

Cincom

SUPRA SERVER PDM

RDM PL/1 Programming Quick Reference
(OS/390 & VSE)

P26-8375-63



SUPRA[®] Server PDM RDM PL/1 Programming Quick Reference (OS/390 & VSE)

Publication Number P26-8375-63

© 1986, 1988, 1994, 1996, 1998, 2000, 2002 Cincom Systems, Inc.
All rights reserved

This document contains unpublished, confidential, and proprietary information of Cincom. No disclosure or use of any portion of the contents of these materials may be made without the express written consent of Cincom.

The following are trademarks, registered trademarks, or service marks of Cincom Systems, Inc.:

AD/Advantage [®]	iD CinDoc [™]	MANTIS [®]
C+A-RE [™]	iD CinDoc Web [™]	Socrates [®]
CINCOM [®]	iD Consulting [™]	Socrates [®] XML
Cincom Encompass [®]	iD Correspondence [™]	SPECTRA [™]
Cincom Smalltalk [™]	iD Correspondence Express [™]	SUPRA [®]
Cincom SupportWeb [®]	iD Environment [™]	SUPRA [®] Server
CINCOM SYSTEMS [®]	iD Solutions [™]	Visual Smalltalk [®]
 gOOj [™]	Intelligent Document Solutions [™]	VisualWorks [®]
	Intermax [™]	

UniSQL[™] is a trademark of UniSQL, Inc.
ObjectStudio[®] is a registered trademark of CinMark Systems, Inc.

All other trademarks are trademarks or registered trademarks of their respective companies.

Cincom Systems, Inc.
55 Merchant Street
Cincinnati, Ohio 45246-3732
U.S.A.

PHONE: (513) 612-2300
FAX: (513) 612-2000
WORLD WIDE WEB: <http://www.cincom.com>

Attention:

Some Cincom products, programs, or services referred to in this publication may not be available in all countries in which Cincom does business. Additionally, some Cincom products, programs, or services may not be available for all operating systems or all product releases. Contact your Cincom representative to be certain the items are available to you.

Release information for this manual

The *SUPRA Server PDM RDM PL/1 Programming Quick Reference (OS/390 & VSE)*, P26-8375-63, is dated January 15, 2002. This document supports Release 2.7 of SUPRA Server PDM in IBM mainframe environments.

We welcome your comments

We encourage critiques concerning the technical content and organization of this manual. Please take the [survey](#) provided with the online documentation at your convenience.

Cincom Technical Support for SUPRA Server PDM

FAX: (513) 612-2000
Attn: SUPRA Server Support

E-mail: helpna@cincom.com

Phone: 1-800-727-3525

Mail: Cincom Systems, Inc.
Attn: SUPRA Server Support
55 Merchant Street
Cincinnati, OH 45246-3732
U.S.A.



Contents

Conventions	vii
INCLUDE statements.....	9
INCLUDE view-data statement	9
INCLUDE TIS_CONTROL statement	9
RDML statements	10
COMMIT statement.....	10
DELETE statement	10
FORGET statement	10
GET statement.....	11
INSERT statement	11
MARK statement	12
RELEASE statement.....	12
RESET statement	12
SIGN_OFF statement	12
SIGN_ON statement	12
UPDATE statement.....	12
DBAID commands.....	13
= command	13
Bye command	13
BY-LEVEL command	13
CAUTIOUS command.....	13
COLUMN-TEXT command	13
COMMIT command.....	14
DELETE command	14
ERASE command	14
FIELD-DEFN command	14
FORGET command	14
GET command.....	15
GO command.....	15
INSERT command	16
KEEP command.....	16
LINESIZE command	16
MARK command.....	16
MARKS command.....	16
OPEN command	17
PAGESIZE command	17
RELEASE command.....	17

RESET command.....	17
SIGN-OFF command.....	17
SIGN-ON command	17
SURE command.....	18
UPDATE command	18
USER-LIST command.....	18
USERS command	18
VIEW-DEFN command	18
VIEWS command	19
VIEWS-FOR-USER command	19
DBAID considerations.....	19
ASI values.....	19
FSI values.....	20
VSI values.....	20
Programming considerations.....	21
Compiling.....	21
Linking	22
OS/390.....	22
VSE.....	22

Conventions

The following table describes the conventions used in this document series:

Convention	Description	Example
Constant width type	Represents screen images and segments of code.	<pre>PUT 'customer.dat' GET 'miller\customer.dat' PUT '\DEV\RMT0'</pre>
Slashed b (b)	Indicates a space (blank). The example indicates that four spaces appear between the keywords.	<pre>BEGINbbbbSERIAL</pre>
Brackets []	Indicate optional selection of parameters. (Do not attempt to enter brackets or to stack parameters.) Brackets indicate one of the following situations: A single item enclosed by brackets indicates that the item is optional and can be omitted. The example indicates that you can optionally enter a WHERE clause.	<pre>[WHERE <i>search-condition</i>]</pre>
	Stacked items enclosed by brackets represent optional alternatives, one of which can be selected. The example indicates that you can optionally enter either WAIT or NOWAIT. (WAIT is underlined to signify that it is the default.)	<pre><u>(WAIT)</u> (NOWAIT)</pre>
Braces { }	Indicate selection of parameters. (Do not attempt to enter braces or to stack parameters.) Braces surrounding stacked items represent alternatives, one of which you must select. The example indicates that you must enter ON or OFF when using the MONITOR statement.	<pre>MONITOR {ON } {OFF}</pre>

Convention	Description	Example
<p><u>Underlining</u> (In syntax)</p>	<p>Indicates the default value supplied when you omit a parameter.</p> <p>The example indicates that if you do not choose a parameter, the system defaults to WAIT.</p> <p>Underlining also indicates an allowable abbreviation or the shortest truncation allowed.</p> <p>The example indicates that you can enter either STAT or STATISTICS.</p>	<p>[WAIT] [NOWAIT]</p> <p><u>STATISTICS</u></p>
<p>Ellipsis points...</p>	<p>Indicate that the preceding item can be repeated.</p> <p>The example indicates that you can enter multiple host variables and associated indicator variables.</p>	<p>INTO :host-variable [:ind-variable],...</p>
<p>UPPERCASE lowercase</p>	<p>In most operating environments, keywords are not case-sensitive, and they are represented in uppercase. You can enter them in either uppercase or lowercase.</p>	<p>COPY MY_DATA.SEQ HOLD_DATA.SEQ</p>
<p><i>Italics</i></p>	<p>Indicate variables you replace with a value, a column name, a file name, and so on.</p> <p>The example indicates that you must substitute the name of a table.</p>	<p>FROM <i>table-name</i></p>
<p>Punctuation marks</p>	<p>Indicate required syntax that you must code exactly as presented.</p> <p>() parentheses . period , comma : colon ' ' single quotation marks</p>	<p>(<i>user-id, password, db-name</i>) INFILE 'Cust.Memo' CONTROL LEN4</p>
<p>SMALL CAPS</p>	<p>Represent a keystroke. Multiple keystrokes are hyphenated.</p>	<p>ALT-TAB</p>

INCLUDE statements

INCLUDE view-data statement

Indicates which views are needed by the program and their placement.

```
INCLUDE [user-view-name]=
```

```
    view-name
```

```
    [user-column-list];
```

INCLUDE TIS_CONTROL statement

Must be coded in every program that issues an RDML request.

```
INCLUDE TIS_CONTROL;
```



The INCLUDE statement is not the same as the PL/1 %INCLUDE.

Code an INCLUDE statement in the LINKAGE SECTION to pass rows to external modules (subroutines).

Use TIS-OPTIONS to specify DEBUG and TRACE.

RDML statements

COMMIT statement

Identifies the recovery point of the preceding task.

COMMIT;

DELETE statement

Removes a view row from the database.

DELETE [ALL] *view-name*;

FORGET statement

Frees storage allocated by a previous MARK statement.

FORGET *data-item*
[NOT FOUND PL/1 *do-group*]
[ELSE PL/1 *do-group*];

GET statement

Identifies the row to be retrieved from the indicated view.

```
GET [ NEXT  
    LAST  
    SAME  
    FIRST  
    PRIOR ] view-name  
  
    [FOR UPDATE] [ USING data-item1 [...data-itemg ]  
                  AT mark-data-item ]  
  
    [NOT FOUND PL/1 do-group]  
    [ELSE PL/1 do-group];
```

INSERT statement

Inserts a new row into the view.

```
INSERT [ NEXT  
        LAST  
        FIRST  
        PRIOR ] view-name  
  
        [DUP KEY PL/1 do-group];
```

MARK statement

Records the current position of the view established by the previous GET, UPDATE or INSERT statement.

MARK *view-name* AT *data-item*;

RELEASE statement

Closes a specific view or all views that have been opened, and frees internal storage.

RELEASE [*view-name*];

RESET statement

Rolls back any database updates since the most recent commit point.

RESET;

SIGN_OFF statement

Signs the user off from RDM.

SIGN_OFF;

SIGN_ON statement

Identifies the user to RDM.

SIGN_ON *user-name* [*password*];

UPDATE statement

Updates column values in the database.

UPDATE *view-name*;

DBAID commands

= command

Reissues the previous RDML command.

=

Bye command

Exits DBAID.

BYE

BY-LEVEL command

Displays the column names in a view by level of occurrence, starting with the 0 level, followed by level 1, and so on.

BY-LEVEL [*view-name* [*column-number*]]

CAUTIOUS command

Prohibits an automatic COMMIT.

CAUTIOUS

COLUMN-TEXT command

Displays the short and long text for a column in a view.

COLUMN-TEXT [*view-name* [*column-name*]]

COMMIT command

Makes all updates since the most recent commit point permanent in the database.

COMMIT

DELETE command

Removes a view row from the database.

DELETE [ALL] *view-name*

ERASE command

Issues an RDM RESET if an "X" FSI is returned. This command is the opposite of KEEP.

ERASE

FIELD-DEFN command

Displays the full description of columns in a view.

FIELD-DEFN [*view-name* [*column-name*]]

FORGET command

Frees the storage allocated by a previously issued MARK command.

FORGET *mark-name*

GET command

Retrieves and displays a row for the indicated view.

```

GET [ NEXT
      LAST
      SAME
      FIRST
      PRIOR ] view-name

      [FOR UPDATE]

      [ [AT mark-name]
        [USING literal1[literal2...literaln]] ]

```

GO command

Issues a penetration GET request followed by a series of sweeping GET requests and displays the rows in tabular format.

```

GO [ NEXT
      PRIOR ] view-name

      [ START { NEXT
                LAST
                SAME
                FIRST
                PRIOR
                AT mark-name } ]

      [FOR number-of-rows]

      [ [FROM ]
        [USING ] literal1[literal2...literaln] ]

```

INSERT command

Places a view row in the physical database at the relative location specified.

```
INSERT [ NEXT  
       LAST  
       FIRST  
       PRIOR ] view-name [MASS]
```

KEEP command

Prohibits an automatic RESET.

```
KEEP
```

LINESIZE command

Specifies the number of characters to be displayed in a line.

```
LINESIZE [number-of-characters]
```

MARK command

Marks the current position of the view row established by the previous GET command.

```
MARK view-name AT mark-name
```

MARKS command

Lists all open MARKs and the view they are marking.

```
MARKS
```

OPEN command

Readies a saved or virtual view for use by DBAID.

```
OPEN [user-view-name=]  
view-name [column1[,...,columnn]]
```

PAGESIZE command

Specifies the number of lines to display on a screen/page.

```
PAGESIZE [number-of-lines]
```

RELEASE command

Closes a specific view or all views that have been opened, and releases the occupied storage within RDM.

```
RELEASE [view-name]
```

RESET command

Rolls back any database updates since the most recent commit point.

```
RESET
```

SIGN-OFF command

Signs the user off from DBAID.

```
SIGN-OFF
```

SIGN-ON command

Identifies the user to DBAID.

```
SIGN-ON user-name [password]
```

SURE command

Causes a commit point after each successful INSERT, UPDATE or DELETE.

SURE

UPDATE command

Updates data values in the database.

UPDATE *view-name*

[*column*₁:=*literal*₁[,...,*column*_{*n*}:=*literal*_{*n*}]

USER-LIST command

Displays the column list for the specified user view.

USER-LIST *view-name*

USERS command

Displays information about the current users of the system.

USERS

VIEW-DEFN command

Displays a condensed description of a view.

VIEW-DEFN [*view-name*]

VIEWS command

Displays all of the views currently active in DBAID.

VIEWS

VIEWS-FOR-USER command

Lists the names and short text for the views related to the signed-on user.

VIEWS-FOR-USER

DBAID considerations

An asterisk has two uses:

- ◆ Substitutes for the last view name used
- ◆ Denotes a comment line when entered in column 1

Enter one command per line with a maximum of 72 characters.

ASI values

Value	Meaning
=	The column exists but has not changed since the last access. (GET processing only.)
-	The column is missing and has a null value. (GET processing only.)
+	The column exists but has changed since the last access. (GET processing only.)
V	The column contains an invalid value.
C	The column value has been changed by another view.
N	The column is set to its null value. (INSERT and UPDATE processing only.)

FSI values

Value	Meaning
*	Successful Completion
D	Data Error
F	Failure
N	Failure Due to Occurrence Problem
S	Security
U	Unavailable Resource
X	Reset Recommended

VSI values

Value	Meaning
C	The column value was changed by another view.
V	At least one invalid ASI was returned.
-	No invalid ASIs were returned, but at least one missing ASI was returned.
+	No invalid or missing ASIs were returned, but at least one new physical occurrence in the database was returned.
=	No invalid, missing, or new physical occurrences were returned by this RDM function.

Programming considerations

- ◆ Each RDML statement must start on a line by itself and end with a semicolon.
- ◆ Code an INCLUDE TIS_CONTROL statement in every program that issues an RDML request.
- ◆ Use the Relational Data Manager PL/1 Programmer's Report for a description of row layout and column names in the view.
- ◆ You can code both INCLUDE and RDML statements at different levels in your program.
- ◆ View names and column names must not contain hyphens, but may contain underscores.
- ◆ CICS users must use command level PL/1. Macro level PL/1 is not supported.

Compiling

Three parameters are used for precompiling:

- ◆ *Required.* Actual precompiler name:
 - Batch. Specify compiler name of IELOAA on parm of RDML Precompiler execution step.
 - CICS. Specify CICS command level preprocessor name on parm of RDML compiler step.
- ◆ *Required.* Schema name supplied by DBA.
- ◆ *Optional.* Miscellaneous parameters; any parameters for PL/1 Compiler or CICS preprocessor.

Linking

OS/390

- ◆ **Batch.** Link with CSVILUV.
- ◆ **CICS.** Link with CSVCLUV, CSVICICS or CSVNICIC (required for running or allocating parameters above 16 megabyte line).

VSE

- ◆ **Batch.** Link with CSVIOSVS.
- ◆ **CICS.** Link with CSVCLUV, CSVICICS or CSVNICIC (required for running or allocating parameters above 16 megabyte line).