CA IDMS - 19.0
Using DML Online

Date: 15-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

**CA IDMS DMLO Features** ................................................................. 10
- Multiple/Distributed Database Support ........................................ 11
- Multiple Teleprocessing Environment Support ........................ 11
- Support of CA IDMS DML Commands ........................................ 11
- Support of CA IDMS LRF Commands .......................................... 11
- Support of CA IDMS/DC SCRATCH/QUEUE Commands .......... 12
- Access to Dictionary Records .................................................... 12
- Extensive Set of CA IDMS DMLO Commands ............................ 12
- Session PROFILES ................................................................... 12
- CLIST Programming .................................................................. 13
- Comprehensive Online Documentation ................................. 13
- Data Manipulation and Display .................................................. 14
- PF Key Usage .......................................................................... 14
- EQUATE Facility ....................................................................... 14
- Selection Lists .......................................................................... 14
- Macros and Variables ............................................................... 15
- Access to CA IDMS Statistics ................................................... 15
- Extensive DBKEY Capabilities .................................................. 15
- Menu/Assist Mode .................................................................... 15
- Access Security ......................................................................... 16
- Installation Customization ....................................................... 16
- Mixed Page Group Support ...................................................... 17
- Enter Key Usage ......................................................................... 17
- User Exit Module ...................................................................... 17
- How Can CA IDMS DMLO Improve Productivity? .................... 18
- Is CA IDMS DMLO Easy to Use? ................................................ 18

**CA IDMS DMLO Session** ............................................................. 18
- Text Split Command ............................................................... 19
- Text Flow Command .............................................................. 19
- Text Entry Command .............................................................. 19
- Destructive Line Shift Examples .............................................. 20
Protective Line Shift Command .............................................................. 20
Introduction to the CA IDMS DMLO Session ........................................ 21
The CA IDMS DMLO Session ................................................................. 22
  Invoking CA IDMS DMLO ................................................................. 23
  PROFILE Selection/Maintenance ....................................................... 25
  Subschema Selection ........................................................................ 26
CA IDMS DMLO Session Mainline Screens ........................................... 26
  Record Display Screen .................................................................... 27
  Area Display Screen ....................................................................... 27
  Data Manipulation Screen ............................................................... 28
  Options Screen .............................................................................. 29
  PF Key Display Screen ................................................................... 30
  EQUATE Display Screen ................................................................ 31
  Database Key Display Screen .......................................................... 31
  Sets Display Screen ....................................................................... 32
  Scratch Display Screen .................................................................. 32
  Menu/Assist Mode Screen ............................................................... 33
  Expanded Command (LRF) Screen .................................................. 34
  Interrupt Screen ............................................................................ 35
CA IDMS DMLO Session Termination .................................................... 35
  PROFILE Save Screen ..................................................................... 36
Using the CA IDMS DMLO CLIST Processor ........................................... 36
  What is a CLIST? ............................................................................ 37
  CLIST Creation .............................................................................. 37
  CLIST Selection/Maintenance ......................................................... 38
  CLIST Execution .......................................................................... 38
  CLIST Execution Modes .................................................................. 39
  CLIST Commands ......................................................................... 40
  CLIST Examples ........................................................................... 41

Commands 2 .......................................................................................... 43
CA IDMS DML Commands and Restrictions .......................................... 44
CA IDMS DML Control Statements ......................................................... 44
CA IDMS DML Retrieval Statements ....................................................... 45
  Using FIND/OBTAIN Commands .................................................. 46
  FIND/OBTAIN Special Considerations .......................................... 48
CA IDMS DML Modification Statements ................................................ 48
CA IDMS DML Save Statements ............................................................ 49
  ACCEPT/RETURN Special Considerations ...................................... 49
LRF Commands .................................................................................... 50
Primary Commands ........................................................................................................ 96
Line Commands .................................................................................................................. 96
Program Function Keys ....................................................................................................... 97
Entering Commands ................................................................................................................ 97
Scroll Options ..................................................................................................................... 98
Primary Commands 4 ........................................................................................................... 98
BOTTOM Command ........................................................................................................... 99
BOUNDS Command .......................................................................................................... 99
CANCEL Command ............................................................................................................ 100
CAPS Command .................................................................................................................. 100
CHANGE Command ............................................................................................................ 100
COPY Command .................................................................................................................. 101
CURSOR Command ............................................................................................................. 101
DOWN Command ............................................................................................................... 102
EDITOR-ID Command ......................................................................................................... 102
ECHO Command ................................................................................................................... 102
END Command .................................................................................................................... 103
ENTER Command ................................................................................................................. 103
EXCLUDE Command ............................................................................................................ 103
FIND Command .................................................................................................................... 103
FIRST Command .................................................................................................................. 104
KEYS Command ................................................................................................................. 104
LAST Command ................................................................................................................... 104
LEFT Command ................................................................................................................... 105
LOCATE Command .............................................................................................................. 105
MEMORY Command .......................................................................................................... 105
NULLS Command ............................................................................................................... 106
PROFILE Command .......................................................................................................... 106
RCHANGE Command ......................................................................................................... 106
RESET Command ................................................................................................................. 107
RESHOW Command ............................................................................................................ 108
RFIND Command ............................................................................................................... 108
RIGHT Command ............................................................................................................... 108
SAVE Command ................................................................................................................. 109
TABB Command ................................................................................................................... 109
TABF Command .................................................................................................................... 109
TABS Command ................................................................................................................. 109
TIME Command .................................................................................................................. 110
TOP Command ................................................................................................................... 110
UP Command ...................................................................................................................... 110
Line Commands 4 ............................................................................................................... 111
Entering Line Commands .................................................................................................. 111
Using DML Online

This section provides the information needed to navigate smoothly through a database using CA IDMS DMLO. In addition, the many features that CA IDMS DMLO offers are documented to assist you.

CA IDMS DMLO is an interactive CA IDMS productivity tool which allows ad hoc navigation, retrieval, and update of CA IDMS/DB. CA IDMS DMLO is a full-screen editor for database records that is as easy to use as a text editor. For database navigation, CA IDMS DMLO uses an interactive data manipulation language compatible with CA IDMS DMLC. CA IDMS DMLO accesses dictionary information to provide COBOL-like structured record displays with element values, and to allow updating the database with changed values. Because CA IDMS DMLO uses CA IDMS DMLC commands, it is an excellent learning tool for database navigation techniques. Commands are executed interactively, and results are immediately available.

CA IDMS DMLO also provides an extensive set of control commands which allow you to manage the online session. CA IDMS/DMLO includes security and audit capabilities for protecting the database from unauthorized access.

For more information, see the following topics:
- CA IDMS DMLO Features (see page 10)
- CA IDMS DMLO Session (see page 18)
- Commands 2 (see page 43)
- Operation (see page 92)
- CLIST Editing Commands (see page 95)
- Batch CLIST Processing (see page 115)

CA IDMS DMLO Features

You can perform the following IDMS DMLO tasks:
- Multiple/Distributed Database Support (see page 11)
- Multiple Teleprocessing Environment Support (see page 11)
- Support of CA IDMS DML Commands (see page 11)
- Support of CA IDMS LRF Commands (see page 11)
- Support of CA IDMS/DC SCRATCH/QUEUE Commands (see page 12)
- Access to Dictionary Records (see page 12)
- Extensive Set of CA IDMS DMLO Commands (see page 12)
- Session PROFILES (see page 12)
- CLIST Programming (see page 13)
- Comprehensive Online Documentation (see page 13)
- Data Manipulation and Display (see page 14)
- PF Key Usage (see page 14)
- EQUATE Facility (see page 14)
- Selection Lists (see page 14)
Multiple/Distributed Database Support

CA IDMS DMLO fully supports multiple and distributed database and dictionary configurations described by the CA IDMS DBNAME/DBNODE and DICTNAME/DICTNODE parameters.

Multiple Teleprocessing Environment Support

CA IDMS DMLO operates in the CA IDMS/DC, CICS/VS, TSO, and VM/ESA teleprocessing environments. Some functions may not be available in all environments.

Support of CA IDMS DML Commands

CA IDMS DMLO supports all CA IDMS Data Manipulation Language commands associated with database navigation. These commands are implemented in their COBOL syntax (CA IDMS DMLC).

Support of CA IDMS LRF Commands

CA IDMS DMLO supports access to both LR and mixed subschemas. All LRF commands are available. Support for the WHERE clause supports complex arithmetic and boolean expressions.

Because an LRF command can be quite lengthy, CA IDMS DMLO provides for a much expanded command area which can be requested by an option of the CA IDMS DMLO SET command.

CA IDMS DMLO provides an excellent vehicle for the database administrator to develop, debug, and evaluate LRF path definitions.
Support of CA IDMS/DC SCRATCH/QUEUE Commands

CA IDMS DMLO provides a complete implementation of CA IDMS/DC scratch/queue processing, including all standard CA IDMS DMLC syntax, as well as powerful enhancements.

Access to Dictionary Records

CA IDMS DMLO allows you to access record descriptions stored in the dictionary which are not part of the subschema being processed. These records may be used as work areas or scratch/queue data areas.

Extensive Set of CA IDMS DMLO Commands

CA IDMS DMLO provides an extensive set of commands which allow you to optimize your online session. These commands supplement the CA IDMS/DMLC commands and allow you to manipulate data, control the data display, define abbreviations, use PF keys, and analyze the environment.

Session PROFILES

At session termination CA IDMS DMLO provides an opportunity to preserve user-defined characteristics of a CA IDMS DMLO session, allowing convenient restoration of those parameters when another session is started. This set of session parameters is called a PROFILE and includes specifications of the following:

- Session Control Parameters
  - Subschema, schema, version
  - DBNAME, DBNODE, DICTNAME, DICTNODE
  - Display and processing options
  - User exit specifications
- READY modes for areas
- Abbreviations and Macro Commands (EQUATE settings)
- PF Key Assignments
Multiple PROFILEs are allowed for each user. The concept of "global" PROFILEs is also supported; i.e., the system administrator may define generically useful PROFILEs which are available to all users. For efficiency, these PROFILEs are stored in a small database structure established at product installation.

At session signon you may also request a selection list of available PROFILEs. This same screen provides PROFILE maintenance capabilities, allowing you to COPY, RENAME, and KILL (delete) PROFILEs.

**CLIST Programming**

The CA IDMS DMLO CLIST facility provides a means for creating and executing sequences of CA IDMS /DML and CA IDMS DMLO commands. Up to nine variable arguments (%1-%9) may appear within the CLIST, and provide a means for defining general purpose routines. The CLIST processor includes structured constructs for IF-ELSE, looping, data manipulation, nesting of CLIST calls, and a prompt facility for arguments. CA IDMS DMLO includes an SPF-like text editor for entry and updates to the CLIST programs.

While the CA IDMS DMLO CLIST processor is not intended to be a full-featured interpretive programming language, it does offer adequate facilities to prepare utility routines of some complexity.

THE CLIST facility also provides for "global" CLISTs which may be created and maintained by the CA IDMS DMLO administrator but accessed by all users.

At any time during the session you may request a selection list of available CLISTs. You may also request COPY, RENAME, and KILL (delete) maintenance functions.

**Comprehensive Online Documentation**

At any time during the CA IDMS DMLO session, you may press an installation-defined PF key to access CA IDMS DMLO online documentation. The online documentation is intended to provide extensive information at the terminal when you need it.

Online documentation includes:

- CA IDMS DMLO screen information
- CA IDMS DML command restrictions which apply in CA IDMS DMLO
- Comprehensive section to CA IDMS DMLO commands
- Online message facility.
Data Manipulation and Display

CA IDMS DMLO uses the dictionary to obtain record and element definitions necessary to allow a COBOL-format structured display of database and work records. CA IDMS DMLO allows you to select the format for display and entry of data into these fields. Possible formats are NATIVE (dictionary definition), HEX, and DUMP (simultaneous character and hex).

CA IDMS DMLO performs extensive data editing of data content and entry, and will detect and indicate invalid or non-displayable data. You may select AUTOHEX display mode, in which CA IDMS DMLO will automatically alter the display format to HEX for fields containing invalid data.

PF Key Usage

CA IDMS DMLO allows you to associate frequently used commands with PF keys, thus reducing your keystrokes. You may assign commands to PF keys in two ways. First, you may enter an executable command and press a PF key rather than the ENTER key. This assigns the command to the PF key. Second, you may enter the "KEYS" command. This will display all current PF key settings, and allow updates to multiple keys, by simply entering the desired command opposite the PF key number. An additional capability of PF key processing is the ability to merge command line input with the command string represented by the PF key.

Any PF key assignments you make can be saved in a PROFILE.

EQUATE Facility

Another time-saving facility of CA IDMS DMLO is the availability of a logical set of predefined abbreviations for CA IDMS DML keywords, and the ability for you to define additional abbreviations (EQUATEs) for frequently used subschema entities and command phrases. These phrases may be partial or prototype macro commands, and may include previously equated words or phrases. Any EQUATES you define can be saved in a PROFILE.

Selection Lists

CA IDMS DMLO provides "pop-up" selection lists for subschema entities. Such lists can eliminate the need for the user to recall the names of all records, sets, and areas in a given subschema, or active scratch ID's. These selection lists are invoked by the appearance of ?R, ?S, ?A or ?T within, or instead of, a command.
Macros and Variables

CA IDMS DMLO provides for the use of variable symbols to construct executable CA IDMS DML commands. These symbols are of two types. First, there are the symbols representing active DB/DC entities &D, &R, &S, &A, &T, &Q. Second, there are the macro variables, &1 - &9, @1 - @9, which represent positional tokens from the command line to be merged with a prototype command.

For a complete description of the capabilities, of CA IDMS DMLO variable processing, see CA IDMS DMLO Session (see page 18).

Access to CA IDMS Statistics

CA IDMS DMLO allows you to evaluate the efficiency of database access strategies by providing access to CA IDMS statistics. These statistics are always available as cumulative values since the start of the session. In addition, you may "reset" the values to allow determination of incremental statistics associated with execution of a smaller number of commands. Effectively, CA IDMS DMLO calculates the difference in cumulative statistics between two points that you specify.

Extensive DBKEY Capabilities

CA IDMS DMLO provides extensive facilities for manipulation and display of database keys. Whenever a CA IDMS DMLC command requires a database key, you may use page-line, hexadecimal, or decimal literals. In addition, CA IDMS DMLO provides ten KEYPADS which may serve as source/target fields for commands. The SHOW KEYPADS command will display all three formats of these KEYPADS. Thus, you can use CA IDMS DMLO as a modest hex calculator.

In addition to providing data-base keys, page-information values may be supplied on the FIND /OBTAIN DML statement. CA IDMS DMLO provides ten page-info KEYPADS PGR0 - PGR9. The SHOW KEYPADS command will display both DBKEY and PAGE-INFO values.

Menu/Assist Mode

The Menu/Assist Mode of CA IDMS DMLO can provide the less experienced CA IDMS user with a more structured environment which requires far less detailed knowledge of CA IDMS DML syntax. This mode can also assist in learning the full syntax while being productive with CA IDMS/DMLO.

CA IDMS DMLO provides the following features in Menu/Assist Mode:

- Structured, simplified presentation and entry of CA IDMS DMLC commands
- Instructional echo of standard syntax
- Installation-defined menu/assist display format
Access Security

Three levels of security are available to CA IDMS DMLO users. You can select the degree of database protection that is needed at your installation.

- **Level One Security** -- You may choose to bypass the CA IDMS DMLO security feature. If you decide not to use security, CA IDMS DMLO will verify only that the subschema name entered at signon is valid and usable.

- **Level Two Security** -- Security is achieved through user ID and password entered at signon. CA IDMS DMLO allows you three chances to enter your correct password and ID.

- **Level Three Security** -- At this security level CA IDMS DMLO verifies both password and ID (Level Two) and also makes certain that the user is authorized to access the subschema requested.

**Note:** Any access restrictions imposed by Level Three security will be reflected in the list of subschemas presented if a subschema selection list is requested at signon.

You can augment any CA IDMS DMLO level by also restricting access by ready mode. You can change the standard ready modes CA IDMS DMLO will process at a global level, and at the user ID level. For more information, see the appropriate CA IDMS Installation and Maintenance Section.

Installation Customization

A customization program USDTPARM distributed with CA IDMS DMLO allows the specification of CA IDMS DMLO parameters at installation. Note that some of these parameters are changeable by each user during a CA IDMS DMLO session (and those changes preserved in a PROFILE); while others are permanent in the sense that USDTPARM must be changed.

USDTPARM allows specification of the following session defaults:

- System administrator user IDs
- Signon screen options and default values
- SET command options
- PF key assignments
- Standard abbreviations
Displayable characters -- those which will not trigger an 'INVALID DATA' indication

User exit specifications and messages

Menu/Assist display format

Permanently excluded subschemas, independently of normal CA IDMS DMLO security.

For more information, see the CA IDMS Installation and Maintenance Section -- z/OS.

Mixed Page Group Support

CA IDMS DMLO allows a single CA IDMS rununit to access data from different page groups using the mixed page group support feature.

This feature also permits a single rununit to access a database with segments that have been defined with a different maximum for the number of records that can be stored on a page. You can specify a different maximum for each page group.

You can define up to 32,767 different page groups. A single page group can qualify up to 16 million pages which significantly increases the size of a database that can be accessed from a single rununit. Furthermore, by allowing portions of a database to have different maximum records per page, you can use the available pages more effectively.

Exit 34 is provided for use with the mixed page group support feature. You can use this exit to help identify and correct applications that may require modification to function correctly when the mixed page group support feature is enabled. For more information on user exits, see the CA IDMS System Operations Section.

Enter Key Usage

CA IDMS DMLO allows you to change the default setting for the Enter key. By changing the default, the last command on the DMLO command line is re-executed. This feature is useful, for example, when navigating a database to repeat FIND/OBTAIN NEXT/PRIOR records without having to overtype data on the command line.

User Exit Module

CA IDMS DMLO gives you the ability to customize an assembler user exit module which can be invoked after each DML command is executed. Upon return from the user exit to DMLO a return code value can be set to indicate an error condition and termination or continued processing.
How Can CA IDMS DMLO Improve Productivity?

Uses for CA IDMS DMLO span all aspects of database activity, including application development, database design and analysis, performance evaluation, production support, and CA IDMS training.

CA IDMS DMLO allows you to do the following:

- Create OOAK records
- Create and maintain test database structures
- Check the results of program testing
- Locate and fix data integrity problems in production databases
- Validate navigation logic before coding
- Test LRF path definitions
- Analyze the I/O efficiency of access strategies.

It is even possible to perform complicated mass changes to the database—all without special programs.

Is CA IDMS DMLO Easy to Use?

CA IDMS DMLO allows you to read and update database records without learning a new access language, and without writing special programs. Further, the data is presented in a familiar structured format.

Although CA IDMS DMLO supports standard CA IDMS DML syntax, it has a broad range of features and extensions which allow you to perform the above functions using shorthand notation, PF keys, macro commands, and repeatable series of commands.

CA IDMS DMLO is a powerful interactive query/update tool for the experienced CA IDMS technician.

CA IDMS DMLO also provides facilities by which the less experienced CA IDMS programmer can gain familiarity with database techniques and CA IDMS DML syntax in a more structured environment.

CA IDMS DMLO Session

This section is a step-by-step section that shows you how to use CA IDMS DMLO. It includes a conceptual view of the CA IDMS DMLO session, gives information on initiating a session and executing commands, and describes how to interrupt and terminate a session. More detailed CA IDMS DMLO screen information is available through the online documentation.
Text Split Command

\[ \text{TS number-of-lines} \]

where:

number-of-lines specifies the number of lines to be inserted between the split line. The default is 1.

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.

See the online documentation for examples of the TS line command.

Text Flow Command

\[ \text{TF rb} \]

where:

rb specifies the right bound for the text.

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.

See the online documentation for examples of Text Flow parameters.

Text Entry Command

\[ \text{TE number-of-lines} \]
where:

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

The TE line command formats the screen with an unnumbered open text entry area which 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.

### Destructive Line Shift Examples

**number-of-columns**

where:

**number-of-columns** specifies the number of lines to shift. The default is 1.

- **([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.
  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 )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.

### Protective Line Shift Command

**number-of-columns**

where:
number-of-columns

Specifies the columns to shift. The default is 1.

- \(<\text{number-of-columns}\>\text{number-of-columns}\)\text{number-of-columns}\). . .\text{number-of-columns}\). . .>>
  
  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.

- \(<\text{number-of-columns}\>

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

- \(<\text{number-of-columns}\>

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

- \(<\text{n}\>\text{n}\)

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

- \(<\text{n}\>)\text{nn}

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

- \(<\text{n}\>\text{nn}\)

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

### Introduction to the CA IDMS DMLO Session

A typical CA IDMS DMLO session resembles the logic of a CA IDMS batch program. The comparison shown below illustrates the similarities:

<table>
<thead>
<tr>
<th>CA IDMS DMLO Session Component</th>
<th>Batch Program Component</th>
</tr>
</thead>
<tbody>
<tr>
<td>SIGNON</td>
<td>BIND subschema</td>
</tr>
<tr>
<td>* BIND records</td>
<td>BIND records</td>
</tr>
<tr>
<td>* READY areas</td>
<td>READY areas</td>
</tr>
<tr>
<td>CA IDMS DML and CA IDMS DMLO commands</td>
<td>Body of program</td>
</tr>
<tr>
<td>FINISH/ROLLBACK</td>
<td>FINISH/ROLLBACK</td>
</tr>
<tr>
<td>* PROFILE save</td>
<td>STOP RUN</td>
</tr>
</tbody>
</table>

*Those components marked with "*" are optional in a CA IDMS DMLO session. The functions are still essential parts of a CA IDMS DMLO session, but capabilities allow for automatic processing of these functions.*
The CA IDMS DMLO Session

The detailed steps you will follow during a CA IDMS DMLO session will not follow a fully structured sequence. This is because CA IDMS DMLO allows you to determine not only what functions are appropriate for your processing needs, but also the sequence in which they are applied. However, the following list summarizes required and optional steps to be followed:

1. Invoke CA IDMS DMLO by entering the task code assigned at installation.

2. If quick-in startup is not used, complete the Signon screen.
   - From the Signon screen, you can request the Subschema List screen
   - From the Subschema List screen, you can select a subschema
   - From the Signon screen, you can request the PROFILE List screen
   - From the PROFILE Selection/Maintenance screen, you can perform PROFILE maintenance
   - From the PROFILE Selection/Maintenance screen, you can select a PROFILE

3. After startup is complete, CA IDMS DMLO may present the Record Display screen.
   - You may BIND records at this point, or defer the BIND

4. If the Record Display Screen is not presented, CA IDMS DMLO presents the Area Display screen showing READY mode of each area.
   - You may READY areas at this point, or defer the READY

5. From this point on, all screen displays are determined by your actions.

6. You may issue any CA IDMS DML command. These normally result in the display of the Data Manipulation screen.
   - You can alter the data content of that record.
   - You can issue CA IDMS DML modification commands for that record.

7. You may issue any CA IDMS DMLO command. Many of these commands result in a particular screen being displayed. Details are described below in the discussions of each screen format.

8. When you have completed all database activities, you may issue a ROLLBACK or FINISH command.
   - If you specify the SAVE or NOSAVE option, that option will be applied to the specified PROFILE, and the session will be terminated.
   - If neither SAVE nor NOSAVE is specified, the PROFILE Save screen is displayed.
9. You may specify which, if any, components of a PROFILE you wish to save, and the PROFILE name to be created or updated.

- The session will be terminated after performing the requested PROFILE function.

- Invoking CA IDMS DMLO (see page 23)
- PROFILE Selection/Maintenance (see page 25)
- Subschema Selection (see page 26)

**Invoking CA IDMS DMLO**

There are two ways to begin a CA IDMS DMLO session. Standard session startup requires that you complete the Signon Screen before entering the session mainline. Quick-in session startup allows you to specify a PROFILE name at the time the CA IDMS DMLO task is invoked, and to bypass the Signon screen.

**Standard Session Startup**

The standard method of session startup consists of the following steps:

1. Enter the CA IDMS DMLO task code. This results in display of the Signon screen.

2. Enter required information on the Signon screen.

3. If you enter an unknown PROFILE name or press the indicated PF key CA IDMS DMLO will display the PROFILE Selection/Maintenance screen. (See the following example).

4. If you press the indicated PF key, you can request online documentation for the Signon details.

5. If you enter a "?" in the subschema or schema field, CA IDMS DMLO displays a list of subschemas which you are authorized to access.

6. When you have provided sufficient Signon screen information CA IDMS DMLO will complete the session startup and proceed to the next screen.

7. If the AUTOBIND is not in effect and you did not specify a ready mode on the Signon screen, CA IDMS DMLO will present the Record Display screen.
   - You may BIND records in the list.
   - You may enter the BIND command now, or at any time during the session.
   - You may SET AUTOBIND ON to eliminate the necessity for doing any explicit BIND.

8. If AUTOBIND is in effect or you specified a ready mode on the Signon screen, CA IDMS DMLO will present the Area Display screen. This list will show the current ready mode, if any, of each area in the current subschema.
   - You may enter a ready mode next to any area(s) in the list.
9. You may now enter any CA IDMS DML or CA IDMS DMLO commands.

You may issue an explicit CA IDMS DML READY command.

You may defer any further READY activity to a later point in the session.

Required Signon Parameters

The CA IDMS DMLO online documentation provides a detailed explanation of the contents of the Signon screen. However, it is important to note that you must provide a subschema name in order to begin a CA IDMS DMLO session. You may provide the subschema explicitly by entry on the Signon screen or selection from a list, or implicitly by requesting a PROFILE.

Entry of a schema name and version on the Signon screen is only necessary if the subschema name entered is not unique in the dictionary specified by DICTNAME/DICTNODE.

Whether you specify PROFILE or subschema on the Signon screen, you may specify READY mode.

Note: When you are communicating across CV's, you must specify DICTNAME and DICTNODE on the DMLO Signon Screen.

Quick-In Session Startup (CA IDMS/DC only)

If you know the name of a PROFILE you wish to use, you may bypass the Signon screen as follows:

1. Enter the following command at "ENTER NEXT TASK CODE" :

   DML0 profile-name ready-mode

   Where:

   profile-name
   The PROFILE to be used for this CA IDMS DMLO session

   ready-mode
   Abbreviation for area READY specification (optional):
1. U (shared Update)

2. R (shared Retrieval)

3. P (as previously stored in the PROFILE)

2. CA IDMS DMLO loads the specified PROFILE, readies all areas as indicated, and displays the Area Display screen.

   - You may enter a ready mode next to any area(s) in the list.
   - You may issue an explicit CA IDMS DML READY command.
   - You may defer any further READY activity to a later point in the session.

3. You may now enter any CA IDMS DML or CA IDMS DMLO commands.

If a profile name is unknown, the Profile Selection screen will be displayed. Thus, you may bypass the Signon screen and directly enter the Profile Selection screen by typing "DMLO ?" at the "ENTER NEXT TASK CODE" prompt.

### PROFILE Selection/Maintenance

The PROFILE Selection/Maintenance screen is displayed if you press the indicated PF key from the Signon screen. It will also be displayed if you request an unknown PROFILE name (in either standard or quick-in session startup).

You may take the following actions from this screen:

- You may select a PROFILE for use in this session by placing an "S" next to the item of interest.
- You may delete any non-global PROFILES (i.e., owner not = "SYS") by placing "K" next to the item(s) to be deleted.
- You may enter any PROFILE maintenance commands.
- You may enter the "LIST" command to refresh the list of PROFILES being displayed. Execution of maintenance commands will not result in an automatic refresh.

For more information on the commands that may be used at this screen, see PROFILE Maintenance Commands (see page 70).

<table>
<thead>
<tr>
<th>ACTION</th>
<th>PROFILE</th>
<th>SRC</th>
<th>SUBSCH</th>
<th>SCHEMA</th>
<th>DBNAME</th>
<th>DICTNAME</th>
<th>USED</th>
<th>UPDT</th>
</tr>
</thead>
<tbody>
<tr>
<td>_</td>
<td>EMPDEMO</td>
<td>EMPSS01</td>
<td>EMPSCHM</td>
<td>EMPDEMO</td>
<td>mmddyy</td>
<td>mmddyy</td>
<td></td>
<td></td>
</tr>
<tr>
<td>_</td>
<td>EMPPROF</td>
<td>EMPSS01</td>
<td>EMPSCHM</td>
<td></td>
<td>mmddyy</td>
<td>mmddyy</td>
<td></td>
<td></td>
</tr>
<tr>
<td>_</td>
<td>RALPH</td>
<td>IDMSNWKA</td>
<td>IDMSNTWK</td>
<td></td>
<td>mmddyy</td>
<td>mmddyy</td>
<td></td>
<td></td>
</tr>
<tr>
<td>_</td>
<td>TESTINVC</td>
<td>EMPS01</td>
<td>EMPSCHM</td>
<td></td>
<td>mmddyy</td>
<td>mmddyy</td>
<td></td>
<td></td>
</tr>
<tr>
<td>_</td>
<td>X</td>
<td>EMPS01</td>
<td>EMPSCHM</td>
<td></td>
<td>mmddyy</td>
<td>mmddyy</td>
<td></td>
<td></td>
</tr>
<tr>
<td>_</td>
<td>IDMSNWKA</td>
<td>SYS</td>
<td>IDMSNWKA</td>
<td>IDMSNTWK</td>
<td></td>
<td>mmddyy</td>
<td>mmddyy</td>
<td></td>
</tr>
</tbody>
</table>

DML/0 Rnn.nn ============================================================== CA, INC.
Subschema Selection

The Subschema Selection screen is displayed if you enter at least one "?" in either the subschema or schema field of the Signon screen. The list presented contains only subschemas for which you have access. The list is further restricted by matching subschema and schema names to the "mask" patterns entered as subschema and/or schema name.

Mask patterns are processed as follows:

1. "?", "*" (asterisk), or " " (space) in the pattern will match any character in the name.
2. Any other character in the pattern must exactly match the corresponding character in the name.

For example, "???SSRT " will match "IDMSSRT " and "ABCSSRTX".

The list is sorted by subschema name if "?" appears anywhere in the subschema pattern, otherwise the list is sorted by schema.

You can enter "S" next to an item to use that subschema for this CA IDMS DMLO session.

<table>
<thead>
<tr>
<th>ACTION</th>
<th>SCHEMA</th>
<th>VERSION</th>
<th>SUBSCHEMA</th>
<th>ERROR</th>
</tr>
</thead>
<tbody>
<tr>
<td>-</td>
<td>EMPSCHM</td>
<td>0100</td>
<td>EMPSSLR1</td>
<td></td>
</tr>
<tr>
<td>-</td>
<td>EMPSCHM</td>
<td>0100</td>
<td>EMPSS01</td>
<td></td>
</tr>
<tr>
<td>-</td>
<td>ESSSCHM</td>
<td>0001</td>
<td>ESSSUB00</td>
<td></td>
</tr>
<tr>
<td>-</td>
<td>IDMSMTWK</td>
<td>0001</td>
<td>IDMSNWKA</td>
<td></td>
</tr>
<tr>
<td>-</td>
<td>IDMSSECQ</td>
<td>0001</td>
<td>IDMSSECQ</td>
<td></td>
</tr>
<tr>
<td>-</td>
<td>IDMSSECQ</td>
<td>0001</td>
<td>IDMSSECQ</td>
<td></td>
</tr>
<tr>
<td>-</td>
<td>IDMSSECQ</td>
<td>0001</td>
<td>IDMSSECQ</td>
<td></td>
</tr>
<tr>
<td>-</td>
<td>SSKSCHEM</td>
<td>0001</td>
<td>SSKSUB00</td>
<td></td>
</tr>
<tr>
<td>-</td>
<td>USDSCHM</td>
<td>0001</td>
<td>USDSUB00</td>
<td></td>
</tr>
<tr>
<td>-</td>
<td>USDSCHM</td>
<td>0001</td>
<td>USDSUB02</td>
<td></td>
</tr>
</tbody>
</table>

CA IDMS DMLO Session Mainline Screens

Descriptions of the following screens encountered in a CA IDMS DMLO session mainline are as follows:

- Record Display Screen (see page 27)
- Area Display Screen (see page 27)
Record Display Screen

The Record Display screen is displayed:

- After signon if AUTOBIND is not in effect and you did not specify a ready mode.

- If you enter a simple BIND command.

- If you enter the SHOW RECORDS command.

- The selection variable ?R appears as part of, or instead of, a command. Possible actions are indicated by the informational message.

For information on selection variable processing, see CA IDMS DMLO Session (see page 18).

<table>
<thead>
<tr>
<th>ACTION</th>
<th>RECORD</th>
<th>AREA</th>
<th>BND</th>
</tr>
</thead>
<tbody>
<tr>
<td>001</td>
<td>COVERAGE</td>
<td>INS-DEMO-REGION</td>
<td></td>
</tr>
<tr>
<td>002</td>
<td>DENTAL-CLAIM</td>
<td>INS-DEMO-REGION</td>
<td></td>
</tr>
<tr>
<td>003</td>
<td>DEPARTMENT</td>
<td>ORG-DEMO-REGION</td>
<td></td>
</tr>
<tr>
<td>004</td>
<td>EMPLOYEE</td>
<td>EMP-DEMO-REGION</td>
<td></td>
</tr>
<tr>
<td>005</td>
<td>EMPOSITION</td>
<td>EMP-DEMO-REGION</td>
<td></td>
</tr>
<tr>
<td>006</td>
<td>EXPERTISE</td>
<td>EMP-DEMO-REGION</td>
<td></td>
</tr>
<tr>
<td>007</td>
<td>HOSPITAL-CLAIM</td>
<td>INS-DEMO-REGION</td>
<td></td>
</tr>
<tr>
<td>008</td>
<td>INSURANCE-PLAN</td>
<td>INS-DEMO-REGION</td>
<td></td>
</tr>
<tr>
<td>009</td>
<td>JOB</td>
<td>ORG-DEMO-REGION</td>
<td></td>
</tr>
<tr>
<td>010</td>
<td>NON-HOSP-CLAIM</td>
<td>INS-DEMO-REGION</td>
<td></td>
</tr>
<tr>
<td>011</td>
<td>OFFICE</td>
<td>ORG-DEMO-REGION</td>
<td></td>
</tr>
<tr>
<td>012</td>
<td>SKILL</td>
<td>ORG-DEMO-REGION</td>
<td></td>
</tr>
<tr>
<td>013</td>
<td>STRUCTURE</td>
<td>EMP-DEMO-REGION</td>
<td></td>
</tr>
</tbody>
</table>

Area Display Screen

The Area Display screen is displayed:

- After signon if AUTOBIND is in effect or you specified a ready mode.
If you enter the SHOW AREA command.

If the selection variable ?A appears as part of, or instead of, a command.

Possible actions are indicated by the informational message on the screen.

For information on selection variable processing, see CA IDMS DMLO Session (see page 18).

<table>
<thead>
<tr>
<th>ACTION</th>
<th>AREA</th>
<th>MODE</th>
</tr>
</thead>
<tbody>
<tr>
<td>__</td>
<td>001 EMP-DEMO-REGION</td>
<td></td>
</tr>
<tr>
<td>__</td>
<td>002 INS-DEMO-REGION</td>
<td></td>
</tr>
<tr>
<td>__</td>
<td>003 ORG-DEMO-REGION</td>
<td></td>
</tr>
</tbody>
</table>

Data Manipulation Screen

After session startup, you may issue CA IDMS DML commands to access database records or scratch queue records. Any command that results in data transfer will cause the appropriate record buffer to be formatted and displayed, that is, FIND will not alter the display, but OBTAIN and GET will.

You may alter data values on this screen and issue modification commands.

The presentation format of data may be controlled in some detail with CA IDMS DMLO options and commands. For detailed information on the SET and DISPLAY commands, see CA IDMS DMLO Session (see page 18).

mm/dd/yy........................RECORD : EMPLOYEE......................hh:mm:ss
02 EMP-ID-0415..............................N 0120
02 EMP-NAME-0415............................G
03 EMP-FIRST-NAME-0415.....................A MICHAEL
03 EMP-LAST-NAME-0415......................A ANGELO
02 EMP-ADDRESS-0415.......................G
03 EMP-STREET-0415.........................A 507 CISTINE DR
03 EMP-CITY-0415...........................A WELLESLEY
03 EMP-STATE-0415..........................A MA
03 EMP-ZIP-0415............................G
04 EMP-ZIP-FIRST-FIVE-0415...............A 01568
04 EMP-ZIP-LAST-FOUR-0415.................AX 40404040
02 EMP-PHONE-0415..........................N 6178870235
02 STATUS-0415..............................A FFFF
02 SS-NUMBER-0415..........................N 127675593
02 START-DATE-0415.........................G
03 START-YEAR-0415........................N 79
03 START-MONTH-0415........................N 09
03 START-DAY-0415..........................N 08
DML/O Rnn.nn =----------------------------------- CA, INC.
RECORD= EMPLOYEE                      STATUS=0000  DBKEY=0000075083-0001 KEY0=0000000000-0000
DIS EMPLOYEE
SUBSCHEMA=EMPSS01   SCHEMA=EMPSCHM VER=0100  COL 001-080 LINE 0001 OF 0027
mm/dd/yy........................RECORD : EMPLOYEE......................hh:mm:ss
02 EMP-ID-0415..............................NX F0F1F0
02 EMP-NAME-0415............................AX D4C9C3C1C5D3404040C1D5C7C5D36
The Options screen is displayed when you enter the SHOW OPTIONS command. This screen displays information from the Signon screen, as well as a series of options controlled by the SET command. Any or all of these options can be changed at once. Make any desired changes, and press the ENTER key to return to the prior display.

For detailed information on the SET command options, see CA IDMS DMLO Session (see page 18)

The specific forms of the SET command which correspond to the entries on the Options Screen are as follows:

1. SET MENU
2. SET COBDISPLAY
3. SET AUTOHEX
4. SET CMDDISPLAY
5. SET LRF
6. SET AUTOBIND
7. SET MAPIN

Options Screen
8. SET CLIST
9. SET EXIT
10. SET INVCHAR

------------- CA IDMS DML ONLINE RELEASE nn.nn -------------
USER ID ==> FULAL01
PROFILE NAME ==> EMPPROF
READY MODE ==> 
SUBSCHEMA ==> EMPSS01
SCHEMA ==> EMPSCHM
VERSION ==> 0100
DICTNAME ==> 
DICTNODE ==> 
DBNAME ==> 
DBNODE ==> 
PRINT CLASS ==> A
INTERRUPT ==> PA1
LOGICAL DISPLAY WIDTH ==> 080
LOWER CASE ==> N
1. DML/O "MENU" MODE ==> N
2. COBOL DISPLAY FORMAT ==> Y
3. AUTOHEX DATA DISPLAY ==> Y
4. AS USED CMND DISPLAY ==> Y
5. LRF EXPAND CMND AREA ==> N
6. AUTO BIND RECORDS ==> Y
7. FAST DATA INPUT MODE ==> Y
8. FAST CLIST EXEC MODE ==> Y
9. ACTIVE USER EXIT ==> N
10. INVALID CHAR REPLACE ==> _

ALL RIGHTS RESERVED (C) 2007
================ CA, Inc. ================

---

PF Key Display Screen

The PF Key Display screen appears when you enter the SHOW KEYS command. You may view settings of all PF/PA keys by scrolling up/down. You may also enter or change any PF key text assignment by keying a command or phrase next to any PF key label.

⚠️ **Note:** A PF key may also be assigned by pressing that PF key rather than the ENTER key to execute a command. For more information, see [Extended Command Processing](#) (see page 85).
EQUATE Display Screen

The EQUATE Display screen appears when you enter the SHOW EQUATES command. This display is a list of all abbreviations created during this session or loaded as part of a PROFILE. Standard abbreviations are not displayed.

This screen has no provisions for data update. Changes are made only with the EQUATE command. See CA IDMS DMLO Session (see page 18) for detailed information on the EQUATE command and its use in Extended Command Processing.

Database Key Display Screen

The Database Key Display screen appears when you enter the SHOW KEYPADS command. It displays database keys and page-information which are part of the subschema-control-block as well as the ten CA IDMS DMLO database work areas for DBKEYS (KEY0 - KEY9) and PAGE-INFO values (PGR0 - PGR9). Note that three equivalent formats are displayed.

This screen has no provision for data update. Changes to DBKEY and PAGE-INFO values occur as a result of executing CA IDMS DML commands, or in the case of DBKEYS as a result of MOVE commands which specify one of the database key fields as a target.
Sets Display Screen

This screen is displayed if:

- You enter the SHOW SETS command
- You use the selection variable ?S as part of, or instead of, a CA IDMS DML or CA IDMS DMLO command.

If ?S is part of a command, you may select a setname by entering "S" next to the item of interest. Otherwise, the display is informational only.

For more information on selection variables, see CA IDMS DMLO Session (see page 18).

<table>
<thead>
<tr>
<th>ACTION</th>
<th>SET NAME</th>
<th>_OWNER</th>
<th>MEMBERS</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>COVERAGE-CLAIMS</td>
<td>COVERAGE</td>
<td>DENTAL-CLAIM</td>
</tr>
<tr>
<td></td>
<td>DEPT-EMPLOYEE</td>
<td>DEPARTMENT</td>
<td>EMPLOYEE</td>
</tr>
<tr>
<td></td>
<td>EMP-COVERAGE</td>
<td>EMPLOYEE</td>
<td>COVERAGE</td>
</tr>
<tr>
<td></td>
<td>EMP-EMPOSITION</td>
<td>EMPLOYEE</td>
<td>EMPOSITION</td>
</tr>
<tr>
<td></td>
<td>EMP-EXPERTISE</td>
<td>EMPLOYEE</td>
<td>EXPERTISE</td>
</tr>
<tr>
<td></td>
<td>EMP-NAME-NDX</td>
<td>SYSTEM</td>
<td>EMPLOYEE</td>
</tr>
<tr>
<td></td>
<td>JOB-EMPOSITION</td>
<td>JOB</td>
<td>EMPOSITION</td>
</tr>
<tr>
<td></td>
<td>JOB-TITLE-NDX</td>
<td>SYSTEM</td>
<td>JOB</td>
</tr>
<tr>
<td></td>
<td>MANAGES</td>
<td>EMPLOYEE</td>
<td>STRUCTURE</td>
</tr>
<tr>
<td></td>
<td>OFFICE-EMPLOYEE</td>
<td>OFFICE</td>
<td>EMPLOYEE</td>
</tr>
<tr>
<td></td>
<td>REPORTS-TO</td>
<td>EMPLOYEE</td>
<td>STRUCTURE</td>
</tr>
<tr>
<td></td>
<td>SKILL-EXPERTISE</td>
<td>SKILL</td>
<td>EXPERTISE</td>
</tr>
<tr>
<td></td>
<td>SKILL-NAME-NDX</td>
<td>SYSTEM</td>
<td>SKILL</td>
</tr>
</tbody>
</table>

Scratch Display Screen

This screen is displayed if:

- You enter the SHOW SCRATCH command.
- You use the selection variable ?T as part of, or instead of, a CA IDMS DML or CA IDMS DMLO command.

You may delete a SCRATCH area by entering "K" next to the item(s) to be deleted.

If ?T is part of a command, you may select a scratch area by entering "S" next to the item of interest. Otherwise, the display is informational only.
Menu/Assist Mode Screen

The Menu/Assist Mode of CA IDMS DMLO provides a more structured environment which can relieve the less experienced CA IDMS user of needing detailed knowledge of CA IDMS DML syntax.

The initial Menu/Assist Mode screen is displayed when you enter the SET MENU ON command, or when you specify this option on the OPTIONS screen. This screen contains installation-defined instructional messages in the data display area.

After entry of the first command in Menu Mode, the upper portion of the screen becomes a data display and entry area.

You may switch back and forth between Menu/Assist Mode and "Expert" Mode as desired using the SET MENU OFF/ON command or the Options screen (by using the SHOW OPTIONS command).

You have selected the "MENU/ASSIST" mode of DML/O.
Please note the following:
All DML and DML/O commands will continue to be recognized
Shorthand commands are now possible.
Each command consists of a code (formed by concatenating parenthesized letters) and one or subschema entities.
Notation for subschema entities is as follows:
  R = record    S = set    A = area    K = dbkey    F = sortfield
  <R> => record is optional   S/A => either set or area is required
If a name is not known, entering ?R, ?S, ?A will display a selection list.
Any abbreviations entered via EQUATE commands will be honored.
Entity variables &R, &S, &A will be honored.
For example, the DML command OBTAIN FIRST EMPLOYEE WITHIN EMPLOYEE-AREA could be entered as

DML/O Rnn.nn ================= ERROR STATUS : 0000 ================= CA, INC.
*-------- (O)BTAIN / (F)IND *
(C)ALC R (F)IRST <R> S/A (E)rase (A)ll R (F)inish
(D)BEKEY <R> K (N)EXT <R> S/A (C)onnect R S (C)omit
(D)UP R (P)rior <R> S/A (D)isconnect R S (R)ollback
(C)URRENT R/S/A (L)ast <R> S/A (M)odify R
(O)wner S (U)Sing R S F (S)tores R PF1: HELP
Expanded Command (LRF) Screen

The standard Data Manipulation screen contains only one line for entry of CA IDMS DML and CA IDMS DMLO commands. This is normally sufficient. However, CA IDMS DML does allow for the possibility of very long commands. This could occur particularly in the case of complex WHERE expressions for LRF records. The Expanded Command screen is displayed when you enter the command SET LRF ON, or when you specify that option on the Options screen. This screen format allows for a multi-line command input area. If this is still insufficient for a particular command, you may use the EQUATE facility to establish abbreviations for phrases/words in the command.

You may switch back and forth between expanded command and standard format via the SET LRF OFF/ON command or the OPTIONS screen as desired.

mm/dd/yy........................RECORD : EMPLOYEE......................hh:mm:ss
02 EMP-ID-0415..............................N 0023
02 EMP-NAME-0415............................G
03 EMP-FIRST-NAME-0415.....................A KATHERINE
03 EMP-LAST-NAME-0415......................A O'HEARN
02 EMP-ADDRESS-0415.............................G
03 EMP-STREET-0415.........................A 12 EAST SPEEN ST
03 EMP-CITY-0415............................A NATICK
03 EMP-STATE-0415............................A MA
03 EMP-ZIP-0415..............................G
04 EMP-ZIP-FIRST-FIVE-0415................A 02364
04 EMP-ZIP-LAST-FOUR-0415................A 9999
02 EMP-PHONE-0415...........................N 6178897134
02 STATUS-0415..............................A 01
DML/O Rnn.nn ================================== ERROR STATUS : 0000 ================================== CA, INC.
*--------- (OB)TAIN / FIND *
(C)ALC R (F)IRST <R> S/A (E)RASE (A)LL R (FIN)ISH
(D)BKEY <R> K (N)EXT <R> S/A (CO)NNECT R S (COM)IT
(DU)P R (P)RIOR <R> S/A (DI)SCONNECT R S (ROL)LBACK
(CU)RRENT R/S/A (LO)AST <R> S/A (MODIFY R
(O)WNER S (US)ING R S (STO)RE R
-OK- DBKEY: 75007-1

STATUS=0000 LINE 0001 OF 0027
Interrupt Screen

The Interrupt screen is displayed when you press the "interrupt" key during the mainline CA IDMS DMLO session. The interrupt key is a key pre-defined at CA IDMS DMLO installation and changeable for each session.

The Interrupt screen prompts you to enter an indicator of your "intentions"; i.e., to CONTINUE the session or to ROLLBACK/CONTINUE and terminate the session.

The screen exists for compatibility with earlier versions as well as protection against inadvertent use of the interrupt key.

⚠️ **Note:** The interrupt key is also used to escape from selection lists which expect a selection to complete the syntax of a command. In such a case, the Interrupt screen will not be displayed.

CA IDMS DMLO Session Termination

You may terminate the CA IDMS DMLO session by entering the FINISH command, or the ROLLBACK command without the CONTINUE option. If you do not specify a SAVE/NOSAVE option, the PROFILE Save screen is displayed. If you did specify one of those options, the requested PROFILE is performed and CA IDMS DMLO terminates. For more information on the FINISH and ROLLBACK commands, see CA IDMS DMLO Session (see page 18).
PROFILE Save Screen

The PROFILE Save screen is displayed based on the session termination request as described above. This screen allows you to update the active PROFILE, or to create a new PROFILE. If a PROFILE was loaded at session startup, its name will appear on this screen. You may change the PROFILE name to create/update another. You may also specify which PROFILE components are to be updated. Only those items with a "Y" will be used for update.

```
== CA IDMS DML ONLINE RELEASE nn.nn ==
SAVE SESSION PARAMETERS AS ==> EMPPROF
SELECT/EXCLUDE (Y/N) ITEMS TO BE SAVED :
  SIGNON SCREEN VALUES ==> Y
  BIND/READY OPTIONS  ==> Y
  PF KEY SETTINGS     ==> Y
  EQUATE SETTINGS     ==> Y
AND PRESS ENTER TO SAVE OR PRESS PA1 TO BYPASS FUNCTION
I5012  ERROR-STATUS 0000 RETURNED FROM FINISH
```

Using the CA IDMS DMLO CLIST Processor

The CA IDMS DMLO CLIST processor provides the facility for creating, maintaining, and executing sequences of CA IDMS DML and/or CA IDMS DMLO commands. Storage of CLIST modules and module-level maintenance is similar to storage and maintenance of PROFILES.
What is a CLIST?

A CA IDMS DMLO CLIST is a sequence of CA IDMS DML and CA IDMS DMLO commands which may be executed as a mini-program. Any CA IDMS DML and CA IDMS DMLO command may be incorporated in a CLIST. In addition, there are special CLIST commands and constructs which allow a CA IDMS DMLO CLIST to incorporate looping and conditional capabilities.

A CLIST may be written using variable arguments which are supplied when execution of the CLIST is requested. This allows you to write general-purpose routines for frequently needed database functions.

Global CLISTs are supported. These CLISTs are prepared by the CA IDMS DMLO system administrator, and are accessible in read-only form to all users of the system. When you request a list of available CLISTs, global CLISTs are flagged "SYS".

CLIST Creation

Use the EDIT command to create or update a CLIST. This command invokes the CA editor for CLIST text processing. This command is available only in a CA IDMS/DC environment.

EDIT [S.]clist-name

Where:

- S. -- forces access to global CLIST if both a user and global CLIST of same name exist.
- clist-name -- 1-8 character CLIST identifier.

Details of the CA IDMS DMLO CLIST editing commands are discussed in CLIST Editing Commands (see page 95).

If you are not the CA IDMS DMLO system administrator, an attempt to edit a global CLIST forces you into a BROWSE rather than an EDIT session. You may, however, copy global CLISTs into one of your own CLISTs during an EDIT session.

EDIT_EDIT CLIST : GETCALC                       COLUMNS 001 072
COMMAND ==>                               SCROLL ==> PAGE
****** *** TOP OF DATA **************************** CA IDMS/DMLO ***
000001 * GENERIC OBTAIN CALC ROUTINE
000002 :%1 = RECORD TO BE OBTAINED
000003 :%2 = CALC-KEY FIELD NAME
000004 :%3 = VALUE OF CALC-KEY TO BE READ
000005 *
000006 CINIT %1
000007 MOVE %3 TO %2
000008 OBTAIN CALC %1
000009 ENDC
****** *** BOTTOM OF DATA **************************** CA IDMS/DMLO
CLIST Selection/Maintenance

At any time during the session you may enter the LIST command. This will result in a selection list of available CLIST modules. The list includes CLISTs stored under your user ID as well as global CLISTs. If the first statement in a CLIST is a comment ("*" in column 1), part of that comment will be displayed in the list.

You may enter module-level maintenance commands for your CLISTs. These commands allow you to COPY (possibly from another user), RENAME, and KILL (delete) CLISTs. For complete descriptions of the maintenance commands, see CA IDMS DMLO Session (see page 18).

You may enter CLIST maintenance commands at anytime during a session, not only when the CLIST Selection/Maintenance screen is displayed.

<table>
<thead>
<tr>
<th>ACTION</th>
<th>CLIST</th>
<th>SRC</th>
<th>DESCRIPTION</th>
<th>USED</th>
<th>ADDED</th>
</tr>
</thead>
<tbody>
<tr>
<td>_</td>
<td>PUTTESTQ</td>
<td>*</td>
<td>BUILD TEST QUEUES</td>
<td>mmdyy</td>
<td>mmdyy</td>
</tr>
<tr>
<td>_</td>
<td>PUTTESTS</td>
<td>*</td>
<td>BUILD TEST SCRATCH</td>
<td>mmdyy</td>
<td>mmdyy</td>
</tr>
<tr>
<td>_</td>
<td>TEST01</td>
<td>*</td>
<td>NULL TEST CLIST</td>
<td>mmdyy</td>
<td>mmdyy</td>
</tr>
<tr>
<td>_</td>
<td>TEST02</td>
<td>*</td>
<td>NULL TEST CLIST</td>
<td>mmdyy</td>
<td>mmdyy</td>
</tr>
<tr>
<td>_</td>
<td>TEST03</td>
<td>*</td>
<td>NULL TEST CLIST</td>
<td>mmdyy</td>
<td>mmdyy</td>
</tr>
<tr>
<td>_</td>
<td>OBTCALC</td>
<td>SYS</td>
<td>GENERIC OBTAIN CALC</td>
<td>mmdyy</td>
<td>mmdyy</td>
</tr>
</tbody>
</table>

DML/O Rnn.nn =============== CA, INC. RECORDD=employee STATUS=0000 DBKEY=0000075008-0001 KEY4=0000075007-0001 LIST I6010 ACTION = (E)dit, (X)ecute, or (K)ill

CLIST Execution

Use the following command to request execution of a CLIST:

```
EXEC [S.][clist-name] a1 a2 a3 a4 a5 a6 a7 a8 a9
```

Where:

- **S.** - forces access to global CLIST if both a user and global CLIST of same name exist.

- **clist-name** - 1-8 character CLIST identifier.

- **a1 ... a9** - represent optional arguments for replacement of CLIST variables appearing within the body of the CLIST.

You may incorporate CLIST variables within any CLIST statements. These variables are of the form %1 ... %9. This variable capability allows you to prepare general-purpose modules for frequently used functions.

Arguments supplied with the EXEC statement may be record, area, set, scratch/queue names, element names, keywords, or literals of any acceptable format. The arguments will replace all occurrences of the corresponding variables within the CLIST statements at the time the statement is executed.
You may indicate that an optional argument is null by entering ",," for that argument. This will cause the corresponding symbol to be replaced by a null string in the CLIST command. You may test if argument $n$ is null with a conditional test for \%NULL$n$. This conditional name is "true" if the corresponding argument is supplied as a null value.

The CLIST facility allows you to use DEFINE symbols (\%$n$) for documenting the use of CLIST arguments ahead of the executable CLIST commands. Not only will this provide documentation for a user who is editing the CLIST, but it allows the CLIST processor to prompt for missing arguments. If a CLIST command is encountered which includes a CLIST variable for which a corresponding argument has not been provided, the processor will display a screen which allows you to supply the missing argument value. If a DEFINE exists for the missing argument, the associated documentation is also displayed as shown below.

For more information, see CLIST Editing Commands (see page 95).

```
mm/dd/yy.....................GETCALC COMMAND PROMPT...................hh:mm:ss

THE FOLLOWING COMMAND CONTAINS UNRESOLVED CLIST ARGUMENTS. YOU MUST PROVIDE
VALUE(S) TO CONTINUE EXECUTION. AFTER ENTERING VALUES, PRESS THE "END" KEY

CINIT %1

:%1 = RECORD TO BE OBTAINED
:%1 001-064..
:%1 065-128..
:%1 129-192..
:%1 193-256..

DML/O Rnn.nn === CLIST = GETCALC DEPTH C=01 R=00 ======== CA, INC.
RECORD=EMPLOYEE STATUS=0000 DBKEY=0000075008-0001 KEY0=0000000000-0000
EXEC GETCALC
SUBSCHEMA=EMPSS01 SCHEMA=EMPSCHM VER=0100 COL 001-080 LINE 0001 OF 0012
```

If you have no idea of what arguments are needed for a given CLIST, "EXEC clist-name ?" will cause the CLIST processor to prompt for all arguments, and display all DEFINE information. Such documentation for any non-trivial CLIST is highly recommended. Note that a CLIST may EXEC another CLIST. Thus, you may define CLIST subroutines called by other CLISTS. CA IDMS DMLO will ensure that recursion does not occur; i.e., a CLIST does not appear at multiple levels in an active EXEC nest.

### CLIST Execution Modes

CA IDMS DMLO provides the following CLIST execution modes:

- **FAST Mode**
- **STEP Mode**
- **STEP/LEARN Mode**

Selection of execution mode is done with the SET CLIST FAST/STEP command or with the Options screen.

Specification of LEARN Mode is done with the LEARN ON/OFF command or with the Options screen.

**FAST Mode Execution**
FAST Mode execution executes series of commands without interruption until a stopping point is reached. A stopping point is any error condition, a PAUSE or PROMPT command, or end of the CLIST. If an error condition occurs, execution will revert to STEP Mode. FAST Mode does not provide any screen displays until a stopping point occurs.

**STEP Mode Execution**

STEP Mode execution executes one command at a time and displays the results of that command. You must press the ENTER key to cause the next command to be executed. In between CLIST commands you may enter additional commands to be executed. When you press the ENTER key without a command line entry, the CLIST will execute its next command.

During STEP Mode, CA IDMS DMLO displays the next command to be executed in the message area of the display.

**STEP/LEARN Mode Execution and Command Input**

The LEARN Mode for command input during STEP Mode CLIST execution will execute and permanently insert into the CLIST source any commands which you enter when the CLIST is paused.

IFC and REPEAT commands may not be entered during LEARN Mode execution.

Use the LEARN ON/OFF command to initiate/terminate LEARN Mode. You may enter the command before starting CLIST execution, or at any time during execution. It may also appear as a command within the CLIST itself.

The CA IDMS DMLO display indicates that you are in LEARN/STEP Mode.

### CLIST Commands

CA IDMS DMLO’s CLIST processor provides extensive CLIST capabilities. Detailed information on CLIST command syntax can be found in the online documentation and in *CA IDMS DMLO Session* (see page 18).

- **CLIST Control Commands**
  - EXEC
  - EXITC
  - ENDC
  - QUIT
- **CLIST Iteration Commands**
  - REPEAT
  - EXITR
  - ENDR
CLIST Status Test Command
- ON

CLIST Conditional Commands
- IFC
- ENDIF
- ELSE

CLIST Data Manipulation Command
- MOVE

CLIST Interrupt Commands
- PAUSE
- RESUME
- PROMPT

CLIST Record Processing Commands
- CBIND
- CINIT

CLIST Examples
This section contains three CLIST examples which use a variety of CA IDMS DMLO CLIST processing features:

Note: For detailed CLIST command syntax information, see the section "CA IDMS DMLO Sessions"

CLIST Example #1
This CLIST deletes a specified record.

EXEC CALCDEL record-name calc-field calc-key del-opt

1. CINIT %1
2. MOVE %3 to %2
3. OBTAIN CALC %1
4. ON ANYERR QUIT REQUESTED RECORD NOT FOUND
5. IFC %NULL4
   ERASE %1
ELSE
ERASE %1 %4 MEMBERS
ENDIF
(6) ENDC

(1) Conditionally BIND and load the dictionary elements of record
(2) Set calc-field to calc-key
(3) Obtain the selected record (record-name)
(4) If any CA IDMS error, stop execution and display 'REQUESTED RECORD NOT FOUND'.
(5) Determine format of CA IDMS DML ERASE command based on content (or NULL status) of del-opt.
(6) Required CLIST terminator

CLIST Example #2

This CLIST sweeps an area or set, looking for a field value in a record. It makes use of variable processing/prompting and conditional statements.

EXEC SCANFOR record set/area search-fld search val max-rec

(1) **** SWEEP AREA OR SET LOOKING FOR FIELD VALUE IN A RECORD
(2) :%1 = PRIMARY RECORD NAME
    :%2 = SET OR AREA NAME TO BE SWEPT
    :%3 = FIELD NAME TO BE TESTED
    :%4 = VALUE TO LOCATE
    :%5 = MAX RECORDS TO CHECK W/O HIT
    *
(3) OBTAIN FIRST %1 IN %2
(4) REPEAT %5
(5) ON 0307 EXITR
(6) IFC %3 = %4
    EXITR
    ENDF
(7) OBTAIN NEXT %1 IN %2
(8) ENDR
(9) ENDC

(1) COMMENT line
(2) Define lines (used during variable prompting)
(3) Start search process
(4) Start of loop
(5) On end-of-set, leave loop
(6) If item of interest found, leave loop
(7) Get next record
(8) Required loop terminator
(9) Required CLIST terminator

⚠️ Note: In order for SCANFOR to work, you must establish currency for the owner if a set is selected.

CLIST Example #3
This CLIST is an example of an 'application' CLIST which incorporates nesting of CLISTS. For a specified R-CUSTOMER record, locate the first R-PO (in CUST-PO set) for which PO-DATE is a specified value. If for that record, PO-UNSHIP = 0.00, delete the record.

EXEC DELZERO 'SMITH' 890501

(1) **** DELETE SELECTED R-PO RECORDS
(2) :%1 = R-CUSTOMER RECORD KEY (MAX 20 BYTES)
    :%2 = PO-DATE VALUE TO BE LOCATED (YYMMDD)
    *
(3) CINIT R-CUSTOMER
(4) MOVE %1 TO RCUST-KEY
(5) OBTAIN CALC R-CUSTOMER
(6) ON DB-REC-NOT-FOUND QUIT RECORD NOT FOUND
(7) EXEC SCANFOR R-PO CUST-PO PO-DATE %2 UNTIL
(8) ON DB-END-OF-SET QUIT NO PO FOR %2
(9) IFC PO-UNSHIP = 0.00
    ERASE R-PO ALL MEMBERS
    ELSE
    QUIT PO HAS NONZERO UNSHIP
    ENDIF
(10) ENDC

(1-2) COMMENT / DEFINEs
(3) Ensure R-CUSTOMER field definitions loaded, and bound
(4) Set desired record key
(5) Read record of interest
(6) If not found, exit with message
(7) Call CLIST SCANFOR to walk CUST-PO set (see CLIST Example #2)
(8) If not found, exit with message
(9) Delete record or exit with message depending on PO-UNSHIP
(10) Required CLIST terminator

Commands 2

CA IDMS DMLO includes a powerful syntax processor which allows you to combine command-line entry, variable symbols, PF key text, and abbreviations (EQUATEs) to create executable commands. This section covers the following command-related topics:

- CA IDMS DML Commands and Restrictions (see page 44)
- CA IDMS DML Control Statements (see page 44)
- CA IDMS DML Retrieval Statements (see page 45)
- CA IDMS DML Modification Statements (see page 48)
- CA IDMS DML Save Statements (see page 49)
- LRF Commands (see page 50)
- CA IDMS DML SCRATCH/QUEUE Processing (see page 51)
- CA IDMS DML Command Restrictions (see page 54)
- CA IDMS DMLO Commands (see page 55)
- PROFILE Maintenance Commands (see page 70)
CA IDMS DML Commands and Restrictions

CA IDMS DMLO supports all CA IDMS DML commands except for those exceptions listed below.

This section assumes that you are generally familiar with CA IDMS DML commands and understand the concept of CA IDMS currency. See the appropriate CA IDMS sections for detailed information on these topics.

CA IDMS DML Control Statements

The following is a list of CA IDMS DML control statements in the formats accepted by CA IDMS DMLO:

/  
BIND < record-name >   ALL  
\  
/  
READY [area-name] USAGE-mode [IS] < PROTected > < RETrieval >   EXCLusive   UPDATE  \  
/  
COMMIT [TASK] [ALL]   NOSAVE  
FINISH [TASK] < SAVE >   SAVE=profile-name  
\  CONTinue  \  
ROLLBACK [TASK] < NOSAVE >  
\  \  SAVE  \  SAVE=profile-name  /  
IF set-name IS [NOT] EMPty [imperative-statement]  
IF [NOT] set-name MEmber [imperative-statement]  
KEEP [EXCLusive] CUrrent record-name  
KEEP [EXCLusive] CUrrent < WITHin > < set-name >  IN  \  area-name /
CA IDMS DML Retrieval Statements

The following is a list of CA IDMS DML retrieval statements in the formats accepted by CA IDMS DMLO:

- Using FIND/OBTAIN Commands (see page 46)
- FIND/OBTAIN Special Considerations (see page 48)

Format 1

< FIND > [KEEP [EXCLusive]] [record-name] DB-KEY [IS] < KEYn >\ OBTain / \ dbkey-literal /

Format 1B

< FIND > [KEEP [EXCLusive]] DB-KEY [IS] < KEYn >\ OBTain / \ dbkey-literal /
PAGE-INFO < PGRn > \ pgr-literal /

Format 2

< FIND > [KEEP [EXCLusive]] CUrrent record-name \ OBTain /
< FIND > [KEEP [EXCLusive]] CUrrent < WIthin > < set-name > \ OBTain / \ IN / \ area-name /

Format 3

/ Next \ PRIor
< FIND > [KEEP [EXCLusive]] < First > [record-name] \ OBTain / \ Last \ number /
/ WIthin / \ set-name \< IN \> < area-name >

Format 4

< FIND > [KEEP [EXCLusive]] OWner < WIthin > set-name\ OBTain / \ area-name \ IN /

Format 5

/ CALC \< FIND > [KEEP [EXCLusive]] < ANY > record-name \ OBTain / \ DUPlicate /

Format 6

< FIND > [KEEP [EXCLusive]] record-name < WIthin > area-name
Using FIND/OBTAIN Commands

In the FORMAT 3 FIND/OBTAIN the clause 'WITHIN set-name/area-name' may be omitted. If omitted, this will default to 'WITHIN area-name' and CA IDMS DMLO will determine the correct area-name for the record requested.

FORMAT 1, 5, and 6 of the FIND/OBTAIN commands require definition of a symbolic key or database key before execution. The following discussion explains how to execute these commands in the CA IDMS DMLO environment.

Format 1

< FIND [KEEP [EXCLusive]] [record-name] DB-KEY [IS] < KEYn > / OBTain / dbkey-literal /

The Format 1 FIND/OBTAIN command requires a database key. The db-key value may be provided as a literal or in one of the ten CA IDMS DMLO KEYPAD fields. The dbkey-literal may be specified as PPPPPP-LLLL (page-line format), X'hhhhhhhh' (hexadecimal format), or F'nnnnnnnn' (decimal format).

A value may be established in the KEYn KEYPAD by using one of the following commands:

1. ACCEPT KEYn FROM ..... CURRENCY
2. RETURN KEYn FROM ..... 
3. MOVE ..... TO KEYn

Format 1B

< FIND [KEEP [EXCLusive]] DB-KEY [IS] < KEYn > PAGE-INFO < PGRn > / OBTain / dbkey-literal / pgr-literal /

The PAGE-INFO value may be provided as a literal or in one of the ten CA IDMS DMLO PGR0 - PGR9 KEYPAD fields. The pgr-literal may be specified as X'hhhhhhhh' (hexadecimal format) or F'nnnnnnnn' (decimal format).

A value may be established in the PGRn KEYPAD by using one of the following commands:

ACCEPT PGRn FOR record-name
ACCEPT KEYn FROM ..... CURRENCY

Format 5

/ CALC /
< FIND [KEEP [EXCLusive]] < ANY > record-name / OBTain / DUPlicate /
To execute a FORMAT 5 FIND/OBTAIN CALC command you must set the record elements comprising the calc key. To accomplish this use one of the following procedures:

1. **Using the Data Manipulation Screen**
   a. Display the record type to be retrieved.
   b. Enter appropriate data in the calc key field(s).
   c. Press the ENTER key.
   d. Enter the FORMAT 5 command in the command area.
   e. Press the ENTER key or a PF key.

2. **Using the Command Line (Option 1)**
   a. MOVE key-value TO sort-key-field (for each element of sort key)
   b. Enter the FORMAT 6 command and press the ENTER key or a PF key.
   c. **Using the Command Line (Option 2)**

3. Enter the format 6 command with an appropriate literal and press the ENTER key or a PF key.

**Format 6**

```plaintext
<FIND> [KEEP [EXCLUSIVE]] record-name <WITH> area-name  
\OBTain / \IN /
/\SORTKEY \\
set-name [CURRENT USing <field-name>]
\literal /
```

The Format 6 FIND/OBTAIN USING command needs the symbolic key of a sorted set key to execute.

The method for executing the Format 6 statement is similar to the method used to execute the Format 5 statement. However, the Format 6 statement pertains to indexed records and sorted sets.

To execute this statement CA IDMS DMLO must be given the value of the sorted or index key.

To supply CA IDMS DMLO with the information, use one of the following methods:

1. **Using the Data Manipulation Screen**
   a. Display the record type to be retrieved.
   b. Enter appropriate data in the sort key field(s).
   c. Press the ENTER key.
   d. Enter the Format 6 command and press the ENTER key or a PF key.

2. **Using the Command Line (Option 1)**
Using the Command Line (Option 1)

MOVE ... sort-key-field

Start the Format 6 command press the ENTER key or a PF key.

Using the Command Line (Option 2)

Enter Format 6 command with literal of appropriate type.

FIND/OBTAIN Special Considerations

Symbolic Keys

In the Format 6 FIND/OBTAIN command, there are three options for specifying the symbolic key:

- **SORTKEY** causes CA IDMS DMLO to gather multiple sort-keys into an internal work area for the use of the actual CA IDMS DML call to CA IDMS. The SORTKEY parameter provides support for multiple non-contiguous sort-key fields, but may also be used for records with contiguous or single sort-key fields.

  If SORTKEY is entered for a set sorted by db-key, then the KEY field, as shown in the command /status area, will be used as the key for the CA IDMS DML call to CA IDMS.

- **field-name** names a field which will be used as the key for the set or index named in the command. This option is used for contiguous or single sort-key fields. The specified field-name need not be an element within the specified record.

- **literal** may be specified for any sorted set. The literal may be in character, hexadecimal, or dbkey format as may be appropriate. Use of a literal behaves as though the data had in fact been keyed into the record fields comprising the SORTKEY (literal length need not match the sum of fields).

- See the discussion of CLIST Example #2 for allowable literal formats.

CA IDMS DML Modification Statements

The following is a list of CA IDMS DML modification statements in the formats accepted by CA IDMS DMLO:

STOre record-name

/ PERManent /

ERAs record-name < SElective >

\ ALL MEmbers /

MODify record-name

CONNect record-name TO set-name

DISConnect record-name FROM set-name
CA IDMS DML Save Statements

Contents
- ACCEPT/RETURN Special Considerations (see page 49)

The following is a list of CA IDMS DML save statements in the formats accepted by CA IDMS DMLO:

```
ACCEpt KEYn FROM / record-name \
    \
    \ Currency / \
    \ set-name / \ PGRn / \ area-name / \
/ NEXT \ 
ACCEpt KEYn FROM set-name < PRIOR > Currency / \ 
    \ OWNER / \ PGRn / \
ACCEpt PGRn FOR record-name \
/ IDMS-STATISTICS \ 
ACCEpt < STATISTICS > [RESET] [DELTA] \
    \ STATS / \
    \ Currency / \
    First [Currency] \ 
    Last [Currency] \
RETURN KEYn FROM index-set-name < Next [Currency] > [KEY INto] field-name \
    Prior [Currency] \ 
    USING index-key-value-v \ 
    USING SORTKEY / 
```

ACCEPT/RETURN Special Considerations

Keyn always refers to one of the ten CA IDMS DMLO KEYPAD fields.

PGRn always refers to one of the ten CA IDMS DMLO PAGE-INFO values.

In the RETURN statement:
- `index-key-value-v` may be an element within a previously referenced record, or may be a literal value
- `field-name` is any element within a previously referenced record.

In the ACCEPT IDMS-STATISTICS statement:
- RESET and DELTA are CA IDMS DMLO extensions to standard CA IDMS DML syntax which permit display of incremental statistics.
- If neither RESET nor DELTA is used, statistics values displayed are cumulative since the start of your CA IDMS DMLO session.
- If DELTA is used, the statistics represent the incremental change since RESET was last used as an ACCEPT parameter, or if RESET was never used, since the beginning of your session.
- If RESET is used, a new "zero" point is established for later DELTA usage.
LRF Commands

Contents

- Entering LRF Commands (see page 50)

The Logical Record Facility of CA IDMS DMLO allows access to both LR and mixed subschemas. All LRF commands are available, as well as some WHERE clause capabilities.

CA IDMS DMLO supports all WHERE clause features except:

- Subscripted fields
- More than 4 levels of qualification

WHERE clause syntax is described below.

The WHERE clause can be used in conjunction with the following LRF commands:

- OBTAIN FIRST <logical-record-name> WHERE . . .
- OBTAIN NEXT <logical-record-name> WHERE . . .
- MODIFY <logical-record-name> WHERE . . .
- ERASE <logical-record-name> WHERE . . .
- STORE <logical-record-name> WHERE . . .

See the appropriate CA ADS or CA IDMS DML section for additional information on these commands.

Entering LRF Commands

Because LRF commands have the potential to be much longer than CA IDMS DML commands, a new optional format for the static part of the screen is available to deal with LRF commands. The new format may be selected using the SET LRF ON/OFF command.

CA IDMS DMLO, CA IDMS DML, and LR commands may always be entered in the MENU/ASSIST Mode or standard screen formats; the larger command area in the 'LR' format is available if needed.

To enter commands longer than the available command line(s), use the EQUATE phrase capability of CA IDMS DMLO to create a shorthand form of the command. The expanded command length may exceed the input area length.

⚠️ Note: The LRF screen is applicable not only for LRF commands, but may be used whenever an expanded command area is needed.

WHERE Clause

/ \  
\ OR /

15-Jan-2018

CA IDMS - 19.0
Where:

**expression**-DBA keyword or comparison

**dbakeyword**-DBA-designated keyword to select path

**comparison**-item1 relational item2

**relational**-MATCHES, CONTAINS, EQ, =, GT, >, LT, <, NE, GE, LE

**item**-literal

idd-defined-field-name [OF group-ID] ...

logical-rec-field-name [OF group-ID] ... [OF LR]

arithmetic expression involving any of the above

---

**CA IDMS DML SCRATCH/QUEUE Processing**

**Contents**

- Abbreviations (see page 52)
- Scratch Display (see page 52)
- Scratch/Queue Entity Variables (see page 52)
- Default Buffer Areas (see page 52)
- Default Data and Length Assignment (see page 52)

The following is a list of CA IDMS DML scratch/queue processing statements in the formats accepted by CA IDMS DMLO.

See the appropriate CA IDMS DML section for detailed information on scratch/queue syntax.

CA IDMS DMLO not only provides support for standard CA IDMS DML syntax for scratch/queue processing but also includes some powerful extensions which simplify your use of these commands. The extensions include the following:

- Abbreviations for most keywords

- SHOW SCRATCH command and selection variable (?T) to display those scratch areas associated with your signon

- Scratch/queue entity variables (&T and &Q)

- Default scratch/queue buffer areas

- Logic to assign data area and length when not included in the command.
Abbreviations

The standard abbreviations and default parameters are described below in the detailed syntax for the scratch/queue processing commands.

Scratch Display

The SHOW SCRATCH command (or ?T selection variable) allows you to list the scratch areas associated with your CA IDMS/DC terminal. You may also invoke this list by using the ?T selection variable.

Scratch/Queue Entity Variables

You may always reference the most recent scratch area ID or queue ID by using the entity variables &T and &Q. These variables may appear anywhere a scratch area ID or queue ID are required.

Default Buffer Areas

You may specify any subschema or dictionary-owned record (or elements within such records) as buffer areas for scratch/queue commands. You may also use SBUFn and QBUFn (n=0-9) which will be automatically allocated by CA IDMS DMLO. SBUF and QBUF are the default values for installation-defined prefixes. Consult your system administrator for the prefix chosen at your installation.

Default Data and Length Assignment

You may omit specification of data area and/or length for a GET or PUT command. CA IDMS DMLO attempt to establish a logical value for the omitted parameter. The rules for establishing such values are as follows:

- If you specify a record name as a data area, the default length will be the record length.

- If you specify a field name as a data area, the default length will be from the start of that field to the end of its record.

- If you omit the data area from a PUTSCR request, the default will be the area and length used in a prior PUT for the same scratch area ID.

- If you omit the data area from a GETSCR request, the default area and length will be selected from:
  - Prior GETSCR request for the same scratch area ID
  - Prior PUTSCR request for the same scratch area ID
Unused SBUFn default buffer name

If you omit the data area from a PUTQUE request, the default will be the area and length used in
a prior PUT for the same queue ID.

If you omit the data area from a GETQUE request, the default area and length will be selected
from:

- Prior GETQUE request for the same queue ID
- Prior PUTQUE request for the same queue ID
- Unused QBUFn default buffer name

GET SCRatch [[[AREA] [ID] scratch-area]
/ Next \ 
 First 
< Last >
Prior
Current
\ RECORD ID scr-rec-id /
/ 
< Delete >
\ Keep /
/ INTO buff-start < TO buff-end >
\ MAX Length max-reclen /
[RETurn LENgth [INTo] len-field]PUT SCRatch [[[AREA] [ID] scratch-area]
/ FROM buff-start < TO buff-end >
\ Length scr-reclen /
[RECORD ID scr-rec-id [REPLACE]]
[RETurn RECORD ID [INTo] rid-field]DELETE SCRatch [[[AREA] [ID] scratch-area]
/ Next \ 
 First 
< Last >
Prior
Current
\ RECORD ID scr-rec-id /
[RETurn RECORD ID [INTo] rid-field]
GET QUEUE [[[ID] queue-area]
/ Next \ 
 First 
< Last >
Prior
N th
\ RECORD ID scr-rec-id | | Sequence record-id-number | |
/ 
< Delete >
\ Keep /
/ < Lock >
\ NOLock /
CA IDMS DML Command Restrictions

Almost all of the CA IDMS DML commands are supported by CA IDMS DMLO. However, there are a few exceptions. The CA IDMS DML commands that are not supported or supported with restrictions are described below:

- These commands are not supported by CA IDMS DMLO:

  ACCEPT bind-address-v FROM record-name BIND ACCEPT proc-control-location-v FROM proc-name PROCEDURE BIND PROCEDURE FOR proc-name TO proc-control-location-v

- In the Format 3 FIND command, the integer must be explicit; no variable can be referenced. For example:

  FIND 4 ORDER WITHIN CUSTOMER-ORDER

- The IF command can be entered with or without a second compound command. When the IF command is entered as a single command the appropriate error-status and the message TRUE or FALSE is displayed. When the expression is true, the second command (if present) will be executed.

- The ACCEPT and RETURN commands normally refer to the user-defined database key and page-info identifiers. In CA IDMS DMLO, database key identifiers are of the form KEY0 - KEY9. KEY is equivalent to KEY0. Page-info identifiers are of the form PGR0-PGR9. Contents of these fields can be displayed with the SHOW KEYPADS command.

- The FIND DB-KEY IS command may specify one of the KEY identifiers or a database key literal in one of three acceptable formats are:

  - PPPPPP-LLLL (page-line format)
  - X'hhhhhhhh' (hexadecimal format)
  - F'nnnnnnnn' (decimal format)

Examples
A value may be established in a KEYn field by using ACCEPT, RETURN, MOVE, or by specifying those clauses of scratch and queue commands which return RECORD IDs.

- ACCEPT db-statistics-v FROM IDMS-STATISTICS has been changed to ACCEPT IDMS-STATISTICS.

- The BIND RECORD command syntax has been extended to include BIND and BIND ALL. The BIND command causes the Record Display screen to appear. If BIND ALL is entered, all of the records in the subschema are bound. If BIND RECORD is entered, only the specified record will be bound.

- CA IDMS DML READY command syntax is fully supported. Note that READY issued with no optional clauses readies all areas in a subschema in shared retrieval usage mode. Areas can also be readied on the Area Display screen. Access the Area Display screen by using the CA IDMS DMLO READYLIST, SHOW AREAS, or ?A commands.

- CA IDMS DMLO performs complete syntactical checks on all commands entered. For example, CA IDMS DMLO will verify that area, record, and set names used in commands are valid (i.e., they exist in the subschema). CA IDMS will check the semantics of all commands and the CA IDMS error status returned in the command area will indicate the nature of any semantic errors.

- A field with multiple subscripts is displayed with only its low-order subscript. Higher subscripts are displayed on the high-order occurring group.

**CA IDMS DMLO Commands**

- DISPLAY Command (see page 56)
- EDIT Command (see page 57)
- EQUATE Command (see page 58)
- EXEC Command (see page 59)
- HELP Command (see page 60)
- INIT Command (see page 60)
- LEARN Command (see page 61)
- LIST Command (see page 61)
- MOVE Command (see page 61)
- PRINT Command (see page 62)
- READYLIST Command (see page 63)
- SAVE Command (see page 63)
- SET AUTOBIND Command (see page 63)
- SET CLIST Command (see page 64)
- SET CMDDisplay Command (see page 64)
- SET COBDISPLAY Command (see page 65)
- SET DEFENTK Command (see page 65)
- SET EXIT Command (see page 65)
CA IDMS DMLO commands are entered in the command area. These commands are summarized below.

<table>
<thead>
<tr>
<th>Command</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>DISPLAY</td>
<td>Display a record, group, or element within a record using various display options.</td>
</tr>
<tr>
<td>EDIT</td>
<td>Create or edit a CLIST.</td>
</tr>
<tr>
<td>EQUATE</td>
<td>Establish abbreviations for any keyword, phrase, or subschema entity.</td>
</tr>
<tr>
<td>EXEC</td>
<td>Execute a CLIST.</td>
</tr>
<tr>
<td>HELP</td>
<td>Access CA IDMS DMLO online documentation.</td>
</tr>
<tr>
<td>INIT</td>
<td>Load record description from the dictionary and/or initialize buffer.</td>
</tr>
<tr>
<td>LISTC</td>
<td>Access the CLIST Selection/Maintenance screen.</td>
</tr>
<tr>
<td>LEARN</td>
<td>Activate Learn Mode during CLIST Step Mode execution.</td>
</tr>
<tr>
<td>MOVE</td>
<td>Transfer data.</td>
</tr>
<tr>
<td>PRINT</td>
<td>Print the complete current formatted display or the currently displayed screen.</td>
</tr>
<tr>
<td>READYLIST</td>
<td>Display the Area Selection screen.</td>
</tr>
<tr>
<td>SAVE</td>
<td>Display the PROFILE Save screen for intermediate PROFILE saves.</td>
</tr>
<tr>
<td>SET</td>
<td>Set various session values.</td>
</tr>
<tr>
<td>SHOW</td>
<td>Display various session parameters and subschema entities.</td>
</tr>
</tbody>
</table>

**DISPLAY Command**

```
DISPLAY / record-name \ / CONTINUE \ < SUBSCHEMA-CTRL > \ HEX \ SSCTRL / < NATive > \ DUMP \ NODUMP /

DISPLAY field-name / OF \ record-name / HEX \```

CA IDMS - 19.0
The DISPLAY command displays a record, group, element within a record, or field. Each record has an I/O buffer that can be displayed independently of any CA IDMS DML command.

You can display a record or field in hex or native, or use the DUMP option to display a record or field in side-by-side hex and character formats. The DUMP option overrides, but does not replace, the current HEX/NATIVE setting for the specified record or field. The NODUMP option causes the display option to revert to HEX or NATIVE, whichever was active when DUMP was specified.

When using the DUMP display option, you can make updates in either format at the same time. Data changes are processed left-to-right as follows:

1. Any changed hex data is applied to the record/field buffer.
2. Any character data which differs from the original buffer contents is applied to the record/field buffer.
3. The changed line is completely reformatted.

A group or element can be displayed to initialize a key field prior to obtaining a calc, indexed, or sorted record. CA IDMS DMLO assigns a unique number to each FILLER field (e.g., FILLER#004). The optional 'WITHIN record-name' clause can be used to request a non-unique element or group name.

The DISPLAY command can also display the current contents of the Subschema-Control block. DISPLAY record-name CONTINUE allows you to begin formatting at the point in the record where storage was exhausted.

**DISPLAY Command Rules**

1. Group and element names cannot be subscripted.
2. A record display may exhaust available CA IDMS DMLO display storage before all fields (or occurrences) are formatted. A warning message is issued to inform you of this condition. DISPLAY CONTINUE allows you to begin formatting at the point in the record where storage was exhausted.
3. CONTINUE may only be requested for the most recently OBTAINed or DISPLAYed record.
4. If neither field-name nor record-name is specified, the most recently displayed record will be redisplayed according to and format parameter specified.

**EDIT Command**

`EDIT [S.]clist-name`

Where:

S.-forced access to a global CLIST when there is a global CLIST with the same name as a CLIST associated with your user ID.

clist-name-1-8 character CLIST identifier.
Use the EDIT command to create or edit a CLIST. This invokes an editor session. See CLIST Editing Commands (see page 95) for more information on CLIST editing commands.

You may also request an EDIT session for a CLIST by entering "E" next to an item on the CLIST Selection/Maintenance screen.

**EQUATE Command**

```
/ old-word \
EQuate new-word < phrase > \
 null /
```

Where:

- **new-word** - alphanumeric token which begins with a letter, and does not include an ";", "@", or "%".
- **old-word** - any word currently defined to the CA IDMS DMLO session. This includes commands, keywords, literals, area-names, record-names, set-names, field-names, or other EQUATE symbols.
- **phrase** - an arbitrary set of tokens which may include any symbols eligible for **old-word** as above, as well as macro variable symbols ("&" and "@”).
- **NULL** - removes the association of **new-word** from the list of session EQUATEs. You can re-use **new-word** during the session.

Use the EQUATE command to establish convenient abbreviations for frequently used entities, commands, or partial commands. Some examples of EQUATE usage are listed below.

See the discussion of Extended Command Processing for a more detailed description of the iteration of PF keys, EQUATEs, variables, and command specification.

**EQUATE Command Examples**

**EQUATE EMP EMPLOYEE**

This example establishes a simple abbreviation for the record EMPLOYEE. Anywhere the record-name could be used, the EQUATE symbol EMP may appear.

**EQUATE OFR OBTAIN FIRST**

This example establishes an EQUATE symbol for a partial command. You could enter OFR IN EMP-DEMO-REGION as an acceptable CA IDMS DML command.

**EQUATE ONWR OBTAIN NEXT &1 WITHIN &2**

This example establishes an EQUATE symbol for partial command. You could enter ONRW EMP &A to result in an effective CA IDMS DML command of OBTAIN NEXT EMPLOYEE WITHIN EMP-DEMO-REGION.

**EQUATE Command Rules**

1. EQUATE symbols you define during a session may be preserved in a PROFILE when you end the session. They will then be re-established at the start of subsequent session if you request that PROFILE.
2. EQUATEs may be nested three deep; i.e., you may EQUATE a symbol to a phrase which includes EQUATE symbols, and some of those symbols may be phrases which include EQUATE symbols which reference phrases.

3. EQUATE symbols may be incorporated in PF/PA key text.

4. CLIST variable symbols (%n) may not be contained within EQUATE phrases or as part of EQUATE symbols.

**EXEC Command**

```
EXEC [S.]clist-name < A1 ... A9 > \
? /
```

Where:

- **S.**-forced access to a global CLIST when there is a global CLIST with the same name as a CLIST associated with your user ID.

- **clist-name**-1-8 character CLIST identifier.

- **a1...a9**-represent optional arguments used for macro-like replacement within the body of the CLIST. These arguments may be keywords, subschema entity names, or literals of any acceptable format. Within the CLIST, a1 will replace each occurrence of %1, etc.

- **?**-causes the CLIST processor to prompt for all arguments and display all DEFINE information.

Use the EXEC command to execute an existing CLIST.

You may incorporate CLIST variables within any CLIST statements. These variables are of the form %1 ... %9. This variable capability allows you to prepare general-purpose modules for frequently used functions.

Arguments supplied with the EXEC statement may be record, area, set, scratch/queue names, element names, keywords, or literals of any acceptable format. The arguments will replace all occurrences of the corresponding variables within the CLIST statements at the time the statement is executed.

You may indicate that an optional argument is null by entering ",," for that argument. This will cause the corresponding symbol to be replaced by a null string in the CLIST command. You may test if argument n is null with a conditional test for %NULLn. This conditional name is "true" if the corresponding argument is supplied as a null value.

The CLIST facility allows you to use DEFINE symbols (%n) for documenting the use of CLIST arguments ahead of the executable CLIST commands. Not only will this provide documentation for a user who is editing the CLIST, but it allows the CLIST processor to prompt for missing arguments. If a CLIST command is encountered which includes a CLIST variable for which a corresponding argument has not been provided, the processor will display a screen which allows you to supply the missing argument value. If a DEFINE exists for the missing argument, the associated documentation is also displayed.
If you have no idea of what arguments are needed for a given CLIST, "EXEC clist-name ?" will cause the CLIST processor to prompt for all arguments, and display all DEFINE information. Such documentation for any non-trivial CLIST is highly recommended. Note that a CLIST may EXEC another CLIST. Thus, you may define CLIST subroutines called by other CLISTS. CA IDMS DMLO will ensure that recursion doesn’t occur; i.e., a CLIST doesn’t appear at multiple levels in an active EXEC nest.

HELP Command

HELP

Use the HELP command to access CA IDMS DMLO online documentation.

CA IDMS DMLO online documentation includes:

- Complete descriptions of all CA IDMS DMLO screens
- Comprehensive information concerning various CA IDMS DMLO processing topics
- New features summaries
- Online message facility. You can access the online message text by entering option M at the first screen of any online documentation module.

The CA IDMS DMLO online documentation print utility gives you the ability to create a structured hardcopy printout of any CA IDMS DMLO online documentation module, including CA IDMS DMLO messages. See Commands for instructions on using the online documentation print utility.

INIT Command

INIT rec-name [Version] < Highest > [REPL]
\ Lowest /

Where:

nnnn-record version number

REPL-force reload of the record element descriptions for dictionary-owned records

Use the INIT command to ensure that all subschema or dictionary-owned record element descriptions are available and to initialize all elements to null values appropriate to their usage: numeric fields to zero, and others to spaces.

Normally, CA IDMS DMLO will automatically issue an INIT for a record the first time that record appears in an CA IDMS DML command. However, if you need to refer to an element within that record before the record itself has been referenced, you must issue the INIT command to force the load of record elements from the dictionary.
Once a subschema record has been loaded, you cannot reload the elements from the dictionary. However, you can reload a dictionary-owned record to allow access to a different version, or to refresh the currently loaded version if an update to the dictionary has occurred since you started the CA IDMS DMLO session.

Dictionary-owned records may be used as CLIST work areas or as input/output areas for scratch and queue processing.

**LEARN Command**

```
/ ON \ 
LEARN < OFF > \
```

Use the LEARN command to activate the Learn Mode of command input during CLIST Step Mode execution. This mode allows you to key in commands. The commands are then executed and inserted into the active CLIST at the current point. IF and REPEAT statements are not eligible for Learn Mode.

The Learn Mode gives you the ability to dynamically update CLISTs during your normal CA IDMS DMLO processing.

You can initiate Learn Mode before the CLIST is executed or at any time during CLIST execution. It can also be used within a CLIST itself.

CA IDMS DMLO displays *(LEARN)* on the screen to let you know that you are in Learn/Step Mode.

**LIST Command**

```
LIST
```

Use the LIST command to:

- During session startup to redisplay the PROFILE Maintenance screen.

- At any other time during the CA IDMS DMLO session mainline to access the CLIST Selection /Maintenance screen.

**MOVE Command**

```
MOVE source TO target 
```

Where:

- *source* and *target* are any of the following:
  - Fields in records
  - Subschema control entities
  - ERROR-STATUS
CA IDMS DMLO keypads (KEY0-KEY9)
  - DBKEY
  - DIRECT-DBKEY

Figurative constants
  - SPACES
  - ZEROS
  - LOW-VALUES
  - HIGH-VALUES

Literals of all types
  - alpha
  - numeric
  - hex
  - database key
  - full word

Use the MOVE command to move data without having to display the records containing the target fields.

PRINT Command

PRINT [ALL] [optional-heading]

Where:

ALL-Print the complete current formatted display

optional-heading-An optional heading line

Use the PRINT command to instruct CA IDMS DMLO to print the currently displayed screen or the complete current formatted display (ALL keyword specified).

Print Command Rules

1. PRINT is valid in a TSO environment.

2. A valid PRINT command causes one of the following messages:
   - PRINT SUCCESSFUL (self explanatory)
   - PRINT ERROR (open or allocation error)
If no message occurs, you are in an unsupported environment for the PRINT command (CICS).

3. CA IDMS DMLO does not verify that the current formatted display matches the actual contents of the database. CA IDMS DMLO prints field modifications even if a record has not been stored or modified on the database.

4. Maximum length for the optional heading is restricted to the remainder of the command area following the PRINT command.

5. PRINT can be issued only from screens which have a command line area.

**READYLIST Command**

```
< READYList >
\ RL       /
```

Use the READYLIST command to display the Area Display screen.

You can use the Area Display screen to check the ready status of areas in the subschema and to ready areas. The Area Display screen displays the following information:

- The names of the areas in the subschema
- The usage mode of each area that has been readied

This command is included only for compatibility with earlier versions of CA IDMS DMLO. It is equivalent to the SHOW AREAS command.

**SAVE Command**

```
SAVE
```

Use the SAVE command to display the PROFILE Save screen to perform intermediate PROFILE saves.

**SET AUTOBIND Command**

```
[SET] < AUTOBind > < ON >
\ ABIND      / \ OFF /
```

Use the SET AUTOBIND command to specify requirements for BINDing subschema records during an CA IDMS DMLO session. AUTOBIND ON will cause CA IDMS DMLO to automatically issue a BIND record-name command the first time that record-name is referenced in a command. AUTOBIND OFF forces you to enter an explicit BIND command before you can use a record. Note that it is more efficient to use the AUTOBIND ON option rather than issuing BIND ALL or individual BIND commands.

If ON or OFF is not specified, the option is toggled. SET AUTOBIND corresponds to "AUTO BIND RECORDS" on the Options Screen.
Use the SET AUTOHEX command to automatically change the 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 AUTOHEX corresponds to "AUTOHEX DATA DISPLAY" on the Options Screen.

**SET CLIST Command**

Use the SET CLIST command to change the execution mode for execution of CA IDMS DMLO CLISTs. If FAST or STEP is not specified, the option is toggled.

FAST mode execution will execute CLIST commands continuously until a PAUSE, PROMPT, QUIT, or ENDC command is encountered. STEP mode execution executes one CLIST command at a time. You must press ENTER after each command to continue execution. The next command in the CLIST will be displayed in the message display area.

**Note:** Display of status and error messages takes precedence over pending command display.

**SET CMDDISPLAY Command**

Use the CMDDISPLAY command to specify whether commands will be displayed as I(input) or U(used) -- i.e., after the expansion of variables.

If I or U is not specified, the option is toggled. SET CMDDISPLAY corresponds to "AS USED CMND DISPLAY" on the Options Screen.

**Example**

As an example of this option, consider the following:
PF3 has been set to OBT N &1 in &2

TEACHER &A was keyed into the command line and PF3 pressed

If option "I" is in effect, the command line will be unchanged.

If option "U" is in effect, the command line will contain:

OBT N TEACHER IN &A

**SET COBDISPLAY Command**

```
[SET] < COBDisplay > < ON > < OFF >
```

Use the SET COBDISPLAY command to change the display format to one with indentation and level numbers similar to the standard COBOL record layout.

CA IDMS DMLO displays the native and the display usage of the item.

If OFF is specified, the display format will be vertical. If ON or OFF is not specified, the option is toggled.

The SET COBDISPLAY corresponds to "COBOL DISPLAY FORMAT" on the Options Screen.

**SET DEFENTK Command**

```
SET DEFEntk < ON or OFF >
```

Use the SET DEFENTK command to change the default ENTER key use, when the ENTER key alone is hit, with no other data being typed/overtyped on the command line. By default, hitting the ENTER key alone will clear the line. If DEFENTK is set to OFF, then the last command on the DMLO command line (if any) will be re-executed. This feature is useful, for example, when navigating a database to repeat FIND/OBTAIN NEXT/PRIOR records without having to overtype data on the command line.

This value is initially set at install time and may be subsequently altered by changing the source of parameter module USDTPARM and re-assembling/re-linking same. It can also be switched dynamically at run time with the above SET DEFENTK command.

**SET EXIT Command**

```
[SET] EXIT < ON > < ALL > < Verb >
```

Use the SET EXIT command to turn on/off calls to the installed exit program, either completely or for specified CA IDMS DML verbs.

The ability to use the SET USEREXIT command is specified at product installation using the customization macro. See your system administrator if you have any questions.
If ON or OFF is not specified, the option is toggled.

**SET HEX/NATIVE Command**

```
SET record-name / HEX \ < NATive >
  DUMP \ NODUMP /

SET field-name / OF \ < Within > record-name / HEX \ < NATive >
  DUMP \ NODUMP /
```

Use the SET HEX/NATIVE command to explicitly change the display format of the specified entity to/from hexadecimal format or the subschema-defined format, or in side-by-side hex and character formats (DUMP option).

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 DMLO displays the level number and usages (as shown below) of the specified element in the message area.

The DUMP option overrides, but does not replace, the current SET HEX/NATIVE setting for the specified record.

The NODUMP option causes the display option to revert to HEX or NATIVE, whichever was active when DUMP was specified.

You can make updates in either format at the same time. Data changes are processed left-to-right as follows:

1. Any changed hex data is applied to the record buffer.
2. Any character data which differs from the original buffer contents is applied to the record buffer.
3. The changed line is completely reformatted.

**SET INVCHAR Command**

```
[SET] INVCHAR C
```

Where "c" represents a character which will be substituted for any non-displayable character when DUMP is selected as the display option.

SET INVCHAR corresponds to "INVALID CHAR REPLACE" on the Options Screen.

```
<table>
<thead>
<tr>
<th>CA IDMS DMLO Descriptions</th>
<th>Schema Description Usage</th>
<th>Schema Description Picture</th>
</tr>
</thead>
<tbody>
<tr>
<td>GROUP</td>
<td>N/A</td>
<td>X(n)</td>
</tr>
</tbody>
</table>
```
### CA IDMS DMLO Descriptions

<table>
<thead>
<tr>
<th>CA IDMS DMLO Descriptions</th>
<th>Schema Description Usage</th>
<th>Schema Description Picture</th>
</tr>
</thead>
<tbody>
<tr>
<td>FILLER</td>
<td>N/A</td>
<td>X(n)</td>
</tr>
<tr>
<td>BIT</td>
<td>BIT</td>
<td>X(n)</td>
</tr>
<tr>
<td>DISPLAY</td>
<td>DISPLAY</td>
<td>X(n)</td>
</tr>
<tr>
<td>SDISPLAY</td>
<td>DISPLAY</td>
<td>S9(n)</td>
</tr>
<tr>
<td>UDISPLAY</td>
<td>DISPLAY</td>
<td>9(n)</td>
</tr>
<tr>
<td>SPACKED</td>
<td>COMP-3</td>
<td>S9(n)</td>
</tr>
<tr>
<td>UPACKED</td>
<td>COMP-3</td>
<td>9(n)</td>
</tr>
<tr>
<td>SBINARY</td>
<td>COMP</td>
<td>S9(n)</td>
</tr>
<tr>
<td>UBINARY</td>
<td>COMP</td>
<td>9(n)</td>
</tr>
</tbody>
</table>

### SET LOWERCASE Command

```
[SET] LOWERcase / \ ON < > \ OFF /
```

Use the SET LOWERCASE command to allow CA IDMS DMLO to accept lowercase data.

If ON or OFF is not specified, the option is toggled.

⚠️ **Note:** In an CA IDMS/DC environment, this option will be ineffective unless the DC LTE will accept lowercase input; i.e., specified via the DCUF SET UPLOW command (or CA task code CAPS OFF).

### SET LRF Command

```
[SET] LRF / \ ON < > \ OFF /
```

Use the SET LRF command to change the display format to create a larger command area to allow for longer LRF command input.

If ON or OFF is not specified, the option is toggled. The SET LRF corresponds to "LRF EXPAND CMND AREA" on the Options Screen.

⚠️ **Note:** You do not need to be using an LRF subschema for this command to be possible.
SET MAPIN Command

[SET] MAPIN < FAST > \ STEP /

Where:

FAST - Allow you to enter a command and change data in a record buffer in the same converse.

STEP - Do not allow command entry and data change in a record buffer in the same converse.

If neither FAST nor STEP is specified, the option is toggled. The SET MAPIN corresponds to "FAST DATA INPUT MODE" on the Options Screen.

SET MENU Command

[SET] MENU < ON > \ OFF /

Use the SET MENU command to select the Menu Mode operation of CA IDMS DMLO.

The Menu Mode provides the less experienced CA IDMS user with a friendlier, structured environment, as well as a tool by which user CA IDMS DML syntax can be learned.

The following features are provided with the CA IDMS DMLO Menu Mode:

- Structured presentation of CA IDMS DML commands
- User-tailorable menu
- Entity selection lists
- Instructional command echo.

If ON or OFF is not specified, the option is toggled. The SET MENU corresponds to "DML/O 'MENU' MODE" on the Options Screen.

SHOW AREANAMES Command

[SHOW] < AREANAMES > \ AREAS /

Use the SHOW AREANAMES command to display all areas in the subschema. CA IDMS DMLO responds by displaying the Area Display screen. Entry of the selection variable ?A as the command results in the same effect as if a SHOW AREA NAMES command were entered.

SHOW EQUATES Command

[SHOW] EQuates
Use the SHOW EQUATES command to display all user-defined EQUATE symbols. CA IDMS DMLO responds by displaying the Equates Display screen. An example of this screen is shown in Data Manipulation and Display (see page 14).

SHOW KEYPADS Command

[SHOW] KEYPads

Use the SHOW KEYPADS command to display all database key and page-information values. CA IDMS DMLO responds by displaying the Database Key Display screen. An example of this screen is shown in PF Key Usage (see page 14).

SHOW OPTIONS Command

[SHOW] OPTIONS

Use the SHOW OPTIONS command to display the Options screen. The Options screen is used to set various session parameters specified on the Signon screen or using various options of the SET command. An example of this screen is shown in CLIST Programming (see page 13).

SHOW PFKEYS Command

[SHOW] PFKeys

Use the SHOW PFKEYS command to display the current PF key settings on the PF Key Display screen. An example of this screen is shown in Comprehensive Online Documentation (see page 13).

For more information on using the PF Key Display screen, see the Comprehensive Online Documentation (see page 13).

SHOW RECORDNAMES Command

[SHOW] < REcordnames >
  \ RECS /

Use the SHOW RECORDNAMES command to display all records in the subschema. CA IDMS DMLO responds by displaying the Record Display screen.

Entry of the selection variable ?R as the command results in the same effect as if a SHOW RECORDNAMES command were entered.

SHOW SCRATCH Command

[SHOW] SCRatch
Use the SHOW SCRATCH command to display all scratch areas. CA IDMS DMLO responds by displaying the Scratch Display screen. An example of this screen is shown in Selection Lists (see page 14).

Entry of the selection variable ?T as the command results in the same effect as if a SHOW SCRATCH command were entered.

SHOW SETNAMES Command

[SHOW] < SETNAMES >
\SETS /

Use the SHOW SETNAMES command to display all sets in the subschema. CA IDMS DMLO responds by displaying the Sets Display screen. An example of this screen is shown in EQUATE Facility.

Entry of the selection variable ?S as the command results in the same effect as if a SHOW SETNAMES command were entered.

SHOW VARIABLES Command

[SHOW] VARIABLES

Use the SHOW VARIABLES command to display the current values of subschema variables. CA IDMS DMLO responds by displaying current value of the variables in the message area.

PROFILE Maintenance Commands

Contents

- User PROFILE Maintenance Commands (see page 70)
- System Administrator PROFILE Maintenance Commands (see page 72)

PROFILE maintenance commands can only be entered on the PROFILE Selection/Menu screen. PROFILE maintenance command syntax can be divided into the following groups:

- **User Maintenance Commands**-allow maintenance of PROFILEs associated with the signon user ID.

- **System Administrator Maintenance Commands**-allow the system administrator to act upon PROFILEs of any/all users of the system.

User PROFILE Maintenance Commands

User PROFILE maintenance commands include:

- LIST
LIST Command
LIST [userid]

Use the LIST command to refresh the list of available PROFILEs. By specifying a userid, you can view the PROFILEs associated with another user. The list includes global PROFILEs.

KILL Command
KILL profile-name

Use the KILL command to delete a PROFILE associated with your user ID. You may also delete a PROFILE by entering a "K" next to a PROFILE associated with your user ID on the PROFILE Maintenance screen.

After entering the KILL command, enter the LIST command to refresh the display.

COPY/REPL Command

< COPY > [S.]profile1 [OF user1] [TO profile2]
\ REPL /

Where:

S.-forces access to globally-owned profile1 if you have a PROFILE of the same name.

profile1-name of source PROFILE.

OF USER1-specifies a user ID other than your own.

TO profile2-specified target PROFILE name. The default is source name.

REPL-must be used to overlay a pre-existing PROFILE whose name matches the target name in the request.

Use the COPY command to copy a PROFILE.

After entering this command, enter the LIST command to refresh the display.

RENAME Command
RENAME profile1 TO profile2

Use the RENAME command to change the name of a PROFILE associated with your user ID.

After entering this command, enter the LIST command to refresh the display.
System Administrator PROFILE Maintenance Commands

System administrator PROFILE maintenance commands include:

- KILL ALL
- RENAME ALL

**KILL ALL Command**

```
KILL ALL PROFILE < OF userid > \ BY DATE mm/dd/yy /
```

Where:

- **OF userid** - specifies a mass delete of all PROFILEs associated with the specified userid.
- **BY DATE mm/dd/yy** - specifies the mass delete of all PROFILEs which have not been accessed since the specified date.

Use the KILL ALL command to delete all the PROFILEs associated with a user ID or all the PROFILEs that have not been accessed since the specified date.

⚠️ **Note:** CA IDMS DMLO updates a PROFILE with the current date whenever it is selected for a session. The BY DATE option allow you to purge dormant PROFILEs. You may enter "U" next to a PROFILE on the selection list to force the last-used date to be updated to the current date. This allows you to quickly protect a set of PROFILEs from BY DATE deletion.

**RENAME ALL Command**

```
RENAME ALL user1 TO user2 [REPL]
```

Use the RENAME ALL command to remove a PROFILE from ownership of user1 and associate it with user2. If user2 has a PROFILE of the same name as user1, that PROFILE remains with user2 unless the REPL (replace) option is specified. If user2 doesn't exist, it is created.

⚠️ **Note:** The system administrator also has the ability to execute the "LIST userid" command, and delete PROFILEs directly from the list by entering "K" next to the selected items. Users other than the system administrator may issue the LIST userid command, but may not perform any update/delete functions against them.

**CLIST Maintenance Commands**

**Contents**
CLIST maintenance commands may be entered at any time after the session startup procedure is complete. Normally they are entered in the command line of the CLIST Maintenance screen. Use the LIST command to display the CLIST Maintenance screen.

CLIST maintenance command syntax can be divided into the following groups:

- **User Maintenance Commands** - allow maintenance of CLISTs associated with the signon user ID.

- **System Administrator Maintenance Commands** - allow the system administrator to act upon CLISTs of any/all users of the system.

### User CLIST Maintenance Commands

User CLIST maintenance commands include:

- LIST
- KILL
- COPY
- REPL
- RENAME

**LIST Command**

LIST [userid]

Use the LIST command to refresh the list of available CLISTs. By specifying a *userid*, you can view the CLISTs associated with another user. If no user ID is specified, the list includes global CLISTs.

**KILL Command**

KILL clist-name

Use the KILL command to delete a CLIST associated with your user ID. You may also delete a CLIST by entering a "K" next to a CLIST associated with your user ID on the CLIST Maintenance screen.

After entering the KILL command, enter the LIST command to refresh the display.

**COPY/REPL Command**

< COPY > [S.]clist1 [OF user1] [TO clist2] \ REPL /

Where:

*S.*-forces access to globally-owned *clist1* if you have a CLIST of the same name.
clist1-name of source CLIST/

OF user1-specifies a user ID other than your own.

TO clist2-specified target CLIST name. The default is source name.

REPL - must be used to overlay a pre-existing PROFILE whose name matches the target name in the request.

Use the COPY command to copy a CLIST.

After entering this command, enter the LIST command to refresh the display.

RENAME Command

RENAME clist1 TO clist2

Use the RENAME command to change the name of a CLIST associated with your user ID.

After entering this command, enter the LIST command to refresh the display.

System Administrator CLIST Maintenance Commands

System administrator CLIST maintenance commands include the following:

- KILL ALL
- RENAME ALL

KILL ALL Command

KILL ALL CLIST < OF userid > \ BY DATE mm/dd/yy /

Where:

OF userid-specifies a mass delete of all CLISTs associated with the specified userid.

BY DATE mm/dd/yy-specifies the mass delete of all CLISTs which have not been accessed since the specified date.

Use the KILL ALL command to delete all the CLISTs associated with a userid or all the CLISTs that have not been accessed since the specified date.

⚠️ Note: CA IDMS DMLO updates a CLIST with the current date whenever it is selected for a session. The BY DATE option allows you to purge dormant CLISTs. You may enter "U" next to a CLIST on the selection list to force the last-used date to be updated to the current date. This allows you to quickly protect a set of CLISTs from BY DATE deletion.
RENAME ALL Command

RENAME ALL user1 TO user2 [REPL]

Use the RENAME ALL command to remove a CLIST from ownership of user1 and associate it with user2. If user2 has a CLIST of the same name as user1, that CLIST remains with user1 unless the REPL (replace) option is specified. If user2 doesn't exist, it is created.

⚠️ Note: The system administrator also has the ability to execute the "LIST userid" command, and delete CLISTS directly from the list by entering "K" next to the selected items. Users other than the system administrator may issue the "LIST userid" command, but may not perform any update/delete functions against them.

CLIST Commands 1

Contents
- CLIST Control Commands (see page 76)
- CLIST Documentation Commands (see page 76)
- CLIST Iteration Commands (see page 77)
- CLIST Status Test Command (see page 78)
- CLIST Conditional Commands (see page 78)
- CLIST Data Manipulation Command (see page 80)
- CLIST Interrupt Commands (see page 80)
- CLIST Record Processing Commands (see page 81)

CLIST command syntax can be divided into the following groups:
- Control Commands
- Documentation Commands
- Iteration Commands
- Status Test Command
- Conditional Commands
- Data Manipulation Command
- Interrupt Commands
- Record Processing Commands

Additional instructions on using CLIST commands, including CLIST examples, can be found in Enter Key Usage (see page 17).
CLIST Control Commands

CLIST control commands include:

- EXEC
- ENDC
- EXITC
- QUIT

EXEC Command

EXEC

Use the EXEC command to invoke execution of a CLIST. CLIST executions may be nested 9 deep. See CLIST Execution (see page 38) for more information on executing CLISTS.

ENDC Command

ENDC

Use the ENDC command to indicate the physical end of a CLIST module. If encountered during execution, it functions as a GOBACK; i.e., it returns to the next higher CLIST, or to the native CA IDMS DMLO session controller.

EXITC Command

EXITC

Use the EXITC command to provide for immediate exit from the currently active CLIST. EXITC behaves like a GOBACK.

QUIT Command

QUIT

Use the QUIT command to provide a POPTOP function. It has the effect of consecutive EXITC's for each level of active CLIST. Any text following the QUIT verb on the command line will be displayed in the message area. Thus, it provides a convenient "panic-abort" function.

CLIST Documentation Commands

CLIST documentation commands are identified by symbols rather than by keywords. The documentation commands are:

- COMMENT identified by "*" in column 1 of the CLIST command
- DEFINE identified by ":%n" in column 1 of the CLIST command

COMMENT Command
A COMMENT is a non-executable CLIST command which you may use for any documentational purpose within the body of a CLIST. The only COMMENT of any significance to the CLIST processor is the first command of a CLIST. If this command is a COMMENT, it will appear on the CLIST Maintenance screen as part of the informational display.

**DEFINE Command**

The DEFINE command is used to document the use of variable symbols within a CLIST. You may use one DEFINE command (:%n) for each CLIST variable (%n) appearing in the CLIST. The DEFINE command is not executable, but does have special significance to the CLIST processor. If you fail to supply a value (or NULL indicator) for a CLIST variable when you EXEC the CLIST, the CLIST processor will recognize that fact when it encounters a command using that CLIST variable (%n). The CLIST processor will construct the CLIST Command Prompt screen using the DEFINE command(s) for the missing variable(s) in the current command. You may then supply a value for the CLIST variable(s) which will be used until CLIST execution terminates.

**CLIST Iteration Commands**

CLIST iteration commands include:

- REPEAT
- ENDR
- EXITR

**REPEAT Command**

REPEAT UNTIL
REPEAT nnnn

Where:

UNTIL—signifies the loop is indefinite (i.e. commands within the loop will indicate termination).

nnnn—an integral literal representing a specific number of iterations. REPEAT groups may be nested to a level of 9. Each group must begin with REPEAT and end with ENDR.

Use the REPEAT command to provide an iteration facility for the CA IDMS DMLO CLIST.

**ENDR Command**

ENDR

Use the ENDR command to indicate the end of a REPEAT group.

**EXITR Command**

EXITR

Use the EXITR command to cause the CLIST execution to continue with the first command following the next ENDR command; i.e., to immediately exit the current level of REPEAT.
CLIST Status Test Command

**ON Command**

\texttt{ON \textit{x...x s...s}}

Where:

- \textit{x...x}-one of the following:
  - \textit{nnnn}-4-digit CA IDMS ERROR-STATUS code
  - \textit{mnemonic}-equivalent of 4-digit code
  - \textit{lrf-status-text} to match LR-STATUS

- \textit{s...s}-any CA IDMS DML or CA IDMS DMLO command (except IFC and REPEAT)

Use the \texttt{ON} command to provide an easy way to test the CA IDMS ERROR-STATUS or LR-STATUS.

**Examples**

\begin{verbatim}
ON 0307 EXITR
(If end-of-set exit current loop.)

ON ANYERR OBTAIN OWNER WITHIN EMP-POSITION
(If error-status non-zero execute CA IDMS DML obtain.)

ON DB-REC-NOT-FOUND QUIT SELECTED RECORD NOT FOUND
(If record-not-found stop the CLIST processing and display message.)

ON LR-NOT-FOUND QUIT SELECTED LR RECORD NOT FOUND
(If specified LR not found, stop CLIST processing and display message.)

ON LR-PROBLEM QUIT LR MISHAP TOOK PLACE
(If DBA-defined path status occurs, stop CLIST and display message.)
\end{verbatim}

**Standard Mnemonics Provided**

The following are standard mnemonics. Additional mnemonics can also be established using the EQUATE command.

- 0000 DB-STATUS-OK
- \textit{nnnn} ANYERROR
- 0307 DB-END-OF-SET
- 0326 DB-REC-NOT-FOUND
- 1205 DB-DUP-ON-STORE
- 0805 DB-DUP-ON-MOD
- 0705 DB-DUP-ON-CONN

**CLIST Conditional Commands**

CLIST conditional commands include:

- IFC
- ELSE
These commands provide a conditional facility for the CA IDMS DMLO CLIST. Allowable formats are:

```
IFC conditional
  true-stmt1
  ...
  true-stmtn
ELSE
  false-stmt1
  ...
  false-stmtn
ENDIF
```

Where:

- **true-stmtx** - represents any CA IDMS DML or CA IDMS DMLO commands, including EXEC, IFC, and REPEAT.
- **false-stmtx** - represents any CA IDMS DML or CA IDMS DMLO commands, including EXEC, IFC, and REPEAT.
- **valid conditionals** - operand1, relational, operand2, or %NULLn
- **valid relationals** - = < > EQ GT GE LE NE NG NL MATCHES CONTAINS
- **valid operands** - all the following are valid operands:
  - field-name [OF record-name]
  - ERROR-STATUS (subschema control)
  - DIRECT-DBKEY (subschema control)
  - DBKEY (subschema control)
  - CA IDMS DMLO KEYPADS (KEY0 - KEY9)
  - Character literal 'ABCDEFG'
  - Hexadecimal literal X'hhhhhhhh'
  - Numeric literal 123 or 12.345
  - Fullword literal F'1234567'
  - Database key literal pppppp-lll
  - Figurative constant HIGH-VALUES LOW-VALUES SPACES ZEROS
- **%NULLn** (only to test %n for null value)
CLIST Data Manipulation Command

MOVE Command

MOVE source TO target

Where:

source and target may be any of the following:

- Fields in records
  - ERROR-STATUS
- CA IDMS DMLO KEYPADS (KEY0-KEY9)
- Figurative constants
  - SPACES, ZEROS
  - LOW-VALUES, HIGH-VALUES
- Subschema control entities
  - DBKEY, DIRECT-DBKEY
- Literals of all types
  - alpha 'HERE IS A LITERAL' or 'J' 'JONES'
  - numeric -123.45
  - hexadecimal X'005A2E0B
  - database key 20510-11
  - fullword F'5250571'

The generalized MOVE command, although not a CLIST-only command, provides a powerful capability for CLIST execution, and thus is included here.

CLIST Interrupt Commands

CLIST interrupt commands include:

- PAUSE
- PROMPT
- RESUME
PAUSE Command

PAUSE display-string

Where:

display-string - any displayable string

Use the PAUSE command to provide for a temporary interruption of the CLIST execution.

Execution will temporarily be suspended and the display-string will be displayed in the message area. Execution will restart whenever the ENTER key is pressed with no command entry. You may enter commands while CLIST execution is suspended.

PROMPT Command

PROMPT display-string

Where:

display-string - any displayable string

Use the PROMPT command to provide for a temporary interruption of the CLIST execution.

Execution will temporarily be suspended and the display-string will be displayed in the message area. Execution will restart only when RESUME is entered on the command line. Until such time, commands and data may be freely entered.

RESUME Command

RESUME

Use the RESUME command to restart CLIST execution after a PROMPT command is encountered.

**Note:** RESUME may not be part of a CLIST; it is only for section entry on the command line.

CLIST Record Processing Commands

CLIST record processing commands include:

- CBIND
- CINIT

CBIND Command

CBIND record-name

Where:

record-name - any record name
Use the CBIND command to provide for a conditional BIND of a subschema record if it has not previously been bound.

**CINIT Command**

```
 CINIT  record-name
```

Where:

* `record-name`: any record name

Use the CINIT command to provide for a conditional loading of record elements from the dictionary.

CINIT initializes the buffer of the named record. If necessary, the record will be bound, and the elements of the record will be loaded from the dictionary.

---

**PF Key Processing**

CA IDMS DMLO allows commands or partial commands to be assigned to PF keys. This can greatly reduce the command data entry requirements during an CA IDMS DMLO session. PF keys assignments may be established in the following ways:

- Your data administrator may assign default PF key values during product installation.

- You can press a PF key (rather than the ENTER key) immediately after keying a command. This will assign the keyed command text to the PF key.

- You can enter the SHOW PFKEYS command to display all PF key settings, and enter text directly on the screen.

PF key settings can be preserved as part of the session PROFILE.

The only formal restriction as to what text may be associated with a PF key is the restriction against using CLIST variable symbols (%1-%9) within the text. PF keys may contain EQUATE symbols and full or partial commands, and may incorporate variable symbols.

Length of the text associated with a PF key is initially set to the physical screen width. If a longer text area is required (probably in an LRF context), it may be established by keying in a long command on the command line and pressing the desired PF key. Available length will be expanded to the next higher multiple of screen width sufficient to contain the long text string. Once the text area has been expanded, you may use the PF Key Display screen to update whatever length has been allocated.

Text associated with a PF key is normally treated as though it had been keyed ahead of any command line text (unless the command line begins with an CA IDMS DML or CA IDMS DMLO verb, in which case the PF key is assigned to the command line text).

---

**Note:** For a detailed explanation of how the CA IDMS DMLO command processor uses PF key text, see Extended Command Processing (see page 85) and Command Processing Examples (see page 88).
Variable Symbols

To provide flexible processing, CA IDMS DMLO recognizes the following types of variable symbols for use in constructing executable commands:

- Subschema Variables (see page 83)
- Macro Variables (see page 83)
- Positional Variables (see page 84)
- Selection Variables (see page 85)
- CLIST Variables (see page 85)

Subschema Variables

The subschema variables (&R, &D, &A, &S, &T, and &Q) can appear in any CA IDMS DML or CA IDMS DMLO command where an entity of the type represented by the variable is appropriate.

Variable entities are:

- &R-RECORD which is current of transaction.
- &D-RECORD most recently displayed; not necessarily the same as &R.
- &A-AREA associated with whatever record is mentioned in the CA IDMS DML command, or if no RECORD is mentioned, the AREA associated with current RECORD of transaction.
- &S-SET most recently mentioned in any CA IDMS DML command.
- &T-SCRATCH AREA ID most recently mentioned in any CA IDMS DML command.
- &Q-QUEUE ID most recently mentioned in any CA IDMS DML command.

The SHOW VARIABLES command may be used to display current values of &R, &D, &A, and &S.

Macro Variables

The macro variables (&1 . . . &9) give you the ability to replace variable symbols within a predefined (EQUATE or PF key) text string with tokens entered in the command line.

For example, assume the following EQUATE is in effect:

```
EQUATE ONRW OBTAIN NEXT &1 within &2
```

If you entered ONRW EMP-REC &A on the command line, the CA IDMS DML command actually executed would be:

```
OBTAIN NEXT EMP-REC WITHIN &A
```
Each \&n encountered requires that there be a token on the command line available for substitution. The number itself is not really significant; i.e. the first macro variable in the command text string is replaced by the first available token, the next macro variable by the second available token, etc. Thus, the following two text strings would be processed exactly the same by CA IDMS DMLO:

```
OBT NEXT &1 IN &2 OBT NEXT &7 IN &3
```

**Note:** For more information on using macro variables, see Extended Command Processing (see page 85) and Command Processing Examples (see page 88).

### Positional Variables

Positional variables (@1 . . . @9) give you the ability to replace variable symbols within a predefined (EQUATE or PF key) text string with tokens entered on the command line. Positional variables differ from macro variables in that the number (@n) is significant; i.e. @2 always refers to the second available token, and @4 always refers to the fourth. Also, because the number specifies exactly which token is referred to, that token can be used for replacement multiple times.

Suppose the following EQUATE is in effect:

```
EQUATE ONRW OBT NEXT @1 IN @2
```

This example behaves exactly as if the positional variables were macro variables:

Command line entry: ONRW EMP-REC &A

Executed command: OBTAIN NEXT EMP-REC WITHIN &A

Suppose the following EQUATE is in effect:

```
EQUATE OLRW OBT NEXT @1 WHERE @2 = @3 OR @2 = ZERO
```

This example illustrates the difference between positional and macro variables.

Command line entry: OLRW EMP-LR EMP-PAY 12.34

Executed command: OBTAIN NEXT EMP-REC WHERE EMP-PAY = 12.34 OR EMP-PAY = ZERO

Note the multiple use of @2 (= EMP-PAY) in decoding the command.

**Note:** For more information on using positional variables, see Extended Command Processing (see page 85) and Command Processing Examples (see page 88).
Selection Variables

You can use selection variables (?R, ?A, ?S, ?T, and ?) in any CA IDMS DML or CA IDMS DMLO command where an entity of the type represented by the variable is appropriate.

?R, ?A, ?S, and ?T display selection lists of subschema RECORDS, AREAS, SETS, and SCRATCH AREA ID's respectively. If an item is selected, that item will replace the selection variable in the command. The unqualified selection variable (?) displays a list of items based on context; e.g., if either SET or AREA is allowed by the CA IDMS DML syntax, "?" would cause a list of SETS to be presented first.

If no SET selection is made, a list of AREAS is presented.

Whenever a list is displayed for selection into an CA IDMS DML command, the command line is protected. Selection of an item "unprotects" the command line. To escape the selection process without making a selection, use the INTERRUPT key, or the PF key assigned to the END command.

The entity-specific selection variables (?R, ?A, ?S, and ?T) may also be entered as commands to provide lists of the entities.

CLIST Variables

When executing a CLIST, you may specify up to 9 arguments which are used to replace CLIST variables imbedded in the CLIST source commands. The syntax of the EXEC command is:

EXEC [S.]clist-name arg1 arg2 arg3 . . . arg9

where the optional arg1 through arg9 will be used to replace CLIST variables %1 - %9 which may occur throughout the CLIST source.

Each argn may be any type of token acceptable to CA IDMS DMLO. Examples of acceptable tokens are names of RECORDS, SETS, AREAS, FIELDS, literals, figurative constants, keywords, arithmetic or logical operators. Each CLIST command is expanded by replacement of %n with the argn argument string.

CLIST variable symbols (%1-%9) may not be used within EQUATES, and should not be used in PF key-assigned text. These restrictions do not apply to other types of variables.

See CLIST Examples (see page 41) for CLIST examples using CLIST variables.

Extended Command Processing

Contents

- Creating Executable Commands (see page 86)
- Commands Summary (see page 88)
- Variable Symbol Restrictions (see page 88)
This section will give you the concepts used in constructing CA IDMS DMLO and CA IDMS DML commands, together with a comprehensive set of examples.

You can access detailed information concerning CA IDMS DMLO and CA IDMS DML command syntax using the online documentation.

Preparation of commands by the CA IDMS DMLO interpreter makes use of the following:

- Text keyed on the command line
- Text associated with PF/PA keys
- EQUATEd phrases
- Variable symbol substitution

Creating Executable Commands

Use these steps to create an executable command:

1. Determine "raw" command text
2. Expand all levels of EQUATE phrases
3. Substitute variable symbols

Step 1: Determine "Raw" Command

There are 3 possible command entry combinations:

- **Command Line Entry WITHOUT PF Key Selection**
  . . . Raw Command = Entered text from command line

- **PF/PA Key Selection WITHOUT Command Line Entry**
  . . . Raw Command = Text associated with PF/PA key.

- **Command Line Entry AND PF Key Selection**
  . . . IF first token in command line represents an CA IDMS DML or CA IDMS DMLO verb (either directly or as part of an equate phrase), assign command line text to PF key:
  Raw command = Entered text from command line.
  . . . ELSE
  . . . IF first token in command line is 'NULL', Clear any text associated with PF key
  . . . ELSE
  . . . Raw command = Concatenation of (PF text and command line text)

⚠️ **Note:** This also provides one of the techniques for assigning (and clearing) text to a PF key; i.e., enter text on the command line and press the desired PF key. Another method of changing PF key associates is by using the SHOW PFKEYS command to access the PF Keys Display screen.
Step 2: Expansion of All Levels of EQUATE Phrases

The raw command created above is scanned for any words which have been EQUATEd to phrases. All such words are expanded. If any words within the expanded text present EQUATE phrases, they also are expanded -- until no further expansion is possible.

Step 3: Substitution for Variable Symbols

The final step in preparing an executable command is the resolution of any symbols requiring variable substitution contained in the raw command text. Note that there are four categories of variable symbols-only two of which are resolved at this time:

- **Subschema Entity Variables** - &R &A &S &D These variables are resolved later, as the command is executed.

- **CLIST Variables** - %1, %2, . . . %9 These variables were resolved earlier by the CLIST processor to create a text string similar to a command line entry.

- **MACRO Variables** - &1, &2, . . . &9 These variables are resolved at this stage. See the discussion below.

- **POSITION Variables** - @1, @2, . . . @9 These variables are resolved at this stage. See the discussion below.

CA IDMS DMLO replaces variable symbols by tokens which appear at the end of the raw command text. The number and placement of those tokens which are necessary to create a valid command depend on the type of variables which appear in the raw command.

**MACRO Variable Processing**

CA IDMS DMLO processes MACRO variables such that the first "&" variable, regardless of its number, receives the first token available. The second "&" variable, again regardless of its number, receives the second available token. For example, CA IDMS DMLO processes the following raw commands identically:

```
OBT NEXT &1 IN &2 VENDOR VENDOR-AREA
OBT NEXT &4 IN &5 VENDOR VENDOR-AREA
```

The resultant command in both cases would be:

```
OBT NEXT VENDOR IN VENDOR-AREA
```

**Note:** Use of "&" variables precludes multiple use of a token in a command. That is, the appearance of &1 twice in a command is always regarded as a requirement for two different tokens to be substituted rather than for a single token in two places.

**Positional Variable Processing**
Use of positional variables, on the other hand, provides a direct correlation between the number of the "@" variable, and the position of the token available of substitution. That is, the appearance of @4 within a raw command requires that there be at least four tokens for use in replacing "@" variables. Also, because the variable number and the position of the replacement are fixed, the same variable can be used more than once. Consider the following raw command example:

```
OBT FIR LR-VENDOR WHERE @1 = @2 OR @1 = @3 VENDOR-ID 123 456
```

The resultant command is:

```
OBT FIR LR-VENDOR WHERE VENDOR-ID = 123 OR VENDOR-ID = 456
```

**Combining "&" and "@" Variables**

Both types of variable symbols can appear in the same raw command. In this case, tokens associated with "&" symbols must precede tokens associated with "@" symbols. For example:

```
OBT FIR LR-VENDOR WHERE @1 = @2 or @1 = @3 LR-VENDOR VENDOR-ID 123 456
```

would be interpreted as follows:

```
OBT FIR LR-VENDOR WHERE VENDOR-ID = 123 OR VENDOR-ID = 456
```

**Commands Summary**

To summarize, the final tokens of the raw command must comprise one replacement token for each "&" symbol which appears, followed by as many replacement tokens as the highest number associated with any "@" symbol appearing in the raw command text.

**Variable Symbol Restrictions**

There are restrictions on the use of variable symbols:

- CLIST variable symbols may NOT be contained within EQUATE phrases. However, this does not preclude the use of CLIST variables as replacement tokens within the raw command.

- Replacement tokens (for MACRO and positional variables) must appear at the end of the raw command text. The variable symbols themselves may be buried arbitrarily deeply within a nest of EQUATE phrases.

**Command Processing Examples**

This section provides a variety of command processing examples:

- **Example 1** - EQUATE, macro variables
- **Example 2** - EQUATE, macro variables, PF key assignment
- **Example 3** - EQUATE, macro variables, PF key usage
Example 4 - EQUATE, macro variables, PF key usage, command entry

Example 5 - EQUATE, macro variables, PF key usage, command entry

Example 6 - EQUATE, macro variables, PF key usage, command entry

Example 7 - Multiple EQUATEs, macro and positional variables, command entry

Examples 1-4 Assume That The Following Equate Is In Effect:

EQUATE ONRW OB NEXT &1 IN &2

Example #1

Command Line Entry

ONRW VENDOR &A

Key Pressed

ENTER

Result

Execute

Example #2

Command Line Entry

ONRW VENDOR &A

Key Pressed

PF5

Result

Assign ONRW VENDOR &A to PF5

Execute OBT NEXT VENDOR IN &A

Note: Because the phrase represented by ONRW begins with a verb (OBT), CA IDMS DMLO assigns the text to PF5 regardless of whether PF5 already had text associated with it.

Example #3

Command Line Entry

ONRW
Key Pressed
PF5

Result
Assign ONRW to PF5 (See note above). No execution of CA IDMS DML command (unresolved variables).

Example #4

Command Line Entry
VENDOR ?S

Key Pressed
PF5

Result
Create command ONRW VENDOR ?S (by concatenating PF text ahead of the command line text).
Execute OBT NEXT VENDOR IN ?S (results in SET selection list).

Note: Because the command line entry did not begin with a verb, no assignment of text to PF5 was made, but rather currently assigned text was used.

Examples 5-7 Assume That The Following Equates And Pf Key Assignments Are In Effect:

EQUATE ONWP OBT NEXT LR-WARE-PART
EQUATE OLRW OBT &1 LR-VEND-PART
EQUATE XWHE WHERE @1 = @2 OR @1 = @3
PF4 ONWP
PF5 OLRW
PF6 OLRW XWHE

Example #5

Command Line Entry
WHERE WAREHSE-NBR = '012'

Key Pressed
PF4

Result
Execute OBT NEXT LR-WAREHSE-PART WHERE WAREHSE-NBR = '012'
Example #6

Command Line Entry

FIRST

Key Pressed

PF5

Result

Execute OBT FIRST LR-VEND-PART

⚠️ Note: The concatenation of PF text is ahead of command line text giving OLRW FIRST. All EQUATE phrases are expanded giving:

OBT &1 LR-VEND-PART FIRST

&1 <----(variable substitution)

After performing the substitution as indicated, the resultant command as shown above is executed.

Example #7

Command Line Entry

FIRST PART-SUFFIX 6 7

Key Pressed

PF6

Result

Execute

OBT FIRST LR-VEND-PART
WHERE PART-SUFFIX = 6 OR PART-SUFFIX = 7

⚠️ Note: The concatenation of PF text is ahead of command line text giving OLRW XWHE FIRST PART-SUFFIX 6 7.

ALL EQUATE phrases are expanded, giving:

OBT &1 LR-VEND-PART
WHERE @1 = @2 OR @1 = @3 FIRST PART-SUFFIX 6 7
After performing the substitution as indicated, the resultant command as shown above is executed.

**Operation**

**CA IDMS DMLO System Requirements**

**Terminal Type**

IBM3270-type terminal (models 2 through 5), including the 3279 color display

**Library Requirements**

Library requirements depend on the operating environment under which CA IDMS DMLO is executing.

**Security Requirements**

CA IDMS DMLO offers different levels of security. You decide which level of security checking you want CA IDMS DMLO to perform. Based on this decision, you may need to register user and subschema information in the CA IDMS dictionary.

For more information on installing CA IDMS DMLO, see the appropriate *CA IDMS Installation and Maintenance Section*.

**About Online Documentation Print Utility**

The Online Documentation Print Utility provided with CA IDMS DMLO allows error messages and other product information to be printed upon request.

The Target or Distribution source library member GSIPRINT (z/OS), TOOL JCL library member GSIPRINT.S (z/VSE), or the GSIPRINT EXEC (z/VM), downloaded from the CA IDMS DMLO installation media, contains the JCL to execute the Online Documentation Print Utility. The online documentation modules for CA IDMS DMLO processing are as follows.

The printed version of the online documentation is presented one screen per page and includes page reference indices for screen options. Characters highlighted in the online documentation appear bolded in the printed version.
**CA IDMS - 19.0**

**Note:** The characters "{}", which are used to denote optional statements in online documentation modules, appear as "& &" when printed with the Online Documentation Print Utility. The character "\|", used to denote "or" in online documentation modules, appears as "::" when printed with the Online Documentation Print Utility.

<table>
<thead>
<tr>
<th>Module Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>COMMANDS</td>
<td>CLIST Editing Commands</td>
</tr>
<tr>
<td>USDACMD</td>
<td>Product-specific editing commands</td>
</tr>
<tr>
<td>USDAKEY</td>
<td>Update PF Keys (CLIST Edit)</td>
</tr>
<tr>
<td>USDBIND</td>
<td>Bind Screen</td>
</tr>
<tr>
<td>USDCEDIT</td>
<td>CLIST Editing Screen</td>
</tr>
<tr>
<td>USDCLEX</td>
<td>CLIST Examples</td>
</tr>
<tr>
<td>USDCMAIN</td>
<td>CLIST Maintenance Screen (selection list)</td>
</tr>
<tr>
<td>USDCOMEX</td>
<td>Command Processing Examples</td>
</tr>
<tr>
<td>USDDATA</td>
<td>Data Manipulation Screen</td>
</tr>
<tr>
<td>USDDML</td>
<td>DML Commands</td>
</tr>
<tr>
<td>USDDMLO</td>
<td>CA IDMS DMLO Commands</td>
</tr>
<tr>
<td>USDEQUAT</td>
<td>Show Equate Screen</td>
</tr>
<tr>
<td>USDFG55</td>
<td>New Features Part 1.</td>
</tr>
<tr>
<td>USDFG12</td>
<td>New Features Part 2.</td>
</tr>
<tr>
<td>USDINTER</td>
<td>Interrupt Screen</td>
</tr>
<tr>
<td>USDMABB</td>
<td>Standard Abbreviations</td>
</tr>
<tr>
<td>USDMENU</td>
<td>Menu Mode Startup Screen</td>
</tr>
<tr>
<td>USDMSG</td>
<td>Error Messages</td>
</tr>
<tr>
<td>USDOPTN</td>
<td>SHOW Options/Environment Screen</td>
</tr>
<tr>
<td>USDPARAM</td>
<td>Save Session Parameters Screen</td>
</tr>
<tr>
<td>USDPFK</td>
<td>CA IDMS DMLO Mainline PF Key Assignment</td>
</tr>
<tr>
<td>USDPRMNT</td>
<td>PROFILE Maintenance Screen</td>
</tr>
<tr>
<td>USDRDYLS</td>
<td>Ready Mode Information Discussion</td>
</tr>
<tr>
<td>USDRSELAR</td>
<td>Selection List - AREA</td>
</tr>
<tr>
<td>USDRSELRE</td>
<td>Selection List - RECORD</td>
</tr>
<tr>
<td>USDRSEISE</td>
<td>Selection List - SET</td>
</tr>
<tr>
<td>USDSIGN</td>
<td>Signon Screen</td>
</tr>
<tr>
<td>USDTERM</td>
<td>Termination Status Screen</td>
</tr>
<tr>
<td>USDTUT00</td>
<td>Primary Menu</td>
</tr>
<tr>
<td>USDWHERE</td>
<td>Where Clause Syntax</td>
</tr>
<tr>
<td>USDX0001</td>
<td>Secondary Menu</td>
</tr>
</tbody>
</table>
UserExit Module

CA IDMS DMLO gives you the ability to customize a user exit module which can be invoked after each DML command is executed. At installation time the dummy user exit module USDMLXIT is provided. This module simply returns control immediately to the DMLO session.

The calling of the user exit is controlled by the installation parameter option USEREXIT in the USDTPARM module.

- If USEREXIT=YES USDMLXIT is invoked after each DML command executed.
- If USEREXIT=NO USDMLXIT is never invoked.
- If USEREXIT=(DYNAMIC, ON) USDMLXIT is not invoked after each DML command is executed and is active for the session until the SET EXIT OFF command is executed.
- If USEREXIT=(DYNAMIC, OFF) USDMLXIT is not invoked but can be activated for the session with the SET ON command. This is the default setting.

⚠️ **Note:** For more information, see the appropriate *CA IDMS Installation and Maintenance Section.*

The user exit source member Assembler module USDMLXIT is delivered in source and can be customized by the user.

USDMLXIT is dynamically loaded at runtime and invoked by DMLO upon completion of all DML commands. This includes the SCRATCH and QUEUE commands. The module is delivered as a dummy module that simply returns control immediately to the DMLO session via a 'BR R14' instruction.

To customize the module, follow these rules:

1. Use standard IBM linkage conventions.

2. DSECT USDGLOB2 is used to pass information to and from USDMLXIT. Change only certain fields in this DSECT. If the restrictions are ignored, results are unpredictable and can cause subsequent DMLO errors. See source member USDGLOB2 for complete details.

3. Upon return from USDMLXIT to the DMLO session the USERCODE value in USDGLOB2 is processed as follows:
   - If USERCODE=1-9, DMLO terminates with the error message F8801 - F8809. Otherwise the DMLO session continues as normal.
   - If USERCODE is set by the user to a 1-9 value, DMLO processes the USERQUIT code as a completion indicator as follows:
     - 0 ROLLBACK
3. FINISH

- If USERCODE=99, a 64-byte message field is passed back from the User Exit program. If the actual message length to be displayed is less than 64 bytes, then the 64-byte field must be padded with blank characters.

4. The customized source member USDMLXIT should be assembled and the link edited to create an executable load module named USDMLXIT. The link edit input statements to do this are as follows:

   z/OS
   INCLUDE OBJLIB(USDMLXIT)
   MODE AMODE(31), RMODE(ANY)
   ENTRY USDMLXIT
   NAME USDMLXI(R)

   z/VSE
   PHASE USDMLXIT, *
   MODE AMODE(31), RMODE(ANY)
   INCLUDE USDMLXIT
   ENTRY USDMLXIT

See source members USDMLXIT and USDGLOB2 for complete details on the User Exit functionality.

**CLIST Editing Commands**

This section provides a section to the CLIST editing commands that are available in the CA IDMS DML Online CLIST editor. This appendix describes each command and its syntax. If you are not the CA IDMS DMLO system administrator, an attempt to edit a global CLIST forces you into a browse rather than an edit session. You may, however, copy global CLISTS into one of your own CLISTS during an edit session.

- How to Use Editing Commands (see page 95)
- Primary Commands 4 (see page 98)
- Line Commands 4 (see page 111)

**How to Use Editing Commands**

There are the following types of editing commands:

- Scroll Options (see page 96)
- Primary Commands (see page 96)
- Line Commands (see page 96)
- Program Function Keys (see page 97)
- Entering Commands (see page 97)
- Scroll Options (see page 98)
Scroll Options

Scroll options are used to determine how many lines or columns of the module 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 a module
- 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 a module
- Turn the CAPS Mode on or off
- Display the time and date
- Copy source lines from one CLIST module to another
- Terminate Editor session, save changes made to the CLIST, and return to CA IDMS DMLO session mainline
- Save changes made to a CLIST and remain in Editor session

Line Commands

Line commands are used to:

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

PF keys are set to many frequently used commands. This allows you to 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, press that key. The command executes when you press the PF key.

Entering Commands

The following are descriptions of where commands are entered:

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

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

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

<table>
<thead>
<tr>
<th>Example</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>SUSpend</td>
<td>Keywords appear in mixed case. 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>CAPS &lt; ON &gt; \  OFF /</td>
<td>Braces enclose two or more options. Select an option.</td>
</tr>
<tr>
<td>CAPS &lt; ON &gt; \  OFF /</td>
<td>A left arrow indicates the default value.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Item</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>Order of</td>
<td>You must enter a B (before) or an A (after) line command in conjunction with the COPY command 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</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>Blanks</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>In</td>
<td>You can enter multiple primary and line commands. Primary commands must be separated by a semicolon (;).</td>
</tr>
</tbody>
</table>
Scroll Options

/ Page / \\
< Half >
\ Csr \ number-of-lines /

where:

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.

Scroll options are used 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 will remain in effect until you enter a different setting.

Primary Commands 4

Contents
- BOTTOM Command (see page 99)
- BOUNDS Command (see page 99)
- CANCEL Command (see page 100)
- CAPS Command (see page 100)
- CHANGE Command (see page 100)
- COPY Command (see page 101)
- CURSOR Command (see page 101)
- DOWN Command (see page 102)
- EDITOR-ID Command (see page 102)
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

**BOTTOM Command**

**BOTTOM**

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

This command is available in Edit and Browse Modes.

**BOUNDS Command**

**BOUNDS**

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 than the right bound. If bounds are specified incorrectly or without an operand, the default is the dataset minimum.

This command is available in Edit Mode.

**CANCEL Command**

**CANcel**

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

This command is available in Edit Mode.

**CAPS Command**

```
/ ON \  
CAPS < OFF >  
\ /  
```

The CAPS command is used to turn the CAPS Mode on and off. With the CAPS Mode on, all new alpha data is translated into uppercase. With the CAPS Mode off, the data remains unaffected. Data that was initially entered with the CAPS Mode off will remain in lowercase unless you edit the field.

This command is available in Edit and Browse Modes.

**CHANGE Command**

```
/ \  / \ 
CHANGE < ALL > < string > < replacement-string >[lb][rb]< X >  
\ FIRST / \ * / \ *  / \ NX /  
```

where:

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.
Use the CHANGE command to search for and change the next occurrence 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 not found, it is changed to the replacement string.

This command is available in Edit Mode.

**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 either single or double quotes. For example:
  
  ```
  CHANGE 'program name' 'program nmae'
  ```
- 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.

**COPY Command**

```
COPY [S.]clist-name[OF userid]
```

where:

[S.] -forces access to a global CLIST (if there is a CLIST of the same name under your user ID).

clist-name -the name of the CLIST from which you want to copy source lines.

[OF userid] -allows you to specify a CLIST belonging to another CA IDMS DMLO user ID.

Use the COPY command to copy source lines from another CLIST. To specify the location where lines will be copied, enter an 'A' (after) or 'B' (before) line command.

This command is available in Edit Mode.

**CURSOR Command**

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

This command is available in Edit and Browse Modes.

The default keys are PF12 and PF24.

**DOWN Command**

```
/ number-of-lines \
DOWN < Max >
Half Page /
```

where:

- **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.

The DOWN (scroll down) command is used 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.

This command is available in Edit and Browse Modes.

The default keys are PF8 and PF20.

**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.

This command is available in Edit and Browse Modes.

**ECHO Command**

Use the ECHO command to preserve the primary command line. If ECHO is turned on, the last command entered on the command line is preserved and redisplayed. If ECHO is turned off, the last command entered is not preserved. The ECHO setting is maintained in the Editor profile for the signed on CA IDMS/DC user id. The PROFILE command can be used to display all environmental settings, which will include the ECHO setting. The syntax for the ECHO command is the following:

```
ECHO {ON OFF}
```

Default: OFF
This command is available in edit and browse modes.

END Command

END

Use the END command to save the current CLIST module (if changed) and return to the CA IDMS DMLO session mainline.

This command is available in Edit Mode.

ENTER Command

ENTER

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

This command is available in Edit and Browse Modes. The default key is ENTER.

EXCLUDE Command

\ / first-line \ 
EXCLUDE < last-line > \
ALL / 

where:

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 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.

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

This command is available in Edit Mode.

FIND Command

\ / ALL \ / / \ / [lb[rb]] \ / X > \
FIRST \ / * \ / NX / 

where:

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.
The `FIND` command is used to search for a string in the module.

**string** specifies the string 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 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.

Use the `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. 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.

This command is available in Edit Mode.

### FIRST Command

**FIRst**

The FIRST command displays the first screen of the module.

This command is available in Edit and Browse Modes.

### KEYS Command

**KEYS**

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

This command is available in Edit and Browse Modes.

### LAST Command

**LAST**

The LAST command presents the last screen of the module.

This command is available in Edit and Browse Modes.
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.

This command is available in Edit and Browse Modes.

LOCATE Command

`Locate line-number`

where:

`line-number` specifies the number of the line to which you want to move. The line you specify will be the top line displayed on the screen.

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

This command is available in Edit and Browse Modes.

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 module, you can specify 0 as the line number, and the first line of the module will be the top line displayed.

To move 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 /`

where:

`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.

Internal storage is determined by the MEMORY command.

This command is available in Edit and Browse Modes.
NULLS Command

The NULLS command is used 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 easily use the keyboard insert mode key to insert characters, turn NULLS Mode ON so that edit will automatically insert 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 will replace trailing blanks in each field.

If edit places the cursor into a field, only blanks that follow the cursor on the line will be 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.

This command is available in Edit Mode.

PROFILE Command

The PROFILE command is used 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 signon.

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

This command is available in Edit and Browse Modes.

RCHANGE Command

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.
This command is available in Edit Mode.

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 Sequence

<table>
<thead>
<tr>
<th>Step Command</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>Part 1</td>
<td>CHANGE Work-Name-1 Work-Name-2</td>
</tr>
<tr>
<td>Part 2</td>
<td>RFIND key</td>
</tr>
<tr>
<td>Part 3</td>
<td>RCHANGE key</td>
</tr>
<tr>
<td>Part 4</td>
<td>RFIND key</td>
</tr>
</tbody>
</table>

Second Sample Sequence

<table>
<thead>
<tr>
<th>Step Command</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>Part 1</td>
<td>CHANGE Work-Name-1 Work-Name-2 RFIND key</td>
</tr>
<tr>
<td>Part 2</td>
<td>RFIND key</td>
</tr>
<tr>
<td>Part 3</td>
<td>RCHANGE key</td>
</tr>
<tr>
<td>Part 4</td>
<td>RFIND key</td>
</tr>
</tbody>
</table>

RESET Command

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

This command is available in Edit and Browse Modes.

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 will overlay the current screen with the previous screen.

This command is available in Edit and Browse Modes.

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 will reach the bottom and the message line will state, BOTTOM OF DATA REACHED. Entering RFIND again will resume the search at the top of the file. Then if the string is not found in the file, the message line will state: **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.

This command is available in Edit and Browse Modes.

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.

This command is available in Edit and Browse Modes.

The default keys are PF11 and PF23.
SAVE Command

SAVE

Use the SAVE command to save the current CLIST module (if changed). The CLIST edit session will remain active.

This command is available in Edit Mode.

TABB Command

TABB

The TABB (tab backward) command is used 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.

This command is available in Edit Mode.

TABF Command

TABF

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

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

This command is available in Edit Mode.

TABS Command

/TABS < OFF >

\tab-character

\operand

| ON | \ 

where:

- **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.

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

This command is available in Edit Mode.

**TIME Command**

```plaintext
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.

This command is available in Edit and Browse Modes.

**TOP Command**

```plaintext
TOP
```

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

This command is available in Edit and Browse Modes.

**UP Command**

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

where:

- **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.

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

This command is available in Edit and Browse Modes.

The default keys are PF7 and PF19.
Line Commands 4

Contents
- Entering Line Commands (see page 111)
- How to Use Line Commands (see page 111)
- A (after) Command (see page 111)
- B (before) Command (see page 112)
- BNDS (bounds) Command (see page 112)
- COLS (columns) Command (see page 112)
- C (copy) Command (see page 112)
- D (delete) Command (see page 113)
- X (exclude) Command (see page 113)
- I (insert) Command (see page 114)
- M (move) Command (see page 114)
- R (repeat) Command (see page 114)
- TABS Command (see page 115)

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.

A (after) Command

A
The A (after) line command is used in conjunction with the C (copy) and M (move) line commands, and the COPY primary command to copy another CLIST after the line containing the A (after) line command.

B (before) Command

B

The B (before) line command is used in conjunction with the C (copy) and M (move) line commands and the COPY primary command to copy another CLIST before the line containing the B (before) line command.

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.

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.

C (copy) Command

C[number-of-lines]

where:

number-of-lines - specifies the number of lines to be copied. The default is 1.

C - specifies a single line to be copied.

Cn - specifies the first of n lines to be copied.
**CC...CC**-specifies the first and last lines of a block of lines to be copied.

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.

**Rules for Using the Copy Line 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.
- 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.

**D (delete) Command**

\[D[number-of-lines]\]

where:

**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.

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

**X (exclude) Command**

\[X[number-of-lines]\]

where:

**number-of-lines** &ndash. 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.

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

I (insert) Command

I[number-of-lines]

where:

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

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.

M (move) Command

M[number-of-lines]

where:

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.

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.

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.

R (repeat) Command

R[number-of-times]

where:
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.

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

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 on the RR block commands differ, the greater number is used.
- No other line commands can be used on lines being repeated.

TABS Command

Type TABS in the line number field to view the current tab settings. You can also use the TABS command to change the tabs by over striking 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.

Batch CLIST Processing

This appendix provides a description of the batch CLIST processor, USDBCLST. This utility provides an easy way to migrate CLISTs between CA IDMS systems.

The batch CLIST processor supports the following functionality:

- Uploading one or more new CLISTs to the system.
- Printing one or more existing CLISTs from the system.
- Downloading one or more existing CLISTs from the system in a format which can be submitted directly for upload processing.
- Removing one or more existing CLISTs from the system.

The CA IDMS installation media contains the following sample JCL (intended for demonstration purposes only):

- USDBCLST - Contains a skeleton of the JCL required to invoke the batch CLIST processor.
For more information, see

- Work Files (see page 116)
- Batch CLIST Request Cards (see page 116)
- Batch CLIST Upload File (see page 117)

Work Files

For z/OS

Input datasets

<table>
<thead>
<tr>
<th>DD Name</th>
<th>Description</th>
<th>Usage</th>
<th>LRECL</th>
</tr>
</thead>
<tbody>
<tr>
<td>SYSIPT</td>
<td>Request cards</td>
<td>Required for PRINT, DOWNLOAD, REMOVE</td>
<td>80</td>
</tr>
<tr>
<td>UPLOAD</td>
<td>Source of CLISTs to add</td>
<td>Required for UPLOAD</td>
<td>80</td>
</tr>
</tbody>
</table>

Output datasets

<table>
<thead>
<tr>
<th>DD Name</th>
<th>Description</th>
<th>Usage</th>
<th>LRECL</th>
</tr>
</thead>
<tbody>
<tr>
<td>REPORT</td>
<td>Source of printed CLISTs</td>
<td>Optional for all functions</td>
<td>133</td>
</tr>
<tr>
<td>AUDIT</td>
<td>Status of requests</td>
<td>Required for all functions</td>
<td>133</td>
</tr>
<tr>
<td>DOWNLOAD</td>
<td>Listings of downloaded CLISTs</td>
<td>Required for DOWNLOAD</td>
<td>80</td>
</tr>
</tbody>
</table>

For z/VSE

Input Datasets

<table>
<thead>
<tr>
<th>DD Name</th>
<th>Logical Unit</th>
<th>Description</th>
<th>Usage</th>
<th>LRECL</th>
</tr>
</thead>
<tbody>
<tr>
<td>SYSIPT</td>
<td>SYS015</td>
<td>Request Cards</td>
<td>Required for PRINT, DOWNLOAD, REMOVE</td>
<td>80</td>
</tr>
<tr>
<td>UPLOAD</td>
<td>SYS016</td>
<td>Source of CLISTs to add</td>
<td>Required for UPLOAD</td>
<td>80</td>
</tr>
</tbody>
</table>

Output Datasets

<table>
<thead>
<tr>
<th>DD Name</th>
<th>Logical Unit</th>
<th>Description</th>
<th>Usage</th>
<th>LRECL</th>
</tr>
</thead>
<tbody>
<tr>
<td>REPORT</td>
<td>SYS017</td>
<td>Source of printed CLISTs</td>
<td>Optional for all functions</td>
<td>133</td>
</tr>
<tr>
<td>AUDIT</td>
<td>SYS018</td>
<td>Status of Requests</td>
<td>Required for all functions</td>
<td>133</td>
</tr>
<tr>
<td>DOWNLOAD</td>
<td>SYS019</td>
<td>Listings of downloaded CLISTs</td>
<td>Required for DOWNLOAD</td>
<td>80</td>
</tr>
</tbody>
</table>

Batch CLIST Request Cards

Use request cards to download, print, or remove one or more CLISTs. **Note:** Request cards are not required to upload a CLIST.
A request card describes the CLIST to be processed and the desired function. All variables must start in the appropriate columns. Values should be padded with spaces when necessary. The request card is described in detail below.

<table>
<thead>
<tr>
<th>Columns</th>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>01 - 08</td>
<td>user-name</td>
<td>Variable specifies a 1-8 character user name for whom the requested CLIST is stored</td>
</tr>
<tr>
<td>09 - 16</td>
<td>clist-name</td>
<td>Variable specifies a 1-8 character CLIST identifier</td>
</tr>
<tr>
<td>17 - 17</td>
<td>operation code</td>
<td>Variable specifies a 1 character operation code</td>
</tr>
</tbody>
</table>

The following table describes the operation code variable.

<table>
<thead>
<tr>
<th>Code</th>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>P</td>
<td>Print only</td>
<td>Print the requested CLIST</td>
</tr>
<tr>
<td>D</td>
<td>Download only</td>
<td>Download the requested CLIST</td>
</tr>
<tr>
<td>X</td>
<td>Print and Download</td>
<td>Print and download the requested CLIST</td>
</tr>
<tr>
<td>R</td>
<td>Remove</td>
<td>Delete the requested CLIST</td>
</tr>
</tbody>
</table>

For example, the following request card will prompt the batch CLIST processor to delete the CLIST named SAMPLE belonging to user USER0001:

USER0001SAMPLE  R

The following request card prompts the batch CLIST processor to print the CLISTs SAMP1 and SAMP2 belonging to USER0002 in the report file. Additionally, the source for these CLISTs will be stored into the download file:

USER0002SAMP1  X
USER0002SAMP2  X

Batch CLIST Upload File

The upload file is used to add one or more CLISTs to the CA IDMS system. Each CLIST described in this file must begin with a parameter card which names the CLIST and assigns it to a user. Each request must end with an end card.

⚠️ Note: No request cards are required to upload a CLIST. If a CLIST is already present on the system, the existing version is overwritten.

The parameter card is described below. All variables must start in the appropriate columns. Values should be padded with spaces when necessary.
Columns | Values | Description
--- | --- | ---
01 - 01 | `~` | constant
| | Specifies a constant representing the start of a parameter or end card
02 - 09 | user-name | Specifies a 1-8 character name of the user for whom the CLIST will be stored
10 - 17 | clist-name | Specifies a 1-8 character CLIST identifier

One or more data cards follow a parameter card. Each data card contains one line of CLIST source. These lines should follow the standard CLIST conventions. Detailed information on CLIST command syntax can be found in CA IDMS DMLO Session (see page 18).

Every CLIST must end with an end card. The end card is described in detail below. All variables must start in the appropriate columns. Values should be padded with spaces when necessary.

Columns | Values
--- | ---
01 - 01 | `~` constant
02 - 09 | `**END***` constant

For example, the following lines coded in an upload file would prompt the batch CLIST processor to add CLISTS SAMP1 and SAMP2 belonging to USER0001:

```plaintext
~USER0001SAMP1
* OBTAIN CLIST
OBTAIN FIRST %1 WITHIN %2
ENDC
~**END***
~USER0001SAMP2
* STORE CLIST
STORE %1
ENDC
~**END***
```