL PRINTER NOCONT'
'CP CLOSE PRINTER NAME IDMSUBSC LISTING'
'CP SPOOL PRINTER OFF'
'FILEDEF * CLEAR'
RETURN
/* ----------------------------------------------------------------- *
/* DDDDEL - DELETE OTHER ENTITIES                                    *
/* NOTE: DO NOT RUN THIS STEP FOR CHANGEONLY PROCESSING              *
/* DDDDEL:                                                           *
/* ----------------------------------------------------------------- */
'CP SPOOL PRINTER NOCONT CLOSE'
'CP SPOOL PRINTER TO * NOHOLD CONT FORM OFF DIST OFF'
'FILEDEF SYSLST PRINTER'
'FILEDEF SYSCH DUMMY'
'FILEDEF SYSIPT DISK FILE DDDDEL a'
'FILEDEF SYSIDMS DISK SYSIDMS INPUT a'
SIGNAL OFF ERROR
SAY 'STARTING DELETION OF DDDL ENTITIES'
'EXECOS OSRUN IDMSDDDL'
IDMSDDDL RC = RC
SIGNAL ON ERROR
'CP SPOOL PRINTER NOCONT'
'CP CLOSE PRINTER NAME IDMSDDDL LISTING'
'CP SPOOL PRINTER OFF'
'FILEDEF * CLEAR'
RETURN
/* */
/* DDDLPD - ADD/MODIFY CLASS, ATTRIBUTE, SYSTEM, RECORD, ELEMENT,    */
/* MESSAGE, MODULE, AND TABLE ENTITIES                               */
/* DDDLPD:                                                           */
/* ----------------------------------------------------------------- */
'CP SPOOL PRINTER NOCONT CLOSE'
'CP SPOOL PRINTER TO * NOHOLD CONT FORM OFF DIST OFF'
'FILEDEF SYSLST PRINTER'
'FILEDEF SYSCH DUMMY'
'FILEDEF SYSIPT DISK FILE DDDLPD a'
'FILEDEF SYSIDMS DISK SYSIDMS INPUT a'
SIGNAL OFF ERROR
SAY 'STARTING ADD/MODIFY OF DDDL ENTITIES'
'EXECOS OSRUN IDMSDDDL'
DDDLUPD RC = RC
SIGNAL ON ERROR
'CP SPOOL PRINTER NOCONT'
'CP CLOSE PRINTER NAME DDDLPD LISTING'
'CP SPOOL PRINTER OFF'
'FILEDEF * CLEAR'
RETURN
/* */
/* TABLLOD - GENERATE TABLE LOAD MODULES                            */
/* TABLLOD:                                                           */
/* ----------------------------------------------------------------- */
'CP SPOOL PRINTER NOCONT CLOSE'
'CP SPOOL PRINTER TO * NOHOLD CONT FORM OFF DIST OFF'
'FILEDEF SYSLST PRINTER'
'FILEDEF SYSCH DUMMY'
'FILEDEF SYSIPT DISK FILE TABLLOD a'
'FILEDEF SYSIDMS DISK SYSIDMS INPUT a'
SIGNAL OFF ERROR
SAY 'STARTING GENERATION OF TABLE LOAD MODULES'
'EXECOS OSRUN IDMSDDDL'
TABLLOD RC = RC
SIGNAL ON ERROR
'CP SPOOL PRINTER NOCONT'
'CP CLOSE PRINTER NAME TABLLOD LISTING'
'CP SPOOL PRINTER OFF'
'FILEDEF * CLEAR'
RETURN
/* */
/* DDDLLLOD - ADD/MODIFY LOAD MODULES FOR SUBSCHEMA, MAP, MAP EDIT    */
/* TABLE AND DIALOG                                                 */
/* DDDLLLOD:                                                          */
/*-----------------------------------------------------------------*/
'CP SPOOL PRINTER NOCONT CLOSE'
'CP SPOOL PRINTER TO * NOHOLD CONT FORM OFF DIST OFF'
'FILEDEF SYSLST PRINTER'
'FILEDEF SYSCH DUMMY'
'FILEDEF SYSIPT DISK FILE DDDLLOD A'
'FILEDEF SYSIDMS DISK SYSIDMS INPUT a'
SIGNAL OFF ERROR
SAY 'STARTING ADD/MODIFY OF DDDL LOAD MODULES'
'EXECOS OSRUN IDMSDDDL'
DDDLLOD RC = RC
SIGNAL ON ERROR
'CP SPOOL PRINTER NOCONT'
'CP CLOSE PRINTER NAME DDDLLOD LISTING'
'CP SPOOL PRINTER OFF'
'FILEDEF * CLEAR'
RETURN
/*-----------------------------------------------------------------*/
/* SCHMUPD - ADD SCHEMA SOURCE*/
/*-----------------------------------------------------------------*/
'CP SPOOL PRINTER NOCONT CLOSE'
'CP SPOOL PRINTER TO * NOHOLD CONT FORM OFF DIST OFF'
'FILEDEF SYSLST PRINTER'
'FILEDEF SYSCH DUMMY'
'FILEDEF SYSIPT DISK FILE SCHMUPD A'
'FILEDEF SYSIDMS DISK SYSIDMS INPUT a'
SIGNAL OFF ERROR
SAY 'STARTING ADD OF SCHEMA SOURCE'
'EXECOS OSRUN IDMSCHEM'
SCHMUPD RC = RC
SIGNAL ON ERROR
'CP SPOOL PRINTER NOCONT'
'CP CLOSE PRINTER NAME SCHMUPD LISTING'
'CP SPOOL PRINTER OFF'
'FILEDEF * CLEAR'
RETURN
/*-----------------------------------------------------------------*/
/* RHDCLOD - COMPILE MAP*/
/*-----------------------------------------------------------------*/
'CP SPOOL PRINTER NOCONT CLOSE'
'CP SPOOL PRINTER TO * NOHOLD CONT FORM OFF DIST OFF'
'FILEDEF SYSLST PRINTER'
'FILEDEF SYSCH DUMMY'
'FILEDEF SYSIPT DISK FILE RHDCLOD A'
'FILEDEF SYSIDMS DISK SYSIDMS INPUT a'
SIGNAL OFF ERROR
SAY 'STARTING MAP COMPILE'
'EXECOS OSRUN RHDCMPUT'
RHDCLOD RC = RC
SIGNAL ON ERROR
'CP SPOOL PRINTER NOCONT'
'CP CLOSE PRINTER NAME RHDCLOD LISTING'
'CP SPOOL PRINTER OFF'
'FILEDEF * CLEAR'
RETURN
/*-----------------------------------------------------------------*/
/* ADSOBGN - GENERATE ADS DIALOG*/
/* NOTE:  THIS STEP CAN ONLY BE EXECUTED IF DIALOG MODULES WERE MIGRATED*/
/*-----------------------------------------------------------------*/
'CP SPOOL PRINTER NOCONT CLOSE'
'CP SPOOL PRINTER TO * NOHOLD CONT FORM OFF DIST OFF'
'FILEDEF SYSLST PRINTER'
'FILEDEF SYSCH DUMMY'
'FILEDEF SYSIPT DISK FILE ADSOBGN A'
'FILEDEF SYSIDMS DISK SYSIDMS INPUT a'
SIGNAL OFF ERROR
SAY 'STARTING GENERATION OF ADS DIALOGS'
'EXECOS OSRUN ADSBGM'
ADSBGMN RC = RC
SIGNAL ON ERROR
'CP SPOOL PRINTER NOCONT'
'CP CLOSE PRINTER NAME ADSBGM LISTING'
'CP SPOOL PRINTER OFF'
'FILEDEF * CLEAR'
RETURN
/* ----------------------------- */
/* SDDLPGM - ADD/MODIFY PROGRAM SYNTAX */
SDDLPGM:
/* ----------------------------- */
'CP SPOOL PRINTER NOCONT CLOSE'
'CP SPOOL PRINTER TO * NOHOLD CONT FORM OFF DIST OFF'
'FILEDEF SYSLST PRINTER'
'FILEDEF SYSCH DUMMY'
'FILEDEF SYSIPT DISK FILE SDDLPGM A'
'FILEDEF SYSIDMS DISK SYSIDMS INPUT a'
SIGNAL OFF ERROR
SAY 'STARTING ADD/MODIFY OF PROGRAM SYNTAX'
'EXECOS OSRUN IDMSDDL'
SDDLPGM RC = RC
SIGNAL ON ERROR
'CP SPOOL PRINTER NOCONT'
'CP CLOSE PRINTER NAME SDDLPGM LISTING'
'CP SPOOL PRINTER OFF'
'FILEDEF * CLEAR'
RETURN
/* */
/* SUBSUPD - ADD SUBSCHEMA SOURCE */
SUBSUPD:
/* */
'CP SPOOL PRINTER NOCONT CLOSE'
'CP SPOOL PRINTER TO * NOHOLD CONT FORM OFF DIST OFF'
'FILEDEF SYSLST PRINTER'
'FILEDEF SYSCH DUMMY'
'FILEDEF SYSIPT DISK FILE SUBSUPD A'
'FILEDEF SYSIDMS DISK SYSIDMS INPUT a'
SIGNAL OFF ERROR
SAY 'STARTING ADD OF SUBSCHEMA SOURCE'
'EXECOS OSRUN IDMSUBSC'
SUBSUPD RC = RC
SIGNAL ON ERROR
'CP SPOOL PRINTER NOCONT'
'CP CLOSE PRINTER NAME SUBSUPD LISTING'
'CP SPOOL PRINTER OFF'
'FILEDEF * CLEAR'
RETURN
/* */
/* SUBSLOD - GENERATE SUBSCHEMA LOAD MODULE */
SUBSLOD:
/* */
'CP SPOOL PRINTER NOCONT CLOSE'
'CP SPOOL PRINTER TO * NOHOLD CONT FORM OFF DIST OFF'
'FILEDEF SYSLST PRINTER'
'FILEDEF SYSCH DUMMY'
'FILEDEF SYSIPT DISK FILE SUBSLOD A'
'FILEDEF SYSIDMS DISK SYSIDMS INPUT a'
SIGNAL OFF ERROR
SAY 'STARTING GENERATION OF SUBSCHEMA LOAD MODULE'
'EXECOS OSRUN IDMSUBSC'
SUBSLOD RC = RC
SIGNAL ON ERROR
'CP SPOOL PRINTER NOCONT'
'CP CLOSE PRINTER NAME SUBSLOD LISTING'
'CP SPOOL PRINTER OFF'
'FILEDEF * CLEAR'
RETURN
/* RHDCUPD - ADD/MODIFY PANEL AND MAP SYNTAX */
RHDCUPD:
/* ---------------------------------------------- */
'CP SPOOL PRINTER NOCONT CLOSE'
'CP SPOOL PRINTER TO * NOHOLD CONT FORM OFF DIST OFF'
'FILEDEF SYSLST PRINTER'
'FILEDEF SYSIPT DISK FILE RHDCUPD A'
'FILEDEF SYSIDMS DISK SYSIDMS INPUT a'
SIGNAL OFF ERROR
SAY 'STARTING ADD/MODIFY OF PANEL AND MAP SYNTAX'
'EXECOS OSRUN RHDCMAP1'
RHDCUPD_RC = RC
SIGNAL ON ERROR
'CP SPOOL PRINTER NOCONT'
'CP close PRINTER NAME RHDCUPD LISTING'
'CP SPOOL PRINTER OFF'
'FILEDEF * CLEAR'
RETURN

/* ADSBTAT - UPDATE TASK APPLICATION TABLE */
/* NOTE: THIS STEP IS ONLY FOR APPLICATIONS WHICH USE THE */
/* REVISED ADSA. */
ADSBTAT:
/* ---------------------------------------------- */
'CP SPOOL PRINTER NOCONT CLOSE'
'CP SPOOL PRINTER TO * NOHOLD CONT FORM OFF DIST OFF'
'FILEDEF SYSLST PRINTER'
'FILEDEF SYSIPT DISK FILE ADSBTAT A'
'FILEDEF SYSIDMS DISK SYSIDMS INPUT a'
SIGNAL OFF ERROR
SAY 'STARTING UPDATE OF TASK APPLICATION TABLE'
'EXECOS OSRUN ADSBTAT'
ADSBTAT_RC = RC
SIGNAL ON ERROR
'CP SPOOL PRINTER NOCONT'
'CP close PRINTER NAME ADSBTAT LISTING'
'CP SPOOL PRINTER OFF'
'FILEDEF * CLEAR'
RETURN

/* BCFUPD - ADD SQL ENTITY DEFINITIONS */
/* --------------------------------------- */
'CP SPOOL PRINTER NOCONT CLOSE'
'CP SPOOL PRINTER TO * NOHOLD CONT FORM OFF DIST OFF'
'FILEDEF SYSLST PRINTER'
'FILEDEF SYSIPT DISK FILE BCFUPD A'
'FILEDEF SYSIDMS DISK SYSIDMS INPUT a'
SIGNAL OFF ERROR
SAY 'STARTING UPLOAD OF SQL ENTITY DEFINITIONS'
'EXECOS OSRUN IDMSBCF'
BCFUPD_RC = RC
SIGNAL ON ERROR
SAY 'BCFUPD STEP ENDED WITH RETURN CODE' BCFUPD_RC
'CP SPOOL PRINTER NOCONT'
'CP close PRINTER NAME USMSQLOD LISTING'
'CP SPOOL PRINTER OFF'
'FILEDEF * CLEAR'
RETURN

/* USERUPD - UPDATE USER PASSWORDS */
/* NOTE: THIS STEP IS ONLY FOR MIGRATIONS WHICH DO NOT USE */
/* DISCONNECT=(USER). THE PASSWORD FROM THE SOURCE */
/* DICTIONARY IS PLACED INTO THE OBJECT DICTIONARY */
/* IF THIS STEP IS NOT RUN, THE PASSWORD ASSIGNED TO ALL */
/* MIGRATED USERS WILL BE "DBMSINT" */
USERUPD:
Delete Migrated Source from Source Dictionary - z/VM

The USMLOAD3 EXEC is an optional job that, when executed, deletes the migrated source from the source dictionary after migration.

⚠️ **Note:** If you want to delete entities from the source dictionary after migration, make a second non-CHANGEONLY run using the same parameters as the real migration, specify the same dictionary for both source and object.
`/*                                                                   */
'CP SPOOL PRINTER NOCONT CLOSE'
'CP SPOOL PRINTER TO * NOHOLD CONT FORM OFF DIST OFF'
'GLOBAL LOADLIB ' CA_LOADLIB_FN IDMS_LOADLIB_FN
`/*                                                                   */
MAX_RC = 0
`/*                                                                   */
CALL RHDCDEL
IF RHDCDEL_RC > 0
   THEN SAY 'RHDCDEL STEP ENDED WITH RETURN CODE' RHDCDEL_RC
IF RHDCDEL_RC > MAX_RC
   THEN MAX_RC = RHDCDEL_RC
`/*                                                                   */
CALL SCHMDEL
IF SCHMDEL_RC > 0
   THEN SAY 'SCHMDEL STEP ENDED WITH RETURN CODE' SCHMDEL_RC
IF SCHMDEL_RC > MAX_RC
   THEN MAX_RC = SCHMDEL_RC
`/*                                                                   */
CALL SUBSDEL
IF SUBSDEL_RC > 0
   THEN SAY 'SUBSDEL STEP ENDED WITH RETURN CODE' SUBSDEL_RC
IF SUBSDEL_RC > MAX_RC
   THEN MAX_RC = SUBSDEL_RC
`/*                                                                   */
CALL DDDLDEL
IF DDDLDEL_RC > 0
   THEN SAY 'DDDLDEL STEP ENDED WITH RETURN CODE' DDDLDEL_RC
IF DDDLDEL_RC > MAX_RC
   THEN MAX_RC = DDDLDEL_RC
`/*                                                                   */
'GLOBAL LOADLIB'
'FILEDEF * CLEAR'
EXIT MAX_RC
`/*                                                                   */
'-----------------------------------------------------------------
DELETE PANELS AND MAPS
-----------------------------------------------------------------
RHDCDEL:
/* ---------------------------- */
'CP SPOOL PRINTER NOCONT CLOSE'
'CP SPOOL PRINTER TO * NOHOLD CONT FORM OFF DIST OFF'
'FILEDEF SYSLST PRINTER'
'FILEDEF SYSPCH DUMMY'
'FILEDEF SYSIPT DISK_FILE RHDCDEL a'
'FILEDEF SYSIDMS DISK SYSIDMS INPUT a'
SIGNAL OFF ERROR
SAY 'STARTING DELETE OF PANS AND MAPS'
'EXECOS OSRUN RHDCMAP1'
RHDCDEL_RC = RC
SIGNAL ON ERROR
'CP SPOOL PRINTER NOCONT'
'CP CLOSE PRINTER NAME RHDCDEL LISTING'
'CP SPOOL PRINTER OFF'
'FILEDEF * CLEAR'
RETURN
/* ---------------------------- */
// DELETE SCHEMA AND SUBSCHEMA SOURCE
/* NOT IN RELEASE 10, USE IDMSCHEM TO DELETE SCHEMAS */
SCHMDEL:
/* ---------------------------- */
'CP SPOOL PRINTER NOCONT CLOSE'
'CP SPOOL PRINTER TO * NOHOLD CONT FORM OFF DIST OFF'
'FILEDEF SYSLST PRINTER'
'FILEDEF SYSPCH DUMMY'
'FILEDEF SYSIPT DISK_FILE SCHMDEL a'
'FILEDEF SYSIDMS DISK SYSIDMS PARMS a'
SIGNAL OFF ERROR
SAY 'STARTING DELETE OF SCHEMA AND SUBSCHEMA SOURCE'
'EXECOS OSRUN IDMSDDDL'
SCHMDEL_RC = RC
SIGNAL ON ERROR
'CP SPOOL PRINTER NOCONT'
'CP CLOSE PRINTER NAME SCHMDEL LISTING'
'CP SPOOL PRINTER OFF'
'FILEDEF * CLEAR'
RETURN

/* delete schema and subschema source
   * note: this is required only if a subschema was migrated without its schema

subsdel:

'CP SPOOL PRINTER NOCONT CLOSE'
'CP SPOOL PRINTER TO * NOHOLD CONT OFF DIST OFF'
'FILEDEF SYSLST PRINTER'
'FILEDEF SYSPOCH DUMMY'
'FILEDEF SYSPCH DUMMY'
'FILEDEF SYSEIPT DISK FILE SUBSDEL a'
'FILEDEF SYSEIDMS DISK SYSIDMS PARMS a'
SIGNAL OFF ERROR
SAY 'starting delete of subschema source'
'EXECOS OSRUN IDMSUBSC'
SUBSDEL_RC = RC
SIGNAL OFF ERROR
SAY 'starting delete of subschema source'
'EXECOS OSRUN IDMSUBSC'
SUBSDEL_RC = RC
SIGNAL ON ERROR
'CP SPOOL PRINTER NOCONT'
'CP CLOSE PRINTER NAME SUBSDEL LISTING'
'CP SPOOL PRINTER OFF'
'FILEDEF * CLEAR'
RETURN

/* delete record, element and table entities

DDDLDEL:

'CP SPOOL PRINTER NOCONT CLOSE'
'CP SPOOL PRINTER TO * NOHOLD CONT OFF DIST OFF'
'FILEDEF SYSLST PRINTER'
'FILEDEF SYSPOCH DUMMY'
'FILEDEF SYSEIPT DISK FILE DDDLDEL a'
'FILEDEF SYSEIDMS DISK SYSIDMS PARMS a'
SIGNAL OFF ERROR
SAY 'starting delete of record, element and table entities'
'EXECOS OSRUN IDMSDDL'
DDDLDEL_RC = RC
SIGNAL ON ERROR
'CP SPOOL PRINTER NOCONT'
'CP CLOSE PRINTER NAME DDDLDEL LISTING'
'CP SPOOL PRINTER OFF'
'FILEDEF * CLEAR'
RETURN

error:

ERROR
ERROR_RC = RC
TRACE OFF; SIGNAL OFF ERROR
SAY 'non-zero return code encountered in exec at line' SIGL
'EXECOS OSRUN IDMSLOAD'
'GLOBAL LOADLIB'
'FILEDEF * CLEAR'
EXIT ERROR_RC

yourlib
-- the file name of the load library into which you downloaded CA IDMS Dictionary
- **yourlib** -- The file name of the load library into which you downloaded CA IDMS Dictionary Migrator.

- **idmslib** -- The file name of the load library containing your CA IDMS SUBSCHEMA and DMCL modules.

- **a** -- The file mode for the relevant file.

## PROCESS Parameter

All CA IDMS Extractor parameter input is processed by the Database Extract Component and is obtained from a PROCESS parameter statement. The information tells the Database Extract Component:

- The name of the Selection Criteria Specification

- The user ID or LTERM ID under which the specification was saved

- The optional name of a transient subschema to be used during Database Load Component processing

- When using a transient subschema, the sign-on ID and password (if required) for the target dictionary

- An optional subschema to be used by the Database Extract Component, instead of the source subschema named in the Selection Criteria Specification

- An optional name of the user exit load module to be invoked whenever a database record meets its Record Level Selection Criteria

- An optional keyword to indicate that the source and target subschemas are not to be compared.

**Note:** The online JCL Submission Component creates the PROCESS parameter statement and places it in the appropriate position in the JCL when writing the JCL stream to the operating system. This allows you to use a single JCL member for multiple Selection Criteria Specifications that reference the same source and target databases. When using the JCL Submission Component, **do not** include a PROCESS parameter statement in the JCL member. Otherwise, the Database Extract Component issues a parameter error. When USVUJCL uploads the model JCL to the CA IDMS Extractor database, the PROCESS statement is not uploaded. This description is included in case you want to submit the JCL manually without using the online JCL Submission Component.
Extracts

This option directs CA IDMS Journal Analyzer to generate extract records from an input Archive Journal. No Journal Reports or Displays are printed when the EXTRACTS option of the PROCESS parameter is specified. Only a historical Extract file is produced.

ALL

This option directs CA IDMS Journal Analyzer to perform all processing of the EXTRACTS, REPORTS, and DISPLAYS options. As a result, record generation, and printing of Journal Reports and Displays can be accomplished in a single execution of CA IDMS Journal Analyzer.

Displays

This option directs CA IDMS Journal Analyzer to honor only valid requests for Journal Displays and bypass the generation of extract records and Journal Reports. If valid user requests for Journal Displays are input, CA IDMS Journal Analyzer will create the requested Journal Displays.

PROCESS Parameter Keywords

Process

Must be coded as is. Indicates that PROCESS keywords follow.

USERId=userid

Required keyword if SPECNAME is coded. Used as primary key for selection of a Selection Criteria Specification. This is the user ID or LTERM ID under which the specification was saved. USERID and UNLOAD are mutually exclusive.

SPECname=selection-criteria-specification-name

Required keyword if UNLOAD is not coded. Used as secondary key for selection of a specification created using the online Selection Criteria Specification component. SPECNAME and UNLOAD are mutually exclusive.

UNLoad

Required keyword if SPECNAME is not coded. Used to unload (extract) the entire database accessible by the SRCSUB subschema. A specification is not used in this case. SPECNAME and UNLOAD are mutually exclusive parameters. In this case, SRCSUB is required and will be used at version 1 as both the source and target subschema.

SIGNonid=dictionary-userid
Optional based upon whether you are using a transient subschema and whether the target dictionary requires a valid SIGNON statement. The user must have authority to add, delete, and punch load modules to/from the target dictionary.

**Password=**dictionary-password

Optional based upon whether the SIGNONID requires a sign-on password for the target dictionary.

**TRANSSub=**< transient-subschema-name >
\ target-subschema-name /

Optional user-supplied name for the transient subschema used during CA IDMS Extractor load processing. If not specified, it defaults to:

- **UVhhmmss**
  Where:
  - UV is a system constant
  - hh is the hour of job start
  - mm is the minute of job start
  - ss is the second of job start

The subschema name used by the Database Load Component is taken from this parameter regardless of whether or not a transient subschema is used. If a transient subschema is not used, specify the name of a subschema that the Database Load Component can successfully use.

**SRCSub=**source-subschema-name

Optional name of a subschema to be used by the Database Extract Component. Specifying a new source subschema allows the Database Extract Component to use a subschema that was not accessible to the online Selection Criteria Specification component but is almost identical to the original subschema. The subschema you specify must contain the same record and sets as the original subschema. The user records and sets must be in the same physical sequence in both subschemas (that is, they must be copied in the same order). If these conditions are not met, unpredictable results can occur.

**SRCDMcl=**source-DMCL-name

Optional name of a DMCL to be used in conjunction with the SRCSUB subschema.

**SRCdictName=**source-dictionary-node-name

Optional name of the dictionary in which the SRCSUB subschema resides if the subschema is in a load area other than the primary dictionary specified for the CV that the Database Extract Component is going to use.

**TRANSdmcl=**transient-dmcl-name
Optional name of a DMCL to use when the LOAD Component runs in LOCAL mode or when the DMCL named in TRANSSUB should not be used.

**SRCDICTNode=source-dictionary-node-name**

Optional name of a DDS node for the SRCSUB subschema. The name identifies the communication link between CVs and CPUs. Specifying this parameter indicates that CA IDMS accesses a CV that exists in a CPU other than the one it is to be executing under to retrieve the specified SRCSUB subschema.

**SRCDBNAme= < source-db > \ segment-name /**

Optional name of a DB name or segment for the SRCSUB subschema. The DB name or segment maps the logical subschema view to a physical database.

**SRCDBNODE=source-database-node-name**

Optional name of a DDS node for the SRCSUB subschema. Specifying this parameter indicates that the Database Extract Component executes in a DDS environment and accesses a CV in another CPU.

**USERExit=user-exit-module-name**

Optional name of the user exit load module to be invoked whenever a database record meets its record level selection criteria.

**NOCompare**

Optional parameter to indicate that the source and target subschemas are not to be compared. Use it if the target subschema is not accessible to the CV under which the Database Extract Component executes.

However, inconsistencies between the source and target subschemas can cause unpredictable results.

---

### Performing the EXTRACT and LOAD Steps--USVEXEC

The sample JCL for the batch Database Extract and Database Load Components of CA IDMS Extractor are contained in:

- z/OS-Target or Distribution source library member USVEXEC
- z/VSE- SAMPJCL library member USVEXEC.S
• z/VM: USVEXEC EXEC

This JCL is used after you create and save a Selection Criteria Specification using the online Selection Criteria Specification Component.

After supplying values for the variables, you can use this JCL as is, or it can be used as a model when you create and save a JCL member of the CA IDMS Extractor database using the Edit JCL option on the Main Menu screen. You can then use the Submit option to submit the JCL for execution.

Be sure to review Allocating Space for DASD Files.

Also review the CA IDMS Extractor Data Flow Information exhibit shown below.

Notes:

• If you create this member using the Edit JCL Option, do not include the PROCESS parameter statement in the member. The appropriate PROCESS parameter statement is created and placed into the JCL for you prior to submitting the job by the Submit Option. Thus, a single JCL member can be used by more than one Selection Criteria Specification.

• If the database name DBNAME of the target database differs from that of the source database, then a SYSIDMS DBNAME parameter should be added to the JCL used in the Load step as follows:

  //SYSIDMS  DBNAME=target-dbname

Also all areas included in the specified target subschema must be included in this DBNAME table.

CA IDMS Extractor Data Flow Information
Performing the EXTRACT and LOAD Steps--USVEXEC

You can access these databases using "instream" JCL or by executing the EXTRACT step in Advantage CA-IDMS's CV.

The PROCESS statement in SYSTP can be created using the online "SUBMIT" function or it can be created manually using your normal procedures.

The PUNCHSUB step can be eliminated if the target subschema is available in OBJECT format.

The linkage editor can be substituted for LOADSUB and used to link the transient subschema into a loadfile image library instead of a directory load area.
z/OS JCL

//USVEXEC JOB job-card-parameters /*
//EXTRACT EXEC PGMS=USVXTRC,REGION=1024K /*
//STEPLIB DD DSN=your.dbx.loadlib,DISP=SHR
//SYSTCL DD DSN=your.idms.sysctl,DISP=SHR
//EXTRACTS DD DSN=your.dbx.extracts,DISP=(NEW,CATLG,DELETE),
// UNIT=DISK, VOL=SER=volser,
// SPACE=(CYL,(primary,secondary))
//COMMFILE DD DSN=your.dbx.commfile,DISP=(NEW,CATLG,DELETE),
// UNIT=DISK, VOL=SER=volser,SPACE=(TRK,1)
//SYNTAX DD DSN=your.dbx.syntax,DISP=(NEW,CATLG,DELETE),
// UNIT=DISK, VOL=SER=volser,SPACE=(TRK,1)
//SYSPRINT DD SYSOUT=a //SYSLST DD SYSOUT=a //SYSIPT DD SYSOUT=a /*

PROCESS,USERID=userid,SPECNAME=selection-criteria-specification-name,
[SIGNONID=dictionary-userid,]
[PASSWORD=directory-password,]
[TRANSIN=transient-or-target-subschema-name,]
[SRCSCB=source-subschema-name,]
[SRCDDL=source-dmcl-name,]
[SRCDICT=source-dictionary-name,]
[SRCNODE=source-db-or-segment-name,]
[SRCNAME=source-db-node-name,]
[USEREXIT=user-exit-module-name,]
[NOCOMPARE] /*

//PUNCHSUB EXEC PGMS=IDMSDDDL, REGION=1024K, CLEAR((5,LT,EXTRACT))
//*******************************************************************************/
//PUNCHSUB - Punch the target-subschema to a work-file based upon */
//*******************************************************************************/
//STEPLIB DD DSN=your.idms.loadlib,DISP=SHR
//SYSTCL DD SYSOUT=a //SYSTCL DD DSN=your.idms.sysctl,DISP=SHR
//SYSPCH DD DSN=your.idms.commfile,DISP=(OLD,DELETE)
//SYSPRINT DD SYSOUT=a //SYSPRINT DD SYSOUT=a //SYSPRINT DD SYSOUT=a /*

//ALTERSUB EXEC PGMS=USVSUBZ,REGION=1024K,PARM=’NOSPIE’/, CLEAR((5,LT,EXTRACT),(5,LT,PUNCHSUB))
//*******************************************************************************/
//ALTERSUB - Alter the target subschema contents and create a */
//*******************************************************************************/
//STEPLIB DD DSN=your.dbx.loadlib,DISP=SHR
//INPUT DD DSN=your.dbx.loadlib,DISP=(OLD,PASS)
//COMMFILE DD DSN=your.dbx.commfile,DISP=(OLD,KEEP)
//SYSTCL DD SYSOUT=a //CAIOUT DD SYSOUT=a //SYSPRINT DD SYSOUT=a /*

//OUTPUT DD DSN=your.dbx.loadlib,DISP=(NEW,PASS),UNIT=DISK,
// SPACE=(TRK,(10,5)),DCB=(BLKSIZE=80,LRECL=80,RECFM=F)
//*******************************************************************************/
//*******************************************************************************/

//LOADSUB EXEC PGMS=IDMSDDDL, REGION=1024K, CLEAR((5,LT,EXTRACT),(5,LT,PUNCHSUB))
//*******************************************************************************/
//LOADSUB - Add the transient-subschema to the target dictionary */
/*
CA IDMS - 19.0

// LOAD EXEC PGM=USVLOAD, REGION=1536K,
// COND=((5,LT,EXTRACT),(5,LT,PUNCHSUB),(0,NE,ALTERSUB),
// (5,LT,LOADSUB))
//*
// You can run the load against an existing database, although
// unexpected problems may arise if an attempt is made to add duplicate
// records to the database and the attempt is not prevented by the
// DUPLICATES NOT ALLOWED option while doing this. Otherwise, run the
// FORMAT Utility
// against the target database prior to executing this
//*
// There is no parameter statement for this step. Processing is
// controlled by information read from the EXTRACTS and COMMFILE
// files.
//*
// STEPLIB DD DSN=your.dbx.loadlib,DISP=SHR
// SYSLIB DD DSN=your.idms.loadlib,DISP=SHR
// EXTRACTS DD DSN=your.dbx.extract,DISP=SHR
// WORKFILE DD UNIT=disk, VOL=SER=volser,
// SPACE=(CYL,(primary,secondary))
// SORTWK01 DD UNIT=disk, SPACE=(CYL,(primary,secondary))
// SORTWK02 DD UNIT=disk, SPACE=(CYL,(primary,secondary))
// SORTWK03 DD UNIT=disk, SPACE=(CYL,(primary,secondary))
// SORTMSG DD SYSOUT=a //SYSPRINT DD SYSOUT=a //SYSLST DD SYSOUT=a
// SYSDUMP DD SYSOUT=a //COMMFILE DD DSN=your.dbx.commfile,DISP=SHR
// SYNTAX DD DSN=your.dbx.syntax,DISP=SHR
//*
// DELTASUB EXEC PGM=IDMSDDDL, REGION=1024K,
// COND=((5,LT,EXTRACT),(5,LT,PUNCHSUB),(0,NE,ALTERSUB),
// (5,LT,LOADSUB),EVEN)
//*****************************************************************************
//* DELTASUB - Remove the transient-subschema from the target dictionary*
// node using IDMSDDDL.*****************************************************************************
// STEPLIB DD DSN=your.idms.loadlib,DISP=SHR
// SYSLIB DD DSN=your.dbx.syntax,DISP=(OLD,DELETE,DELETE)
// SYSLST DD SYSOUT=a //SYSLST DD SYSOUT=a
// SYSPCH DD DUMMY
//*

<table>
<thead>
<tr>
<th>job card parameters</th>
<th>The job card parameters at your site.</th>
</tr>
</thead>
<tbody>
<tr>
<td>your.dbx.loadlib</td>
<td>The name of the load library into which CA IDMS Extractor load modules were link edited.</td>
</tr>
<tr>
<td>your.idms.loadlib</td>
<td>The name of the load library into which CA IDMS load modules were link edited.</td>
</tr>
<tr>
<td>your.idms.sysctl</td>
<td>The data set name of the SYSLIB file for the CV used during CA IDMS Extractor batch processing.</td>
</tr>
<tr>
<td>your.dbx.extracts</td>
<td>The name of the file to which all extracted records from your source database are written.</td>
</tr>
<tr>
<td>disk</td>
<td>An appropriate UNIT designation for each file.</td>
</tr>
<tr>
<td>volser</td>
<td>The volume serial number of the UNIT that contains the file being defined.</td>
</tr>
<tr>
<td>primary, secondary</td>
<td>The primary/secondary space allocation for files that are being allocated to disk.</td>
</tr>
<tr>
<td>your.dbx.commfile</td>
<td>The name of the file used by CA IDMS Extractor for communication between the Database Extract and Database Load components.</td>
</tr>
</tbody>
</table>
Running in Local Mode Under z/OS

This JCL assumes you are going to be running under a CV. To run in local mode:

1. Remove the SYSCTL DD statements.

2. Add the appropriate DD statement(s) for your local journal(s).

3. Add the STEPLIB DD statements for your local DMCL and for your source and transient subschemas.

4. To the EXTRACT step, add DD statements for your:
   a. CA IDMS Extractor database
   b. Source database

5. To the LOAD step, add DD statements for your target database.

If the source subschema exists in a CA IDMS Extractor dictionary load area, the PUNCHSUB and ALTERSUB steps do not need to be changed. If the source subschema exists in a load library, it must be uploaded to a dictionary load area before the transient subschema process is executed. If the source subschema exists in object format, do not execute the PUNCHSUB step, and:

1. Change the INPUT of the ALTERSUB step to be the file of the object subschema.

2. Change the LOADSUB step to execute the linkage editor instead of IDMSDDDL. Supply appropriate JCL for a link edit. The SYSLIN data set should be the OUTPUT from the ALTERSUB step. The SYSLMOD member name should be the TRANSSUB (transient SSC name) specified when the JCL was submitted. Specify an execute parameter of LET and expect various IEW0302 error messages, as well as a condition code of 8.

3. For the LOAD step, change the COND parameter from 5,LT,LOADSUB to 9,LT,LOADSUB.

4. Change the DELETSUB step to execute IEHPROGM instead of IDMSDDDL. Change the COND parameter to remove the reference to the LOADSUB step. Supply a DD statement that references the SYSLMOD data set from the LOADSUB step. The SYSIN data set should be:

   SCRATCH DSN=your.loadlib, VOL=SYSDA=volid, MEMBER=transsub

z/VSE JCL

* SS JOB JNM=USVEXEC
// JOB USVEXEC
* // OPTION LOG,PARTDUMP
* **** CREATE A SYSIDMS PARAMETER FILE *****

// UPSI 1
// DLBL SYSIDMS,'work.file.SYSIDMS',0,SD
// EXTENT SYS041,volserw,,rel-trk-blk,1
// ASSGN SYS041,DISK,VOL=volserw,SHR
// EXEC DITTO

$DITTO CSQ FILEOUT=SYSIDMS

* SYSIDMS parameters for target dictionary
*
/*
// UPSI a
// ASSGN SYS006,SYSLST
*
// ASSGN SYS000,DISK,VOL=

// DLBL tool,'tool.corelib'
// EXTENT ,volserc // DLBL idms,'idms.corelib'
// EXTENT ,volserc // LIBDEF PHASE,SEARCH=(tool.sublib,idms.sublib),TEMP

* *** CORE IMAGE LIBRARY FOR PRODUCT

* *** DBX EXTRACT FILE
// DLBL EXTRACT,'dbx.extract',99/365,SD
// EXTENT SYS010,volserw,,rel-trk-blk,amount // ASSGN SYS010,DISK,VOL=volserw,SHR

* *** DBX COMMUNICATION FILE
// DLBL COMMFL,'dbx.commfile',99/365,SD
// EXTENT SYS015,volserw,,rel-trk-blk,amount // ASSGN SYS015,DISK,VOL=volserw,SHR

* *** IDD SYNTAX FILE
// DLBL SYNTAX,'dbx.syntax',99/365,SD
// EXTENT SYS016,volserw,,rel-trk-blk,amount // ASSGN SYS016,DISK,VOL=volserw,SHR

* *** INPUT
// ASSGN SYSIPT,SYSRDR
* SYSCtl FILE

* DLBL SYSCtl,'your.sysctl.file',,SD
// EXTENT SYS000,volser,,rel-trk-blk,1
// ASSGN SYS000,DISK,VOL=volser,SHR
// DLBL SYSIDMS,'#SYSIPT',0,SD
// EXEC USVXTRC,SIZE=(USVXTRC,400K)

* SYSIDMS parameters
*
// DLBL SYSCT, '#SYSIPT',0,SD
// ASSGN your.sysctl.file
// EXTENT SYS000,volser,,rel-trk-blk,1
// ASSGN SYS000,DISK,VOL=volser,SHR
// DLBL SYSIDMS, 'your.sysctl.file',,SD
// EXEC USVXTRC,SIZE=(USVXTRC,400K)

* PUNCHSUB step. Punch the target subschema from the dictionary
*
// DLBL IJSYSYPH,'dbx.syspch',99/365,SD
// EXTENT SYSACH,volserw,,rel-trk-blk,amount // ASSGN SYSACH,DISK,VOL=volserw,SHR
// DLBL IJSYSIN,'dbx.syntax',0,SD
// EXTENT SYSIPT,volser,,rel-trk-blk,amount // ASSGN SYSIPT,DISK,VOL=volserw,SHR
// DLBL SYSIDMS, work.file.SYSIDMS'
// EXTENT SYS041,volserw // ASSGN SYS041,DISK,VOL=volserw,SHR
// DLBL SYSCTL,'your.sysctl.file',,SD
// EXTENT SYS000,volser,,rel-trk-blk,1
// ASSGN SYS000,DISK,VOL=volser,SHR
// EXEC IDMSDDDL,SIZE=512K
CLOSE SYSIPT,SYSRDR
CLOSE SYSCT,xxxx

* ALTERSUB step. Alter the target subschema and create a transient
* working copy of the target subschema.
*
// DLBL INPUT,'dbx.syspch',0,SD
// EXTENT SYS017,volserw,,rel-trk-blk,amount // ASSGN SYS017,DISK,VOL=volserw,SHR
// DLBL OUTPUT,'dbx.transsub',99/365,SD
LOADSUB step. Add the transient subschema to the target dictionary

```
// DLBL  IJSYSIN,'dbx.transsub',0,SD
// EXTENT SYSIPT,volserw,,rel-trk-bik,amount  ASSGN SYSIPT,DISK,VOL=volserw,SHR
// ASSGN SYSPCH,IGN
// DLBL  SYSCtl,'your.sysctl.file',,,SD
// EXTENT SYS000,volser,,rel-trk-bik,1
// ASSGN SYS000,DISK,VOL=volser,SHR
// DLBL SYSDMMS,work.file.SYSDMMS'
// EXTENT SYS041,volserw  // ASSGN SYS041,DISK,VOL=volserw,SHR
// EXEC IDMSDDDL,SIZE=512K
CLOSE SYSIPT,SYSRDR
```

LOAD step. Add the extracted records to the target database

```
// DLBL  EXTRACT,'dbx.extracts',0,SD
// EXTENT SYS010,volserw,,rel-trk-bik,amount  ASSGN SYS010,DISK,VOL=volserw,SHR
// DLBL WORKFIL,'dbx.workfile',99/365,SD
// EXTENT SYS011,volserw,,rel-trk-bik,1
// DLBL COMMFILE, 'dbx.commfile',0,SD
// EXTENT SYS016,volserw,,rel-trk-bik,amount  ASSGN SYS016,DISK,VOL=volserw,SHR
// DLBL SYNTAX,'dbx.syntax',0,SD
// EXTENT SYS016,volserw,,rel-trk-bik,amount  ASSGN SYS016,DISK,VOL=volserw,SHR
// DLBL SORTWK1,'SORT.WORK1',0,SD
// EXTENT SYS01
// ASSGN SYS01,DISK,VOL=volserw,SHR
// DLBL SORTWK2,'SORT.WORK2',0,SD
// EXTENT SYS02
// ASSGN SYS02,DISK,VOL=volserw,SHR
// DLBL SORTWK3,'SORT.WORK3',0,SD
// EXTENT SYS03
// ASSGN SYS03,DISK,VOL=volserw,SHR
// DLBL SORTWK4,'SORT.WORK4',0,SD
// EXTENT SYS04
// ASSGN SYS04,DISK,VOL=volserw,SHR
// DLBL SYSSD,SYSLST
// DLBL SYSCtl,'your.sysctl.file',,,SD
// EXTENT SYS000,volser,,rel-trk-bik,1
// ASSGN SYS000,DISK,VOL=volser,SHR
// DLBL SYSDMMS,work.file.SYSDMMS'
// EXTENT SYS041,volserw  // ASSGN SYS041,DISK,VOL=volserw,SHR
// EXEC IDMSDDDL,SIZE=512K
CLOSE SYSIPT,SYSRDR
```

SYSIDMS parameters

DELETESUB step. Delete the transient subschema from the dictionary

```
// ASSGN SYSPCH,IGN
// DLBL IJSYSIN,'dbx.transsub',0,SD
// EXTENT SYSIPT,volserw,,rel-trk-bik,amount  ASSGN SYSIPT,DISK,VOL=volserw,SHR
// DLBL SYSCtl,'your.sysctl.file',,,SD
// EXTENT SYS000,volser,,rel-trk-bik,1
// ASSGN SYS000,DISK,VOL=volser,SHR
// DLBL SYSDMMS,work.file.SYSDMMS'
// EXTENT SYS041,volserw  // ASSGN SYS041,DISK,VOL=volserw,SHR
// EXEC IDMSDDDL,SIZE=512K
CLOSE SYSIPT,SYSRDR
```

SS E0J
1. The UPSI switch to invoke CA IDMS CV as specified in your IDMSOPTI module.

tool The file name of the core image library into which the executable phases of CA IDMS Extractor were installed.

tool. corelib The file ID of the core image library into which the executable phases of CA IDMS Extractor were installed.

volser_ The volume serial number or generic assignment of the volume on which the file, as specified in the previous statement, resides. The following letters identify the file in question:

c=core image library

w=work file

r=relocatable library

s=source statement library

idms The file name of the core image library into which the executable phases of CA IDMS were installed.

idms. corelib The file ID of the core image library into which the executable phases of CA IDMS were installed.

.sublib The sublibrary name of the z/VSE library specified in the previous file name.

dbx. extract The file ID of the file to which the CA IDMS Extractor extract records are written.

rel-trk- Relative track or relative block number: The starting position of the DASD for storage of the file specified in the previous statement.

blk

amoun_ The number of tracks or blocks you need for storage of the work file specified in the previous statement.

t

dbx. commf The file ID of the file to which the CA IDMS Extractor communication records are written.

ile

dbx. syntax The file ID of the file to which the IDD syntax records are written.

dbx. syspch The file ID of the file to which the CA IDMS subschema object records are written.

xxx The physical device or logical unit to which the SYSPCH logical unit is to be assigned.

your. sysctl. file The file ID of the SYSCTL file for the CV used during CA IDMS Extractor batch processing.

dbx. transsub The file ID of the file to which the transient subschema is written.

Running in Local Mode Under z/VSE

This JCL assumes you are going to be running under a CV. To run in local mode:
1. Supply the appropriate UPSI switch.

2. Add the appropriate JCL statement(s) for your local journal(s).

3. Add the correct SYSIDMS parameters for your local DMCL.

4. To the EXTRACT step, add JCL statements for your:
   a. CA IDMS Extractor database
   b. source database.

5. If the target subschema exists in a CA IDMS Extractor dictionary load area, the PUNCHSUB and ALTERSUB steps do not need to be changed. If the target subschema exists in a core image library, it must be uploaded to a dictionary load area before the transient subschema process is executed. If the target subschema exists in object format, do not execute the PUNCHSUB step, but change the INPUT of the ALTERSUB step to be the file of the object subschema. Change the LOADSUB step to execute the linkage editor instead of IDMSDDDL. Supply appropriate JCL for a link edit. The linkage editor input data set should be the OUTPUT from the ALTERSUB step. The linkage editor core image phase name should be the TRANSSUB (transient SSC name) specified when the JCL was submitted. Specify a linkage editor parameter of LET and expect various error messages that result from the IDMSDDDL statements appended to the punched subschema object statements.

6. To the LOAD step, add JCL statements for your target database.

7. Change the DELETSUB step to execute LIBR instead of IDMSDDDL. Supply a statement that references the core image library from the LOADSUB step. The SYSIPT dataset should be: DELETE transsub.

8. Remove the DLBLs for the SYSCTL file.

z/VM EXEC

/* */
TRACE OFF; SIGNAL ON ERROR

/* ----- EXEC MODIFICATIONS ---------- 'USVEXEC' SAMPLE EXEC ------- */

/*
DBX_LOADLIB_FN = 'dbxlib'
DBX_LOADLIB_FT = 'LOADLIB'
DBX_LOADLIB_FM = '***'
/* */
IDMS_LOADLIB_FN = 'idmslib'
IDMS_LOADLIB_FT = 'LOADLIB'
IDMS_LOADLIB_FM = '***'
/* */
SORT_SORTLIB_FN = 'sortlib'
SORT_SORTLIB_FT = 'TXTLIB'
SORT_SORTLIB_FM = '***'
/* */
USVXTRC_SYSIPT_FN = 'USVXTRC'
USVXTRC_SYSIPT_FT = 'SYSIPT'
USVXTRC_SYSIPT_FM = 'a'
/* */
MAX_RC = 0
/* */
/* */
CALL CREATE_USVXTRC_SYSIPT_FILE
/* */
CALL EXECUTE_USVXTRC
IF USVXTRC_RC > 4
   THEN CALL USVXTRC_ERROR
END
IF USVXTRC_RC > MAX_RC
   THEN MAX_RC = USVXTRC_RC
END
/* */
CALL EXECUTE_PUNCHSUB
IF USVXTRC_RC > 4
   THEN CALL PUNCHSUB_ERROR
END
IF USVXTRC_RC > MAX_RC
   THEN MAX_RC = USVXTRC_RC
END
/* */
CALL EXECUTE_ALTERSUB
IF USVXTRC_RC > 0
   THEN CALL ALTERSUB_ERROR
END
IF USVXTRC_RC > MAX_RC
   THEN MAX_RC = USVXTRC_RC
END
/* */
CALL EXECUTE_LOADSUB
IF USVXTRC_RC > 4
   THEN CALL LOADSUB_ERROR
END
IF USVXTRC_RC > MAX_RC
   THEN MAX_RC = USVXTRC_RC
END
/* */
CALL EXECUTE_USVLOAD
IF USVLOAD_RC > 4
   THEN CALL USVLOAD_ERROR
END
IF USVXTRC_RC > MAX_RC
   THEN MAX_RC = USVXTRC_RC
END
/* */
IF LOADSUB_RC = 0
   THEN CALL EXECUTE_DELTSUB
IF USVXTRC_RC > 5
   THEN CALL DELTSUB_ERROR
END
IF USVXTRC_RC > MAX_RC
   THEN MAX_RC = USVXTRC_RC
END
/* */
'CP SPOOL PRINTER NOCONT'
'CP CLOSE PRINTER NAME USVEXEC LISTING'
'CP SPOOL PRINTER OFF'
'GLOBAL LOADLIB'
'GLOBAL_sortlib'
'FILEDEF * CLEAR'
/* */
EXIT MAX_RC
/* */
/*++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++*/
CREATE_USVXTRC_SYSIPT_FILE:
/*++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++*/
SIGNAL OFF ERROR
'ERASE' USVXTRC_SYSIPT_FN USVXTRC_SYSIPT_FT USVXTRC_SYSIPT_FM
SIGNAL ON ERROR
/* */
PUSH 'FFILE'
PUSH
PUSH ' NOCOMPARE'
PUSH ' USEREXIT=user-exit-module-name,'
PUSH ' SRCDBNODE=source-db-node-name,'
PUSH ' SRCDBNAME=source-db-or-segment-name,'
PUSH ' SRCDICTNODE=source-dictionary-node-name,'
PUSH ' SRCDICTNAME=source-dictionary-name,'
PUSH ' SRCDMCL=source-dmcl-name,'
PUSH ' SRCSUB=source-subschema-name,'
PUSH ' SRCDICTNODE=source-dictionary-node-name,'
PUSH ' SRCDICTNAME=source-dictionary-name,'
PUSH ' TRANSSUB=transient-or-target-subschema-name,'
PUSH ' PASSWORD=dictionary-password,'
PUSH ' SIGNONID=dictionary-userid,'
PUSH ' SPECNAME=selection-criteria-specification-name,'
/* PUSH ' UNLOAD,SRCSUB=source-subschema-name,' */
PUSH ' USERID=userid,'
PUSH ' PROCESS,'
PUSH ' INPUT'
PUSH ' SET RECL 80'
PUSH ' SET RECFM F'
'XEDIT' USVXTRC_SYSIPT_FN USVXTRC_SYSIPT_FT USVXTRC_SYSIPT_FM,
'(NOPROFILE NOSCREEN NOMSG'
RETURN
/* */ 
EXECUTE_USVXTRC:
/*++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++ */
FILEDEF * CLEAR'
'GLOBAL LOADLIB' DBX LOADLIB_FN IDMS LOADLIB_FN
'FILEDEF EXTRACTS DISK FILE EXTRACTS a'
'FILEDEF COMMFILE DISK FILE COMMFILE a'
'FILEDEF SYNTAX DISK FILE SYNTAX a'
'FILEDEF SYSIDMS DISK SYSIDMS INPUT a'
'FILEDEF SYSCTI DISK fn SYSCTL fm'
'FILEDEF SYSIPT DISK USVXTRC SYSIPT_FN USVXTRC SYSIPT_FT USVXTRC SYSIPT_FM
'FILEDEF SYSLST PRINTER'
FILEDEF SYSPRINT PRINTER
SIGNAL OFF ERROR
SAY 'STARTING CA-IDMS/DATABASE EXTRACTOR EXTRACT'
EXECOS'
'OSRUN USVXTRC'
USVXTRC_RC = RC
SIGNAL ON ERROR
RETURN
/* */
EXECUTE_PUNCHSUB:
/*++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++ */
/*PUNCHSUB - Punch the target-subschema to a work-file based upon */
/* values from previous step. */
/* */
FILEDEF * CLEAR'
'GLOBAL LOADLIB' DBX LOADLIB_FN IDMS LOADLIB_FN
'FILEDEF SYSPCH DISK FILE DDLUPD a'
'FILEDEF SYSIPT DISK FILE SYNTAX a'
'FILEDEF SYSIDMS DISK SYSIDMS INPUT a'
'FILEDEF SYSCTI DISK fn SYSCTI fm'
'FILEDEF SYSLST PRINTER'
SIGNAL OFF ERROR
SAY 'STARTING CA-IDMS/DATABASE EXTRACTOR PUNCH SBSCHMA'
EXECOS'
'OSRUN IDMSDDDL PARM=''CVMACH=machine-id''
USVXTRC_RC = RC
SIGNAL ON ERROR
RETURN
/* */
EXECUTE_ALTERSUB:
/*++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++ */
/*ALTERSUB - Alter the target subschema contents and create a */
/* */
/* transient working copy of the target-subschema. */

FILEDEF * CLEAR
GLOBAL LOADLIB DBX_LOADLIB_FN IDMS_LOADLIB_FN
FILEDEF INPUT DISK FILE DDLUPD a'
FILEDEF COMMFILE DISK FILE COMMFILE a'
FILEDEF OUTPUT DISK FILE ALTSUB a'
FILEDEF SYSLST PRINTER
FILEDEF CAIOUT PRINTER
/

SIGNAL OFF ERROR
SAY 'STARTING CA-IDMS/DATABASE EXTRACTOR ALTER SUBSCHEMA'
EXECOS
'OSRUN USVSUBZ'
USVXTRC_RC = RC
SIGNAL ON ERROR
RETURN
/

EXECUTE_LOADSUB:
/

FILEDEF * CLEAR
GLOBAL LOADLIB IDMS_LOADLIB_FN
FILEDEF SYSPCH DUMMY
FILEDEF SYSLST PRINTER
FILEDEF SYSPCH DISK FILE ALTSUB a'
FILEDEF SYSIDMS DISK SYSIDMS INPUT a'
FILEDEF SYSCTL DISK fn SYSCTL fm'
/

SIGNAL OFF ERROR
SAY 'STARTING CA-IDMS/DATABASE EXTRACTOR RELOAD SUBSCHEMA'
EXECOS
'OSRUN IDMSDDDL PARM=''CVMACH=machine-id'''
LOADSUB_RC = RC
SIGNAL ON ERROR
RETURN
/

EXECUTE_USVLOAD:
/

FILEDEF * CLEAR
GLOBAL LOADLIB DBX_LOADLIB_FN IDMS_LOADLIB_FN
GLOBAL TXTLIB SORT_SORTLIB_FN
FILEDEF EXTRACTS DISK FILE EXTRACTS a'
FILEDEF WORKFILE DISK FILE WORKFILE a'
FILEDEF COMMFILE DISK FILE COMMFILE a'
FILEDEF SYNTAX DISK FILE SYNTAX a'
FILEDEF SYSIDMS DISK SYSIDMS INPUT a'
FILEDEF SYSCTL DISK fn SYSCTL fm'
FILEDEF SYSPRINT PRINTER
FILEDEF SYSCTL PRINTER'
FILEDEF SYSPRINT PRINTER
SIGNAL OFF ERROR
SAY 'STARTING CA-IDMS/DATABASE EXTRACTOR LOAD DATABASE'
EXECOS
'OSRUN USVLOAD PARM=''MSGOPT=A'''
USVXTRC_RC = RC
SIGNAL ON ERROR
RETURN
/

EXECUTE_DELT SUB:
/

SIGNAL OFF ERROR
SAY 'STARTING CA-IDMS/DATABASE EXTRACTOR DELT SUB'
EXECOS
'OSRUN USVLOAD PARM=''MSGOPT=A'''
USVXTRC_RC = RC
SIGNAL ON ERROR
RETURN
/

SIGNAL OFF ERROR
SAY 'STARTING CA-IDMS/DATABASE EXTRACTOR DELT SUB'
EXECOS
'OSRUN USVLOAD PARM=''MSGOPT=A'''
USVXTRC_RC = RC
SIGNAL ON ERROR
RETURN
/

SIGNAL OFF ERROR
SAY 'STARTING CA-IDMS/DATABASE EXTRACTOR DELT SUB'
EXECOS
'OSRUN USVLOAD PARM=''MSGOPT=A'''
USVXTRC_RC = RC
SIGNAL ON ERROR
RETURN
/

SIGNAL OFF ERROR
SAY 'STARTING CA-IDMS/DATABASE EXTRACTOR DELT SUB'
EXECOS
'OSRUN USVLOAD PARM=''MSGOPT=A'''
USVXTRC_RC = RC
SIGNAL ON ERROR
RETURN
/
/*DELTSUB - Remove the transient-subschema from the target dictionary**/
/*
 * node using IDMSDDL.
* */
******************************************************************************
/* */
'FILEDEF * CLEAR'
'GLOBAL LOADLIB' IDMS_LOADLIB_FN
'FILEDEF SYSPCH DUMMY'
'FILEDEF SYSLIST PRINTER'
'FILEDEF SYSLST DISK FILE SYNTAX a'
'FILEDEF SYSIDMS DISK SYSIDMS INPUT a'
'FILEDEF SYSCTL DISK fn SYSCTL fm'
/* */
SIGNAL OFF ERROR
SAY 'STARTING CA-IDMS/DATABASE EXTRACTOR DELETE SUBSCHEMA'
'EXECOS'
'OSRUN IDMSDDL PARM=''CVMACH=machine-id'''
USVXTRC RC = RC
SIGNAL ON ERROR
RETURN
/* */
/*++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++ */
USVXTRC_ERROR:
/*++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++ */
TRACE OFF; SIGNAL OFF ERROR
/* */
SAY 'UNACCEPTABLE USVXTRC RETURN CODE' USVXTRC_RC
/* */
'CP SPOOL PRINTER NOCONT'
'CP CLOSE PRINTER NAME USVEXEC LISTING'
'CP SPOOL PRINTER OFF'
'GLOBAL LOADLIB'
'GLOBAL sortlib'
'FILEDEF * CLEAR'
/* */
EXIT USVXTRC_RC
/* */
/*++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++ */
PUNCHSUB_ERROR:
/*++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++ */
TRACE OFF; SIGNAL OFF ERROR
/* */
SAY 'UNACCEPTABLE PUNCHSUB RETURN CODE' USVXTRC_RC
/* */
'CP SPOOL PRINTER NOCONT'
'CP CLOSE PRINTER NAME USVEXEC LISTING'
'CP SPOOL PRINTER OFF'
'GLOBAL LOADLIB'
'GLOBAL sortlib'
'FILEDEF * CLEAR'
/* */
EXIT USVXTRC_RC
/* */
/*++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++ */
ALTERSUB_ERROR:
/*++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++ */
TRACE OFF; SIGNAL OFF ERROR
/* */
SAY 'UNACCEPTABLE ALTERSUB RETURN CODE' USVXTRC_RC
/* */
'CP SPOOL PRINTER NOCONT'
'CP CLOSE PRINTER NAME USVEXEC LISTING'
'CP SPOOL PRINTER OFF'
'GLOBAL LOADLIB'
'GLOBAL sortlib'
'FILEDEF * CLEAR'
/* */
EXIT USVXTRC_RC
/* */
/*++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++ */
LOADSUB_ERROR:
/*++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++ */
TRACE OFF; SIGNAL OFF ERROR
/* */
  SAY 'UNACCEPTABLE LOADSUB RETURN CODE' USVXTRC_RC
  'CP SPOOL PRINTER NOCONT'
  'CP CLOSE PRINTER NAME USVEXEC LISTING'
  'CP SPOOL PRINTER OFF'
  'GLOBAL LOADLIB'
  'GLOBAL sortlib'
  'FILEDEF * CLEAR'
  /* */
  EXIT USVXTRC_RC
  /* */
  /*++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++*/
USVLOAD_ERROR:
  /*++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++*/
TRACE OFF; SIGNAL OFF ERROR
  SAY 'UNACCEPTABLE USVLOAD RETURN CODE' USVXTRC_RC
  /* */
  'CP SPOOL PRINTER NOCONT'
  'CP CLOSE PRINTER NAME USVEXEC LISTING'
  'CP SPOOL PRINTER OFF'
  'GLOBAL LOADLIB'
  'GLOBAL sortlib'
  'FILEDEF * CLEAR'
  /* */
  EXIT USVXTRC_RC
  /* */
  /*++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++*/
DELTSUB_ERROR:
  /*++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++*/
TRACE OFF; SIGNAL OFF ERROR
  SAY 'UNACCEPTABLE DELTSUB RETURN CODE' USVXTRC_RC
  /* */
  'CP SPOOL PRINTER NOCONT'
  'CP CLOSE PRINTER NAME USVEXEC LISTING'
  'CP SPOOL PRINTER OFF'
  'GLOBAL LOADLIB'
  'GLOBAL sortlib'
  'FILEDEF * CLEAR'
  /* */
  EXIT USVXTRC_RC
  /* */
  /*++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++*/
ERROR:
  /*++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++*/
ERROR_RC = RC
  TRACE OFF; SIGNAL OFF ERROR
  /* */
  SAY 'NON-ZERO RETURN CODE ENCOUNTERED IN EXEC AT LINE' SIGL
  /* */
  'CP SPOOL PRINTER NOCONT'
  'CP CLOSE PRINTER NAME USVEXEC LISTING'
  'CP SPOOL PRINTER OFF'
  'GLOBAL LOADLIB'
  'GLOBAL sortlib'
  'FILEDEF * CLEAR'
  /* */
  EXIT ERROR_RC

<table>
<thead>
<tr>
<th>dbxlib</th>
<th>The file name of the load library into which you downloaded CA IDMS Extractor.</th>
</tr>
</thead>
<tbody>
<tr>
<td>idmslib</td>
<td>The file name of the load library which contains your CA IDMS load modules.</td>
</tr>
<tr>
<td>sortlib</td>
<td>The file name of the text library which contains your Sort text files.</td>
</tr>
<tr>
<td>fn</td>
<td>The file name, file type, and file mode of the SYSCTL file used by the Central Version (CV) during batch processing.</td>
</tr>
</tbody>
</table>
The file mode for each relevant file. If necessary, change the file name and file type to conform to your site's standards.

Running in Local Mode Under z/VM

The JCL assumes you are going to be running under a CV. To run in local mode:

1. Add the appropriate FILEDEFs for your local journal(s).

2. Add the FILEDEF for the LOADLIB containing your local DMCL and for your source and transient subschemas.

3. Add the above loadlib to the GLOBAL LOADLIB statement.

4. To the EXTRACT step, add FILEDEFs for:
   a. your CA IDMS Extractor database
   b. your source database.

5. To the LOAD step, add FILEDEFs for your target database.

6. Remove or comment out the SYSCTL FILEDEF statement.

7. Alter the parameters in your SYSIDMS INPUT file accordingly.

If the target subschema exists in a CA IDMS dictionary load area, the PUNCHSUB and ALTERSUB steps do not need to be changed. If the target subschema exists in a load library, it must be uploaded to a dictionary load area before the transient subschema process is executed.

If the target subschema exists in object format, do not execute the PUNCHSUB step, and:

1. Change the INPUT of the ALTERSUB step to be the file of the object subschema.

2. Change the LOADSUB step to execute the linkage editor instead of IDMSDDDL. (Supply appropriate JCL for a link edit.) The SYSLIN dataset should be the OUTPUT from the ALTERSUB step. The SYSLMOD member name should be the TRANSSUB (transient SSC name) specified when the EXEC was submitted. Specify an execute parameter of LET and expect variousIEW0302 error messages, as well as a condition code of 8.

3. For the LOAD step, change the return code check from 5 to 9.

4. Change the DELETSUB step to execute a LOADLIB DELETE command for the transsub member name.

Printing JCL--USVPJCL

The sample JCL contained in:

- z/OS-Target or Distribution source library member USVPJCL
• z/VSE - SAMPJCL library member USVPJCL.

• z/VM - USVPJCL EXEC

Is used by the Print Utility or the Online JCL Utilities Component after JCL has been uploaded, or edited and saved using the online JCL Component.

The JCL and keys for USVPJCL are shown below.

CA IDMS Extractor z/OS JCL

```
//USVPJCL JOB (job card parameters)
  JOB USVPJCL
  * OPTION LOG,PARTDUMP
  UPSI a
  * *** CORE IMAGE LIBRARY FOR PRODUCT
  DLBL tool,'tool.corelib'
  EXTENT ,volserc // DLBL idms,'idms.corelib'
  EXTENT ,volserc // LIBDEF *,SEARCH=(tool.sublib,idms.sublib),TEMP
  * *** INPUT - SYNTAX
```

```
job card parameter

your.dbx. The name of the load library into which CA IDMS Extractor load modules were link edited.
your.idms. The name of the load library into which CA IDMS load modules were link edited.
your.idms. The data set name of the SYSCTL file for the CV used during CA IDMS Extractor batch processing.
a An appropriate SYSOUT class for your environment.
PROCESS Must be coded as is. Indicates that PROCESS keywords follow.
userid Required keyword used as primary key for selection of a CA IDMS Extractor JCL member. This keyword is the user ID or LTERM ID under which the member was saved.
jcl-member-name Required keyword, used as secondary key for selection of a CA IDMS Extractor JCL member.
```

JCL - z/VSE

```
* SS JOB JNM=USVPJCL
  JOB USVPJCL
  * OPTION LOG,PARTDUMP
  UPSI a
  * *** CORE IMAGE LIBRARY FOR PRODUCT
  DLBL tool,'tool.corelib'
  EXTENT ,volserc // DLBL idms,'idms.corelib'
  EXTENT ,volserc // LIBDEF *,SEARCH=(tool.sublib,idms.sublib),TEMP
  * *** INPUT - SYNTAX
```

10-Jan-2018 223/240
The UPSI switch to invoke CA IDMS CV as specified in your IDMSOPTI module.

The file name of the core image library into which the executable phases of CA IDMS Extractor were installed.

The file ID of the core image library into which the executable phases of CA IDMS Extractor were installed.

The volume serial number or generic assignment of the volume on which the file, as specified in the previous statement, resides. The following letters identify the file in question:

- c=core image library
- w=work file
- r=relocatable library
- s=source statement library

The file name of the core image library into which the executable phases of CA IDMS were installed.

The file ID of the core image library into which the executable phases of CA IDMS were installed.

The sublibrary name of the z/VSE library specified in the previous file name.

The file ID of the SYSCTL file for the CV used during CA IDMS Extractor batch processing.

Link and access the Minidisks containing the required libraries.

Create the input parameter file.
CALL CREATE_INPUT_PARM_FILE
/*
 * Product specific files.
 'FILEDEF SYSLST PRINTER'
 'FILEDEF SYSDUMP PRINTER'
 'FILEDEF SYSIPT DISK USVPJCL SYSIPT A'
/*
 * You must create a file `SYSIDMS INPUT A` containing the SYSIDMS parameters you use to specify your runtime environment.
 'FILEDEF SYSIDMS DISK SYSIDMS INPUT A'
 'FILEDEF SYSCTL DISK fn SYSCTL fm'
/*
SIGNAL OFF ERROR
SAY 'STARTING PRINT OF CA-IDMS/DATABASE EXTRACTOR JCL'
'EXECOS OSRUN USVPJCL'
USVPJCL RC = RC
 'CP SPool PRINTER NOCONT'
 'CP CLOSE PRINTER NAME USVPJCL LISTING'
 'CP SPool PRINTER OFF'
SAY 'USVPJCL FINISHED WITH A RETURN CODE OF' USVPJCL_RC
 'GLOBAL LOADLIB'
 'FILEDEF * CLEAR'
EXIT USVPJCL_RC
/* */

CREATE_INPUT_PARM_FILE: /*++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++*/

SIGNAL OFF ERROR
'ERASE USVPJCL SYSIPT A'
SIGNAL ON ERROR
/* */
PUSH 'FFILE'
PUSH
PUSH ' JCLMBR=jcl.member.name'
PUSH ' USERID=userid,'
PUSH ' PROCESS,'
PUSH ' INPUT'
PUSH 'SET LRECL 80'
PUSH 'SET RECFM F'
'XEDIT USVPJCL SYSIPT A'
,(NOPROFILE NOSCREEN NOMSG'
RETURN
/* */

ERROR:
/*++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++*/
ERROR_RC = RC
TRACE OFF; SIGNAL OFF ERROR
/* */
SAY 'NON-ZERO RETURN CODE ENCOUNTERED IN EXEC AT LINE' SIGL
/* */
 'CP SPool PRINTER NOCONT'
 'CP CLOSE PRINTER NAME USVPJCL LISTING'
 'CP SPool PRINTER OFF'
 'GLOBAL LOADLIB'
 'FILEDEF * CLEAR'
EXIT ERROR_RC
/* */

<table>
<thead>
<tr>
<th>dbxlib</th>
<th>The file name of the load library into which you downloaded CA IDMS Extractor.</th>
</tr>
</thead>
<tbody>
<tr>
<td>idmslib</td>
<td>The file name of the load library which contains your CA IDMS load modules.</td>
</tr>
<tr>
<td>fn</td>
<td>The file name, file type, and file mode of the SYSCTL file used by the Central Version (CV) during batch processing.</td>
</tr>
<tr>
<td>SYSCTL</td>
<td>fn</td>
</tr>
<tr>
<td>fm</td>
<td>PROCESS Must be coded as is. Indicates that PROCESS keywords follow.</td>
</tr>
<tr>
<td>userid</td>
<td></td>
</tr>
</tbody>
</table>

10-Jan-2018
Required keyword used as primary key for selection of a CA IDMS Extractor JCL member. This keyword is the user ID or LTERM ID under which the member was saved.

Required keyword, used as secondary key for selection of a CA IDMS Extractor JCL member.

Running in Local Mode

To run in local mode under z/OS, and z/VSE, remove the SYSCTL DD statement and add appropriate DD statements for your CA IDMS Extractor database and your local journal. To run in local mode under z/VM, remove the SYSCTL FILEDEF statement and add appropriate FILEDEF statements for your CA IDMS Extractor database and your local journals.

Printing a Selection Criteria Specification--USVPSPC

The sample JCL contained in:

- z/OS-Target or Distribution source library member USVPSPC
- z/VSE- SAMPJCL library member USVPSPC.S
- z/VM- USVPSPC EXEC

Is used by the batch Specification Print Utility or the online Specification Utilities Component after a Selection Criteria Specification has been created and saved using the online Selection Criteria Specification Component. The JCL and keys for USVPSPC are shown below.

z/OS JCL 2

```plaintext
//USVPSPC JOB (job-card-parameters)
//* /PRTSPEC EXEC PGM=USVPSPC,REGION=1024K
//STEPLIB DD DSN=your.dbx.loadlib,DISP=SHR
// DD DSN=your.idms.loadlib,DISP=SHR
//SYSCTL DD DSN=your.idms.sysctl,DISP=SHR
//SYSLST DD SYSOUT=a //SYSUDUMP DD SYSOUT=a /*
//SYSIPT DD
// PROCESS,USERID=userid,SPECNAME=selection-criteria-specification-name
```

<table>
<thead>
<tr>
<th>job-card-parameters</th>
<th>The job card parameters at your site.</th>
</tr>
</thead>
<tbody>
<tr>
<td>your.dbx.loadlib</td>
<td>The name of the load library into which CA IDMS Extractor load modules were link edited.</td>
</tr>
<tr>
<td>your.idms.loadlib</td>
<td>The name of the load library into which CA IDMS load modules were link edited.</td>
</tr>
</tbody>
</table>
your.idms.sysctl

The data set name of the SYSCTL file for the CV used during CA IDMS Extractor batch processing.

a

An appropriate SYSOUT class for your environment.

PROCESS

Must be coded as is. Indicates that PROCESS keywords follow.

userid

Required keyword used as primary key for selection of a specification. This is the user ID or LTERM ID under which the specification was saved.

selection-criteria-specification-name

Required keyword, used as secondary key for selection of a specification.

z/VSE JCL 2

* SS JOB JNM=USVPSPC
// JOB USVPSPC
* // OPTION LOG,PARTDUMP
// UPSI 1
* **** CREATE A SYSIDMS PARAMETER FILE *****
// UPSI 1
* // DBLB anyname,'work.file.SYSIDMS',0,SD
// EXTENT SYSnnn,volser,,start-track,1
// ASSGN SYSnnn,DISK,VOL=volser,SHR
// EXEC DITTO
$DITTO CSQ FILEOUT=ANYNAME
ECHO=ON
JOURNAL=OFF
LOCAL=ON-OR-OFF
DMCL=DMCLNAME
DICTNAME=DICTNAME
DBN=DBX
*/
/* // UPSI a
* // **** CORE IMAGE LIBRARY FOR PRODUCT
// DBLB dbms,'dbx.corelib'
// EXTENT ,volser // DBLB idms,'idms.corelib'
// EXTENT ,volser // LIBDEF *,SEARCH=(dbms.sublib,idms.sublib),TEMP
* // **** INPUT - SYNTAX
// ASSGN SYSIPT,SYSRDR
* // **** OUTPUT - PRINT REPORT FILE
// ASSGN SYS006,SYSLST
* // DBLB SYSCTL,'your.sysctl.file',,sd
// EXTENT SYS000,volser,,start-track,1
// ASSGN SYS000,DISK,VOL=volser,SHR
* // DBLB SYSIDMS,'work.file.SYSIDMS',0,SD
// EXTENT SYSnnn,volser // ASSGN SYSnnn,DISK,VOL=volser,SHR
* // EXEC USVPSPC,SIZE=(USVPSPC,400K)
PROCESS,USERID=userid,SPECNAME=selection-criteria-specification-name
/* */
* SS E0J

The UPSI switch to invoke CA IDMS CV as specified in your IDMSOPTI module.
<table>
<thead>
<tr>
<th>Term</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>anyna</td>
<td>Any suitable name for your SYSIDMS file. The name you have chosen must be identical to the FILEOUT value in the DITTO step.</td>
</tr>
<tr>
<td>SYSnnn</td>
<td>An appropriate SYS number for your SYSIDMS file.</td>
</tr>
<tr>
<td>volser_</td>
<td>The volume serial number or generic assignment of the volume on which the file, as specified in the previous statement, resides. The following letters identify the file in question:</td>
</tr>
<tr>
<td></td>
<td>c=core image library</td>
</tr>
<tr>
<td></td>
<td>w=work file</td>
</tr>
<tr>
<td></td>
<td>r=relocatable library</td>
</tr>
<tr>
<td></td>
<td>s=source statement library</td>
</tr>
<tr>
<td>start-track</td>
<td>The starting track number for the specified file.</td>
</tr>
<tr>
<td>dbms</td>
<td>The file name of the core image library into which the executable phases of CA IDMS Extractor were installed.</td>
</tr>
<tr>
<td>dbx. corelib</td>
<td>The file ID of the core image library into which the executable phases of CA IDMS Extractor were installed.</td>
</tr>
<tr>
<td>idms</td>
<td>The file name of the core image library into which the executable phases of CA IDMS were installed.</td>
</tr>
<tr>
<td>idms. corelib</td>
<td>The file ID of the core image library into which the executable phases of CA IDMS were installed.</td>
</tr>
<tr>
<td>dbms. sublib</td>
<td>The sublibrary name of the z/VSE library specified in the previous file name.</td>
</tr>
<tr>
<td>your. sysctl.</td>
<td>The file ID of the SYSCTL file for the CV used during CA IDMS Extractor batch processing.</td>
</tr>
</tbody>
</table>

### z/VM EXEC 1

```plaintext
/* */
TRACE OFF; SIGNAL ON ERROR
/*
DBX_LOADLIB_FN  = 'dbxlib'
IDMS_LOADLIB_FN = 'idmslib'
/*
   Link and access the Minidisks containing the required library(s)
*/
'CP SPOOL PRINTER NOCONT CLOSE'
'CP SPOOL PRINTER TO * NOHOLD CONT FORM OFF DIST OFF'
'GLOBAL LOADLIB ' DBX_LOADLIB_FN IDMS_LOADLIB_FN
/*
   Create the input parameter file.
*/
CALL CREATE_INPUT_PARM_FILE
/*
   Product specific files.
*/
'FILEDEF SYSIPT DISK USVPSPC SYSIPT A'
'FILEDEF SYSLST PRINTER'
'FILEDEF SYSUDUMP PRINTER'
/*
```
You must create a file 'SYSIDMS INPUT A' containing the SYSIDMS parameters you use to specify your runtime environment.

```c
/*
 'FILEDEF SYSIDMS DISK SYSIDMS INPUT A'
 'FILEDEF SYSCTL DISK fn SYSCTL fm'
 */ */
SIGNAL OFF ERROR
SAY 'STARTING PRINT OF CA-IDMS/DATABASE EXTRACTOR SPECIFICATION'
'EXECOS OSRUN USVPSPC'
USVPSPC RC = RC
 'CP SPOOL PRINTER NOCONT'
 'CP CLOSE PRINTER NAME USVPSPC LISTING'
 'CP SPOOL PRINTER OFF'
SAY 'USVPSPC FINISHED WITH A RETURN CODE OF' USVPSPC_RC
 'GLOBAL LOADLIB'
 'FILEDEF * CLEAR'
EXIT USVPSPC_RC

/* */
CREATE_INPUT_PARM_FILE:
/* */
SIGNAL OFF ERROR
'ERASE USVPSPC SYSIPT A'
SIGNAL ON ERROR

/* */
PUSH 'FFILE'
PUSH
PUSH
PUSH ' SPECNAME=specification-name'
PUSH ' USERID=userid,'
PUSH ' PROCESS,'
PUSH ' INPUT'
PUSH 'SET LRECL 80'
PUSH 'SET RECFM F'
'XEDIT USVPSPC SYSIPT A',
   (Noprofile Noscreen NomsG'
RETURN

/* */
ERROR:
/* */
ERROR_RC = RC
TRACE OFF; SIGNAL OFF ERROR
SAY 'NON-ZERO RETURN CODE ENCOUNTERED IN EXEC AT LINE' SIGL

'CP SPOOL PRINTER NOCONT'
 'CP CLOSE PRINTER NAME USVPSPC LISTING'
 'CP SPOOL PRINTER OFF'
 'GLOBAL LOADLIB'
 'FILEDEF * CLEAR'
EXIT ERROR_RC

/* */
```

<table>
<thead>
<tr>
<th><strong>dbxlib</strong></th>
<th>The file name of the load library into which you downloaded CA IDMS Extractor.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>idmslib</strong></td>
<td>The file name of the load library which contains your CA IDMS load modules.</td>
</tr>
<tr>
<td><strong>fn</strong></td>
<td>The file name, file type, and file mode of the SYSCTL file used by the Central Version (CV) during batch processing.</td>
</tr>
<tr>
<td><strong>SYSCTL</strong></td>
<td>fn SYSCTL fm</td>
</tr>
<tr>
<td><strong>fm</strong></td>
<td><strong>PROCESS</strong> Must be coded as is. Indicates that PROCESS keywords follow.</td>
</tr>
<tr>
<td><strong>userid</strong></td>
<td>Required keyword used as primary key for selection of a CA IDMS Extractor JCL member. This keyword is the user ID or LTERM ID under which the member was saved.</td>
</tr>
<tr>
<td><strong>specificat</strong></td>
<td>Required keyword, used as secondary key for selection of a CA IDMS Extractor Selection Criteria Specification.</td>
</tr>
</tbody>
</table>
Running in Local Mode 1

To run in local mode under z/OS and z/VSE, remove the SYSCTL DD statement and add appropriate DD statements for your CA IDMS Extractor database and your local journal. To run in local mode under z/VM, remove the SYSCTL FILEDEF statement and add appropriate FILEDEF statements for your CA IDMS Extractor database and your local journals.

Printing Online Documentation--USVPRINT

The CA Online Documentation Print Utility provided with CA IDMS Extractor allows error messages and other product information to be printed upon request.

- z/OS and-Target or Distribution source library member USVPRINT
- z/VSE- SAMPJCL library member USVPRINT
- z/VM- USVPRINT EXEC

Online documentation modules for CA IDMS Extractor screens other than list or confirmation screens are shown in the table below.

The printed version of the online documentation is presented one screen per page and includes page reference indexes for screen options. Characters highlighted in the online documentation appear bolded in the printed version.
# Online Documentation Modules

<table>
<thead>
<tr>
<th>Module Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Generic Modules:</strong></td>
<td></td>
</tr>
<tr>
<td>COMMANDS</td>
<td>Generic EDITOR documentation.</td>
</tr>
<tr>
<td><strong>CA IDMS Extractor Modules:</strong></td>
<td></td>
</tr>
<tr>
<td>USVAARE</td>
<td>Area Selection List</td>
</tr>
<tr>
<td>USVABER</td>
<td>Begin Edit Record Selection List</td>
</tr>
<tr>
<td>USVACPL</td>
<td>Copy Utility Member List</td>
</tr>
<tr>
<td>USVACPY</td>
<td>Copy Utility</td>
</tr>
<tr>
<td>USVAEDJ</td>
<td>Edit JCL Entry Screen</td>
</tr>
<tr>
<td>USVAENT</td>
<td>Database Entry Point Selection Menu</td>
</tr>
<tr>
<td>USVAFLS</td>
<td>Field Level Selection Criteria Screen</td>
</tr>
<tr>
<td>USVAFMM</td>
<td>Field Mismatch Screen</td>
</tr>
<tr>
<td>USVAIDX</td>
<td>Index Entry Point Selection List</td>
</tr>
<tr>
<td>USVAJUT</td>
<td>JCL Utilities</td>
</tr>
<tr>
<td>USVAKEY</td>
<td>Display PF Key Values</td>
</tr>
<tr>
<td>USVAMEN</td>
<td>Table of Contents for Online Documentation</td>
</tr>
<tr>
<td>USVAREC</td>
<td>Record Entry Point Selection List</td>
</tr>
<tr>
<td>USVARLS</td>
<td>Record Level Selection Criteria Screen</td>
</tr>
<tr>
<td>USVASET</td>
<td>Path-Record Set Selection List</td>
</tr>
<tr>
<td>USVASPC</td>
<td>Specify Database Build Specification Screen</td>
</tr>
<tr>
<td>USVASUB</td>
<td>Submit JCL for Execution</td>
</tr>
<tr>
<td>USVASUT</td>
<td>Specification Utilities</td>
</tr>
<tr>
<td>USVAUTI</td>
<td>Utilities Menu</td>
</tr>
<tr>
<td>USVDBP</td>
<td>Creating a CA IDMS Extractor Database Path</td>
</tr>
<tr>
<td>USVEXIT</td>
<td>Exiting from CA IDMS Extractor</td>
</tr>
<tr>
<td>USVGENRL</td>
<td>General Information about CA IDMS Extractor</td>
</tr>
<tr>
<td>USVMSG</td>
<td>CA IDMS Extractor Messages</td>
</tr>
<tr>
<td>USVXFERF</td>
<td>Transfer Facility</td>
</tr>
</tbody>
</table>
Customizing the CA IDMS Extractor Environment--

USVTPARM

The CA IDMS Extractor customization macro gives you the ability to change the following operational parameters:

- The task code used to invoke CA IDMS Extractor.
- The dictionary into which the CA IDMS Extractor online documentation modules were loaded at installation.
- The number of entries to allocate for the CA IDMS Extractor set stack.
- From whom a user can copy other JCL members and Selection Criteria Specifications.
- The default for the RETAIN PHYSICAL SEQUENCE OF MEMBER RECORDS IN THE SET? field on the Record Level Selection Criteria screen.
- The default for the EXTRACT ALL OWNERS FOR EXTRACTED RECURSIVE RECORDS? field on the Record Level Selection Criteria screen.
- The default for the BEGIN VIEWING/EDITING IN THE MIDDLE OF A PATH DEFINITION? field on the Specify Database Extract Specification screen.
- Whether to have message NLYZ008 be a warning (W) message or an error (E) message. NLYZ008 is displayed at extract time when a mandatory member is being extracted without its owner. An error message prevents the Selection Criteria Specification from being used.

These runtime options can be changed at anytime after initial product installation, either before or after SMP/E ACCEPT (z/OS) or MSHP (z/VSE) processing. See the CA IDMS installation guide for your operating system for detailed instructions on processing customization macro changes.

USVTPARM--Customizing the CA IDMS Extractor Environment

```
TASK=task-code,
HLPDICT=help-dictionary-name,  
HLPNODE=help-dictionary-node,  
HLPVERS=help-dictionary-version, 
STKENTS=set-stack-size, 
COPY=USER|DBXADMIN|ANYONE, 
RETS=Yes|No, 
XRECURO=Yes|No, 
BGINMID=Yes|No, 
NLYZ008=Warning|Error
```

Key to Customizing the CA IDMS Extractor Environment

- task-code
  The task code used to invoke CA IDMS Extractor. The default is DBX.
- **help-dictionary-name**
  The 1-8 character dictionary name of the dictionary into which the CA IDMS Extractor online documentation modules were loaded at installation. The default is NULL for the default dictionary.

- **help-dictionary-node**
  The 1-8 character dictionary node for help-dictionary-name. The default is NULL if no DDS.

- **help-version-number**
  The version number at which USVTUTOR online documentation modules were added. From 1 to 9999. The default is 1.

- **set-stack-size**
  The number of 8-byte entries to allocate for the CA IDMS Extractor set stack. From 30-1000. The default is 50.

- **COPY=USER|DBXADMIN|ANYONE**
  Specify from whom a user can copy other JCL members and Selection Criteria Specifications:
  
  - **USER**
    From only the user
  
  - **DBXADMIN**

  - From the user plus any global members under the DBXADMIN user ID

  - **ANYONE**
    From any user on the CA IDMS Extractor database. This is the default.

- **RETSEQ=Yes|No**
  Specify YES or NO to define the default (Yes or No) for the RETAIN PHYSICAL SEQUENCE OF MEMBER RECORDS IN THE SET? field on the Record Level Selection Criteria screen. The default is YES.

- **XRECURO=Yes|No**
  Specify YES or NO to define the default (Yes or No) for the EXTRACT ALL OWNERS FOR EXTRACTED RECURSIVE RECORDS? field on the Record Level Selection Criteria screen. The default is YES.

- **BGINMID=Yes|No**
  Specify YES or NO to define the default (Yes or No) for the BEGIN VIEWING/EDITING IN THE MIDDLE OF A PATH DEFINITION? field on the Specify Database Extract Specification screen. The default is NO.

- **NLYZ008=Warning|Error**
  Specify WARNING or ERROR to define whether the message NLYZ008 is a warning message or an error message. NLYZ008 is displayed at extract time when a mandatory member is being extracted without its owner. An error message prevents the Selection Criteria Specification from being used. The default is WARNING.
Writing a User Exit Module--CUVUSRXA

The copy book contained in sample JCL library member:

- For Assembler--CUVUSRXA (z/OS), or CUVUSRXA.S (z/VSE)
- For COBOL--CUVUSRXC (z/OS), or CUVUSRXC.S (z/VSE)

Describes the input parameters that will be passed to a user exit module that you provide. This exit module is called by the Database Extract Component when:

- CA IDMS Extractor walking the source database looking for records to extract, and
- A record meets its Record Level Selection Criteria.

You can change the record data, add or delete fields to/from the record data, or prevent the record from being written to the file.

CUVUSRXA Rules and Guidelines

If you change the record's length, you must update the record length (USVRECLN) to reflect the record's new data length. The record buffer is 32,732 bytes in length.

Moving data past the end of any of the passed parameters causes unpredictable results.

CUVUSRXA--Copy Book for Writing a User Exit Module

COPY CUVUSRXA

* THIS COPY BOOK DESCRIBES THE PARAMETERS PASSED TO AN ASSEMBLER USER EXIT MODULE WRITTEN FOR THE DATABASE EXTRACTOR (DBX) EXTRACT COMPONENT. REFER TO CUVUSRXC FOR THE COBOL VERSION. THE USER EXIT IS CALLED:

1) ONLY FOR RECORDS WHICH MEET THEIR RECORD/FIELD LEVEL SELECTION CRITERIA.
2) PRIOR TO EACH RECORD BEING WRITTEN TO THE EXTRACT FILE.

You can change the record data, add or delete fields to/from the record data, or prevent the record from being written to the file. If you change the record's length, you must update the record length (USVRECLN) to reflect the record's new data length. The record buffer is 32,732 bytes in length. Moving data past the end of any of the passed parameters will cause unpredictable results.

There are three control blocks passed to the user exit. Each control block is preceded by a table which explains each parameter field.

* * *

COPY CUVUSRXA

* STATUS CONTROL BLOCK: THIS IS THE FIRST BLOCK OF INFORMATION *
* PASSED TO THE USER EXIT. IT REFLECTS
* THE GENERAL CONDITIONS UNDER WHICH THE USER
* EXIT IS BEING INVOKED. TOTAL LENGTH IS 69
* BYTES.
*
* ITEM USAGE LENGTH DESCRIPTION
* _____________________________ _____________ ____________

| SUBSCHEMA   | ALPHANUMERIC | 8 BYTES | NAME OF SUBSCHEMA BEING USED. |
| DATABASE    | BINARY      | 4 BYTES | DATABASE KEY FOR CURRENT OF |
| KEY         | BINARY      | 2 BYTES | CURRENT OF TRANSACTION RECORD. |
| CANCEL      | BINARY      | 2 BYTES | ZERO TO INDICATE THAT THE |
| INDICATOR   | BINARY      | 2 BYTES | CURRENT OF TRANSACTION RECORD|
|             |             |         | SHOULD BE EXTRACTED; NON-ZERO |
|             |             |         | TO INDICATE THAT THE RECORD |
|             |             |         | SHOULD NOT BE EXTRACTED. |
| RECORD      | ALPHANUMERIC | 18 BYTES | NAME OF RECORD TYPE FOR CURRENT |
| AREA        | ALPHANUMERIC | 18 BYTES | NAME OF AREA TO WHICH CURRENT OF |
| SET NAME    | ALPHANUMERIC | 18 BYTES | NAME OF SET IN WHICH CURRENT OF |
|             |             |         | TRANSACTION RECORD IS ASSIGNED.* |
|             |             |         | AS EITHER AN OWNER OR MEMBER |
|             |             |         | (THE NAME OF THE SET THAT DBX |
|             |             |         | IS CURRENTLY WALKING); IF |
|             |             |         | SPACES, THEN THE CURRENT OF |
|             |             |         | TRANSACTION RECORD WAS OBTAINED* |
| SET         | ALPHANUMERIC | 1 BYTE  | '0' INDICATES THAT THE CURRENT |
| RELATION-   | SHIP        |         | OF TRANSACTION RECORD IS AN |
|             |             |         | OWNER OF SET NAMED ABOVE; 'M' |
|             |             |         | INDICATES THAT THE RECORD IS A |
|             |             |         | MEMBER OF THE SET. |

**USVPARM1 DSECT** PARAMETER #1 - STATUS INFORMATION
**USVSSNM DS CL08** NAME OF SUBSCHEMA BEING USED
**USVDKEY DS F** DATABASE KEY OF RECORD
**USVCANCL DS H** CANCEL INDICATOR:
  0 TO EXTRACT THE RECORD
  NON-ZERO TO CANCEL THE EXTRACTION.
**USVRECNM DS CL18** NAME OF RECORD
**USVARENM DS CL10** NAME OF AREA IN WHICH RECORD RESIDES
**USVSETNM DS CL18** NAME OF SET IN WHICH RECORD PARTICIPATES X
  - SPACES IF AREA SWEEP RECORD.
**USVRELAT DS CL01** RELATIONSHIP OF RECORD TO SET:
**USVOWNER EQU 'O'** OWNER OF THE SET
**USVMEMBR EQU 'M'** MEMBER OF THE SET.
**USVPRM1L EQU *-USVPARM1** LENGTH OF PARAMETER #1

***

* RECORD CONTROL BLOCK: THIS IS THE SECOND BLOCK OF INFORMATION
* PASSED TO THE USER EXIT. IT CONTAINS
* INFORMATION REGARDING THE CURRENT OF
* TRANSACTION RECORD TYPE WHICH HAS MET ITS
* RECORD LEVEL SELECTION CRITERIA. TOTAL
* LENGTH IS 64 BYTES.
*
* ITEM USAGE LENGTH DESCRIPTION
* _____________________________ _____________ ____________

| RECORD      | ALPHANUMERIC | 18 BYTES | NAME OF RECORD TYPE FOR CURRENT |
| AREA        | ALPHANUMERIC | 18 BYTES | NAME OF AREA TO WHICH CURRENT OF |
| RECORD ID   | BINARY      | 2 BYTES  | IDENTIFICATION NUMBER FOR |
| LENGTH      | BINARY      | 2 BYTES  | LENGTH (DATA ONLY), IN BYTES, OF |
|             |             |         | CURRENT OF TRANSACTION RECORD; IF |
|             |             |         | YOU CHANGE THE LENGTH OF THE |
|             |             |         | RECORD, YOU MUST UPDATE THIS |
|             |             |         | FIELD TO REFLECT THE RECORD'S |
NEW DATA LENGTH.

CONTROL BINARY 2 BYTES LENGTH (DATA ONLY), IN BYTES, OF CURRENT OF TRANSACTION RECORD UP TO AND INCLUDING THE LAST CALC OR SORT-CONTROL FIELD.

LENGTH

MAXIMUM BINARY 2 BYTES ACTUAL LENGTH OF FIXED-LENGTH RECORD OR MAXIMUM LENGTH OF A VARIABLE-LENGTH RECORD, IN BYTES.

LENGTH

DATABASE BINARY 4 BYTES DATABASE KEY FOR CURRENT OF TRANSACTION RECORD.

KEY

RECORD BINARY 4 BYTES NUMBER OF LOWEST PAGE ON WHICH RECORDS OF THIS TYPE CAN EXIST.

LOW PAGE

RECORD BINARY 4 BYTES NUMBER OF HIGHEST PAGE ON WHICH RECORDS OF THIS TYPE CAN EXIST.

HIGH PAGE

AREA LOW BINARY 4 BYTES NUMBER OF LOWEST PAGE OF AREA TO WHICH RECORD IS ASSIGNED.

PAGE

AREA HIGH BINARY 4 BYTES NUMBER OF HIGHEST PAGE OF AREA TO WHICH RECORD IS ASSIGNED.

DATABASE

RECORD

AREA LOW

AREA HIGH

---------------------------------------------------------------------

USVPARM2 DSECT

PARAMETER #2 - RECORD INFORMATION

USRNAME DS CL18

NAME OF RECORD

USRARMN DS CL18

NAME OF AREA IN WHICH RECORD RESIDES

USVRECID DS H

RECORD IDENTIFICATION NUMBER

USVRECLN DS H

LENGTH, IN BYTES, OF THE RECORD. IF YOU CHANGE THE RECORD'S DATA LENGTH, YOU MUST UPDATE THIS FIELD ACCORDINGLY.

USVRMAXL DS H

ACTUAL LENGTH IF FIXED LENGTH RECORD OR MAXIMUM LENGTH IF VARIABLE LENGTH RECORD, IN BYTES.

USVRDBK DS F

DATABASE KEY OF RECORD

USVRLOPG DS F

NUMBER OF LOWEST PAGE OF RECORD'S RANGE

USVHIPG DS F

NUMBER OF HIGHEST PAGE OF RECORD'S RANGE

USVALOOPG DS F

NUMBER OF LOWEST PAGE IN AREA

USVAHIPG DS F

NUMBER OF HIGHEST PAGE IN AREA

USVPARM2L EQU *-USVPARM2

LENGTH OF PARAMETER #2

**

RECORD OCCURRENCE BLOCK: THIS IS THE THIRD BLOCK OF INFORMATION PASSED TO THE USER EXIT. IT CONTAINS THE ACTUAL RECORD OCCURRENCE FOR THE CURRENT TRANSACTION RECORD. TOTAL LENGTH IS DEFINED IN THE RECORD TYPE'S SUBSCHEMA DESCRIPTION.

ITEM USAGE LENGTH DESCRIPTION

RECORD AS DEFINED AS DEFINED ACTUAL CURRENT OF OCCURRENCE IN SUBSCHEMA IN SUBSCHEMA TRANSACTION RECORD.

**

USVPARM3 DSECT

PARAMETER #3 - RECORD OCCURRENCE

USVRDATA DS 0X

ACTUAL DATABASE RECORD OCCURRENCE LENGTH AS DEFINED IN THE SUBSCHEMA.
Syntax Notation

<table>
<thead>
<tr>
<th>Example</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>Process=</td>
<td>Keywords appear in mixed case.</td>
</tr>
<tr>
<td></td>
<td>The minimum required portion of each keyword appears in uppercase.</td>
</tr>
<tr>
<td>JCLMBR=jcl-member-name</td>
<td>Variables appear in lowercase. You substitute an appropriate value</td>
</tr>
<tr>
<td></td>
<td>for each variable.</td>
</tr>
<tr>
<td>[PASSWORD=dictionary-password]</td>
<td>Brackets indicate optional clauses or commands.</td>
</tr>
<tr>
<td>/ transient-subschema-name \</td>
<td>Braces enclose two or more options. You must select one of them.</td>
</tr>
<tr>
<td>TRANSSub= &lt; target-subschema-name &gt;</td>
<td></td>
</tr>
</tbody>
</table>

Extract File

Information extracted from the source database by the Database Extract Component of CA IDMS Extractor is written to the Extract File. The Extract File is a variable-blocked file with an LRECL of 32756 and a BLKSIZE of 32760.

⚠️ **Note:** For more information about the size of this file, see Allocating Space for DASD Files (see page 238).

Workfile

The Workfile is a temporary file used by the Database Load Component of CA IDMS Extractor. Information is saved on the Workfile between sort exits. It is a variable-blocked file.

Communication File

The Communication File (COMMFILE) contains a single record which is used to pass information between the CA IDMS Extractor Batch Components.
Syntax File

Appropriate CA IDMS DDDL syntax is written to the Syntax File by various batch components of CA IDMS Extractor to:

- PUNCH the target database from the target dictionary
- ADD the transient subschema to the target dictionary
- DELETE the transient subschema from the target dictionary.

Allocating Space for DASD Files

CA IDMS Extractor requires space allocation for an Extract File, a Workfile, a Communication File, a Syntax File, and sort work files. The Syntax and Communication Files are fixed at one track apiece. Space allocation for a DASD Extract File is dependent on the number of database records extracted and on the number of set occurrences in which those records participate as either owner or member. The table below displays the type and number of extracts created and written to the Extract File. The work file and total sort work space require the same amount of space as the Extract File.
## Type and Number of Extracts Created

<table>
<thead>
<tr>
<th>Type Length In Bytes</th>
<th># Of Records Written</th>
</tr>
</thead>
<tbody>
<tr>
<td>h 256</td>
<td>One (1)</td>
</tr>
<tr>
<td>e a d e r</td>
<td></td>
</tr>
<tr>
<td>t 80</td>
<td>One (1)</td>
</tr>
<tr>
<td>r a i l e r</td>
<td></td>
</tr>
<tr>
<td>s 36</td>
<td>One (1) for each selected record type in the source subschema</td>
</tr>
<tr>
<td>s s r e c</td>
<td></td>
</tr>
<tr>
<td>s 36</td>
<td>One (1) for each selected set type in the source subschema</td>
</tr>
<tr>
<td>s s s e t</td>
<td></td>
</tr>
<tr>
<td>d 28 + data b length of selected database record</td>
<td>One (1) for each selected:</td>
</tr>
<tr>
<td>i 20</td>
<td>One (1) or more for each index to which a selected record is connected. These are written for records retrieved through an area sweep, index, or set walk. If a selected record is both an entry record (or indexed through an entry index) and a selected owner or member of a selected set occurrence, multiple index extracts are written.</td>
</tr>
</tbody>
</table>

Review your Selection Criteria Specification to estimate the number of database records that will be extracted.

The Database Load Component creates a Workfile from the Extract File. You must allocate the same amount of space for the Workfile as the Extract File. The Workfile is also variable-blocked. The Database Load Component calculates the LRECL of the file for you with a BLKSIZE of 6144. You can override this block size.

Provide enough SORT SPACE capacity for the entire Workfile.
PROCESS Parameter Syntax Summary

Process / Specname=selection-criteria-specification-name, UserID=userid \ < Jclmbr=JCL-member-name, UserID=userid > \ Unload / |
|, Signonid=dictionary-userid[, Password=dictionary-password]] |
|, TRANSSub= < transient-subschema-name > |
| \ target-subschema-name / |
[, TRANSdmcl=target-DMCL-name]
[, SRCSub=source-subschema-name]
[, SRCdmcl=source-DMCL-name]
[, SRCdictName=source-dictionary-name]
[, SRCdictNode=source-dictionary-node-name]
[, SRCdbname=source-database-name]
[, SRCdbname=source-database-node-name]
[, USERExit=user-exit-module-name]
[, NOCompare]