

# RDM Embedded 10.1

## Interactive Database Access Utility (IDA)

### Trademarks

Raima Database Manager® (RDM®), RDM Embedded®, RDM Server™ and DataFlow™ are trademarks of Raima Inc. and may be registered in the United States of America and/or other countries. All other names may be trademarks of their respective owners.

This guide may contain links to third-party Web sites that are not under the control of Raima Inc. and Raima Inc. is not responsible for the content on any linked site. If you access a third-party Web site mentioned in this guide, you do so at your own risk. Inclusion of any links does not imply Raima Inc. endorsement or acceptance of the content of those third-party sites.

# Contents

---

<b>Contents</b> .....	<b>3</b>
<b>The ida Utility</b> .....	<b>5</b>
1.1 Accessing ida.....	5
1.2 The ida Main Menu.....	5
Moving the Highlight on a Menu.....	6
1.3 Using ida: Concepts and Procedures.....	6
<b>Using the Main Menu Commands</b> .....	<b>10</b>
2.1 Open command.....	10
<b>Entering and Modifying Records</b> .....	<b>11</b>
3.1 Accessing the Display/Edit Record Submenu.....	11
3.2 Using the Display/Edit Record Submenu Commands.....	12
3.3 Editing a Data Field.....	13
3.4 Using Edit Modes and Field Edit Commands.....	14
<b>Using the Access Menu Commands</b> .....	<b>16</b>
4.1 Using the Access Record Commands.....	16
4.1.1 Keyscan command.....	16
4.1.2 Get command.....	18
4.1.3 Enter command.....	18
4.1.4 Modify command.....	18
4.1.5 Delete command.....	18
4.1.6 Recscan command.....	18
4.2 Using the Access Set Commands.....	19
4.3 Using the Access Currency Commands.....	21
4.3.1 Owner command.....	21
4.3.2 Member command.....	22
4.3.3 Record command.....	22
4.3.4 Auto_set command.....	23

4.3.5 Display command.....	23
4.4 Using the Access Transaction Commands.....	23
4.5 Using the Access Lock Commands.....	24
4.6 Using the Access Free Commands.....	25
4.7 Using the Access Miscellaneous Commands.....	25
<b>RDM Embedded Functions Cross-Referenced to ida Commands.....</b>	<b>27</b>
Database Administration Functions.....	27
Record Manipulation Functions.....	27
Record Navigation Functions.....	27
Set Manipulation Functions.....	28
Set Navigation Functions.....	28
Key Field Access Functions.....	28
Multi-User Database Control Functions.....	28
Currency Control Functions.....	29
Miscellaneous Control Functions.....	29

# The ida Utility

The Interactive Database Access utility (*ida*) is a tool for manipulating RDM Embedded databases. It is a full-screen, non-graphical, menu-driven utility that allows you to:

- Maintain an existing database
- Access data stored in the database
- Test a database design
- Create and modify data
- Learn to use RDM Embedded

The *ida* utility has been designed for application developers and database administrators and therefore is not suitable for most non-technical users.

Although *ida* was originally designed and developed in the 1980's, it remains familiar to many long-term Raima developers. More modern database browsing tools will also be provided as soon as they are available. The form factor of 80 characters wide and 24 characters high comes from its once state-of-the-art legacy.

## 1.1 Accessing ida

To access the *ida* program, start up a command prompt and set up the environment to include a path the RDM Embedded *bin* directory. Then just run the command, *ida*. When the *ida* title page appears, press any key to bring up the *ida* main menu.

## 1.2 The ida Main Menu

Figure 8-1 shows the first *ida* menu, referred to as the main menu. This menu provides access to all other *ida* menus.

```
IDA - RDM Embedded Interactive Database Access Utility
Open Access Initialize Close Parameters Quit
Open an RDM Embedded database
```

Fig. 8-1. *ida* Main Menu

All *ida* menus use the top three lines of each screen. Each menu includes the following:

- |          |   |
|----------|---|
| 1st line | Menu title  |
| 2nd line | A list of available commands, one of which is highlighted |
| 3rd line | A description of the highlighted command                  |

You can see what a command does by moving the highlight to that command. When a command is highlighted, the description of that command appears on the 3rd line.

### Moving the Highlight on a Menu

To move the highlight on a menu:

- To move right, press either the <Space> or the <Right-Arrow> key.
- To move left, press the <Left-Arrow> key.

To leave the `ida` utility, you use the `Quit` command from the main menu. For more information about using the commands on the `ida` main menu, including the `Quit` command, see [Using the Main Menu Commands](#).

## 1.3 Using `ida`: Concepts and Procedures

This section describes the concepts and procedures you will need in order to use the `ida` utility. These concepts and techniques include messages from `ida`, executing a command from an `ida` menu (from the 2nd line of the screen), choosing an option from a numbered list (displayed below the top three lines), moving the highlight on a numbered list, opening a database, accessing a submenu, exiting a submenu, and selecting a record to modify.

### Messages from `ida`

The last two lines of an `ida` screen may display a message from `ida`. A message may tell you that a command has been completed or that there has been an input error or database error. Messages are displayed in reverse video and require pressing a key to return to the menu.

### Executing a Command from a Menu

To execute a command from a menu:

- Highlight the command and press <Enter>

or

- Type the first letter of the command.

Executing a command from an `ida` menu invokes a submenu, a numbered list, an `ida`/RDM Embedded function, or displays information.

### Choosing an Option from a Numbered List

Following the execution of commands such as `Scan` and `Find`, a numbered list of available databases, key fields, records, or sets will be displayed. The entry that was last selected will be highlighted or, initially, the first entry will be highlighted.

Figure 8-2 shows an example of a submenu that offers a numbered option list.

```
Scan and view records based on database address
Scan First Last Next Previous X_exit
```

```
Scan and view records on database address
```

```
SELECT RECORD TYPE:
```

- 0. AUTHOR
- 1. INFO
- 2. BORROWER
- 3. INFOTEXT
- 4. KEY\_WORD
- 5. INTERSECT
- 6. SYSTEM

**Fig. 8-2. Example Selection List**

To choose an option from a numbered list do one of the following:

- Type the number of the option and press <Enter>

or

- Using the procedure for moving the highlight on a numbered list (described below), highlight the option and press <Enter>.

### Moving the Highlight on a Numbered List

When moving the highlight on a numbered list, you use a different procedure than you do when you move the highlight on a menu.

To move the highlight on a numbered list:

- To move down, type the <d> key or a down arrow.
- To move up, type the <u> key or an up arrow.

Selection lists that extend beyond a single screen will either scroll or redisplay, depending on the position of the highlighted selection.

### Opening a Database

Although you can access `ida` and view the main menu, there is little you can do until you open a database. (You can execute the `Parameters` and `Quit` commands, but the `Access`, `Initialize`, and `Close` commands will not work without an open database.) If you are new to `ida`, you can use the sample `tims` database (provided with RDM Embedded) to learn any of the procedures described in this chapter.

To open a database:

1. From the `ida` main menu, highlight and execute the `Open` command.
2. On the `Open an RDM Embedded database` submenu, execute the command for the type of user access you want. (If you are just learning to use `ida`, use the `One_User` command.)
3. Type the name of a database and press <Enter>.

## Accessing a Submenu

To perform most database operations you will use the `Access` command on the main menu. Selecting the `Access` command displays the `Database Access Commands` submenu shown in Figure 8-3.

```
Database Access Commands
Record Set Currency Transaction Lock Free Miscellaneous X_exit
```

Fig. 8-3. Database Access Commands Menu

## Exiting a Submenu

When you exit a submenu, you go back to the previous menu. There are two ways to exit a submenu:

- On the 2nd line of the submenu, highlight the `X_exit` command and press `<Enter>`

or

- Press the `<Esc>` key.

## Selecting a Record

Using `ida`, you can find a record by using the `Keyscan` command to find a keyword or you can use the `Recscan` command to find a record by using a database address. The following procedure uses the `Recscan` command.

To select a record using `Recscan`:

1. Access the `Database Access Commands` submenu (shown in Figure 8-3).
2. From that menu, highlight and execute the `Record` command.

The `Record Manipulation Functions` submenu displays (shown in Figure 8-4).

```
Record Manipulation Functions
Keyscan Get Enter Modify Delete Recscan X_exit
Scan and view records based on key
```

Fig. 8-4. Record Manipulation Functions Menu

3. From the `Record Manipulation Functions` submenu, highlight and execute the `Recscan` command.

The `Scan and view records based on database address` submenu appears (shown in Figure 8-2).

4. From the `Scan and view records based on database address` submenu, highlight and execute the `Scan` command.

## RDMc Interactive Database Access Utility Guide

5. A numbered list titled SELECT RECORD TYPE displays. Type the number of the record type you want to select.

The `Scan and View Records` submenu appears and displays a record (shown in Figure 8-5).

```
Scan and View Records
Next First Select X_exit
Display next page of records

RECORD: AUTHOR Page: 1

RECORD CONTENTS:
  1. {"Martin, James"}
  2. {"Myers, G."}
  3. {"Teorey T. & Fry, J."}
```

**Fig. 8-5. Scan and View Records Menu**

6. Highlight and enter the `Select` command. The highlight will move to the numbered list.

7. Choose an option from the numbered list. The record will be displayed in the `ida` screen

# Using the Main Menu Commands

This section describes the commands on the `ida` main menu, shown in Figure 8-1 above. For instructions on using the main menu to execute commands and access submenus, see [The `ida` Main Menu](#).

## 2.1 Open command

The `Open` command opens a database. If another database is currently open, it will first be closed. After selecting the `Open` command, you must choose the `One_User`, `Shared`, or `Exclusive` access option from the displayed submenu. Enter the name of the database to be opened.

### Access command

The `Access` command brings up the `Database Access Commands` submenu, which provides a set of functions and submenus for manipulating a particular database. This menu, described in [Using the Access Menu Commands](#) includes record functions, set functions, currency functions, transaction functions, and lock functions. The `Access` command may only be called after a database has been opened.

### Initialize command

This command, which corresponds to the `d_initialize` function, initializes the opened RDM Embedded database. This command should only be used after a database has been opened for exclusive access. It will destroy the contents of the database.

### Close command

The `Close` command closes an opened database immediately.

### Quit command

Closes any opened database and ends the `ida` session.

# Entering and Modifying Records

To edit or modify a database record, you use the `Display/Edit Record` submenu. [Accessing the Display/Edit Record Submenu](#) provides information about the commands on the `Display/Edit Record` submenu. [Using the Display/Edit Record Submenu Commands](#) gives instructions for editing a data field and [Editing a Data Field](#) provides information about `ida`'s editing modes and describes field editing commands.

## 3.1 Accessing the Display/Edit Record Submenu

To access an existing record with the `Display/Edit` submenu:

- Open the database you want.
- Access the `Database Access Commands` and `Record` submenus.
- Select a record from the database. You may use the `Keyscan`, `Get`, or `Recscan` options to select a record. The record will be made the current record.
- From the `Record Manipulation Functions` submenu (Figure 8-4), select `Modify` to display the `Display/Edit Record` submenu.

When the `Display/Edit Record` submenu is accessed, the current record is displayed. Header information identifies the name of the record, its length in bytes, the total number of data fields that can be edited, the current page number, and total pages of data fields for the record.

Following the header is the first page of fields. The name of the data field appears on the left, its contents appear on the right. An example display is shown in Figure 8-7 below.

```

Display/Edit Record
Edit Init Next Prev Write Store_key Delete_key Owner Connect X_Exit
Edit record

RECORD: INFO          SIZE: 220    TOTAL FIELDS: 5          PAGE 1 OF 1

ID_CODE               >sw001<
INFO_TITLE            >Software Reliability Principles & Practices<
PUBLISHER             >Wiley-Interscience<
PUB_DATE              >1976<
INFO_TYPE             >0<

```

Fig. 8-7. An Example Display/Edit Record Screen

To create a new record with the `Display/Edit` submenu:

- Open the database you want.
- Access the `Database Access Commands` and `Record` submenus.
- Select the `Enter` submenu and select a record type from the list.
- The `Display/Edit` submenu will show the record information (as in Figure 8-7) with empty field contents.

A field can be edited if it is not the sub-field of a `struct`. Select the `Edit` submenu command to change individual field contents.

### 3.2 Using the Display/Edit Record Submenu Commands

Figure 8-8 shows the `Display/Edit Record` submenu commands.

```
Display/Edit Record
Edit Init Next Prev Write Store_key Delete_key Owner Connect X_exit
```

Fig. 8-8. The Display/Edit Record Submenu

The first line is the menu title, and the second line lists the menu commands. This section describes each of these commands.

#### Edit command

Selecting this command invokes field-level editing of the displayed record. The field edit commands are listed in Table 8-2. The first field is highlighted and displayed in the data field edit area. Field editing is initially entered in Command mode. For information about `ida` editing modes, see [Using Edit Modes and Field Edit Commands](#).

#### Init command

Use this command to initialize a new record. Selecting this command clears the displayed record and enters field editing (see [Using Edit Modes and Field Edit Commands](#)) in Insert mode.

#### Next command

If the displayed record type consists of more data fields than can be shown on one screen, this command can be used to display the next page of fields. Note that when entering data, each page of data fields must be entered before the record is stored. Use the Next command to access each page.

#### Prev command

Selecting this command displays the previous page of data fields. Each page of data fields must be entered before the record is stored.

#### Write command

Selecting this command writes the entered or modified record to the database. If the automatic set connection feature is enabled, then the record will also be connected to each set of which it is a member and for which a current owner has previously been set.

### Store\_key command

This command is used to store optional keys in the key file. A selection list of optional keys will be displayed. Select the one to be stored for the current record.

### Delete\_key command

This command is used to delete optional keys from the key file. A selection list of optional keys will be displayed. Select the one to be deleted from the current record.

### Owner command

This command is used to assign, from the current record, the current owner of the selected set. It is provided here to facilitate data entry. Note that all sets are listed, but not all sets will allow the current record to be the owner record.

### Connect command

Selecting this command connects the current record to the set selected from the displayed list of sets. It is provided here to facilitate data entry. Note that all sets are listed, but not all sets will allow the current record to be a member record. To succeed, the current record must be of the correct type, must not already be connected to the set, and a current owner for the set instance must already be established.

## 3.3 Editing a Data Field

In the `Display/Edit Record` submenu, selecting the `Edit` or the `Init` command allows you to edit a record one data field at a time. `ida` editing incorporates many of the VI editor commands (see Table 8-2).

All editing is performed in the editing field. Just above the editing field, the name of the field being edited is displayed. If this field already contains data, the data will be displayed in the editing field. Arrows on either end of the editing field point inward if all the data in the field fits on the screen, or outward in the direction where more data can be found. The entire contents of the field are accessed by horizontally scrolling or repositioning through the field, using cursor-positioning field edit commands.

You can move to different data fields by using the up and down arrow keys, or by pressing `<Enter>` to move to the next field. Pressing `<Enter>` on the last field, or pressing `<Esc>` at any point, will return you to the `Display/Edit Record` submenu.

The format of the displayed data is similar to the format that would be used if the data were statically initialized in a C variable declaration. Each item of an arrayed (non-string) data field is separated by commas and specified in its natural sequence. Structure fields must be enclosed by braces (`{ }`). Character strings do not need to be delimited by quotation marks except when included as part of a struct field. Sub-fields of struct fields cannot be edited independently and, in fact, they are displayed only in the struct field. All C escape codes (for example, `\n`, `\f`, `\r`, `\ddd`) can be entered into character fields.

## RDM Interactive Database Access Utility Guide

One-dimensional character arrays are always treated as strings, terminated by a null byte (\0). Multi-dimensional character arrays will display the entire length of the field.

Fields of type DB\_ADDR are displayed and entered as [f : r], where f is the file number and r is the record number. As an example, [3 : 11717] is the database address of the record located at slot number 11717 in file number 3.

Repeating data items in arrayed data fields are indicated with a d\* prefix where d is the decimal number of repetitions and "\*" indicates a repeating field.

The following table lists some examples of the above requirements.

**Table 8-1. Example Data Field Contents**

Field Spec	Display
<code>char str[255];</code>	This needs to be scrolled to see the >
<code>int array[10];</code>	>12,15,312,74,5,5*0<
<code>struct {   char name[15];   int age; } children[7];</code>	>{"Joelle",6},{"Rebecca",4},{"Scott",2},{"K& >
<code>DB_ADDR tree[2];</code>	>[2:1377],[2:1381]<
<code>char lines[10][10]</code>	>line 1\n\0\0\0line 2\n\0\0\0line3\n\>

## 3.4 Using Edit Modes and Field Edit Commands

The current editing mode is shown in the upper right corner of the screen. There are three modes:

*Command mode:* Indicates that the field edit commands are enabled. (Table 8-2 lists edit commands available in Command mode.)  
*Insert mode:* Initiated by the i, a, and A commands. Allows data to be inserted into the field. Pressing <Esc> returns to Command mode.

*Replace mode:* R command. Allows a type-over capability, where characters under the cursor are replaced by those typed at the keyboard. Pressing <Esc> returns to Command mode.

**Table 8-2. ida Field Edit Commands and Descriptions**

Command	Description
i	Insert characters before the cursor.
a	Append characters after the cursor.
A	Append characters to the end of the field.
r	Replace one character.
R	Replace (types over) existing characters.

h	Move the cursor one character to the left.
l	Move the cursor one character to the right.
w	Move the cursor forward one "small" word, delimited by anything other than letters, digits, or underscores.
W	Move cursor forward one "big" word, delimited by spaces.
b	Move the cursor back one small word.
B	Move the cursor back one big word.
^	Position the cursor at the beginning of the field.
\$	Position the cursor at the end of the field.
x	Delete the character beneath the cursor.
u	Undo the last change.
f<c>	Find the next occurrence of character <c> to the right.
F<c>	Find the next occurrence of character <c> to the left.

Additionally, some field edit commands can be preceded by a command repetition number. These commands are: h, l, w, W, b, B, x, f, and F. For example typing 30h moves the cursor 30 characters to the left.

Table 8-3 lists the function keys (if available on your keyboard), that have been assigned the above edit commands.

Command	Function Key
h	<Left Arrow>
l	<Right Arrow>
^	<Home>
\$	<End>
i	<Ins>
x	<Del>
u	<F1>
;	<F2>
W	<F3>
w	<F4>
B	<F5>
b	<F6>
R	<F7>
r	<F8>
A	<F9>
a	<F10>

# Using the Access Menu Commands

This section describes all of the database access commands that are available when you select the Access command from the main menu, shown in Figure 8-1. Selecting the Access command from the main menu accesses the `Database Access Commands` submenu, shown earlier in Figure 8-3.

The commands on `Database Access Commands` submenu are described in the following sections of this chapter:

- Record command
- Set command
- Currency command
- Transaction command
- Lock command
- Free command
- Miscellaneous command

## 4.1 Using the Access Record Commands

The `Record` command, selected from the `Database Access Commands` submenu, accesses the `Record Manipulation Functions` submenu, from which you can select a set of additional commands and submenus for managing records. This submenu is shown in Figure 8-9, followed by a description of each of its commands and their submenus.

```
Record Manipulation Functions
Keyscan Get Enter Modify Delete Recscan X_exit
```

Fig. 8-9. Record Manipulation Functions Submenu

### 4.1.1 Keyscan command

Selecting the `Keyscan` command accesses the `Scan and View Record Based on Key` submenu, from which you can choose how to locate a record based on a key. This submenu is shown in Figure 8-10, followed by a description of its commands.

```
Scan and View Record Based on Key
Scan Keyfind First Last Next Previous X_exit
```

Fig. 8-10. Scan and View Record Based on Key Submenu

### Scan

Scans records based on a key field. A list of all key field names will be displayed, from which you should select the field to be scanned. You will then be prompted to enter the value of the key where the scan will begin. If you press "Enter" instead of a key value, the scanning will begin from the first key.

The records located by the scan will be paged and displayed in a horizontal packed format. This is similar to the format that would result if the struct variable corresponding to the record were statically initialized in a C program. That is, it would be enclosed in braces, {}, with each sub-field grouping separated by commas. For more information about the display of various RDM Embedded data types, see [Editing a Data Field](#).

Figure 8-11 shows a sample record scan display.

```
Record: EMPLOYEE          Key: EMP_ID          Page: 1

ENTRY    RECORD CONTENTS
 1.      {"7901",{"Pfeil","Carl","L."},"MGR1"}
 2.      {"7902",{"Rogers","Jim",""}, "MGR1"}
 3.      {"7903",{"Fleagle","Jenifer","C."},"MGR2"}
 4.      {"7904",{"Smith","Stephen","P."},"MGR1"}
 5.      {"7906",{"Warren","Wayne","L."},"MGR1"}
 6.      {"7907",{"Merilatt","Randall","L."},"MGR1"}
 7.      {"8001",{"Bryant","John","W."},"MGR2"}
 8.      {"8002",{"Willis","Kirt","L."},"MGR2"}
 9.      {"8003",{"Charles","Thomas",""}, "ENGR"}
10.      {"8101",{"Temple","Anita","J."}, "ADMN"}
11.      {"8102",{"Rodovsky","Eliska",""}, "ENGR"}
12.      {"8102",{"Marsh","Kyle","G."}, "ENGR"}
13.      {"8103",{"Carlson","Bob","C."}, "MGR1"}
14.      {"8104",{"Bustamante","Tina","L."}, "ENGR"}
```

Fig. 8-11. A Sample Keyscan/Scan Display

### Keyfind

Quickly finds and displays one record. When you select a key field from the displayed list and enter a value, this command displays either the record associated with the specified key value, or if the key is not found, the record associated with the next key. Compound key and struct fields must be enclosed by braces ({}). Character strings do not need to be delimited by quotation marks except when included as part of a struct field.

### First

Finds and displays the record associated with the first key in the key file.

### Last

Finds and displays the record associated with the last key in the key file.

### Next

Finds and displays the record associated with the next key in the key file. If a Next command causes a "no more keys" message at the end of a file, a subsequent Next command will return the first key.

### Previous

Finds and displays the record associated with the previous key in the key file. If a Previous command causes a "no more keys" message at the end of a file, the next Previous command will return the last key.

### 4.1.2 Get command

```
Displays the record located at a specified database address.  
When prompted for the database address, enter it as:
```

```
enter database address: [f:r]
```

where *f* is the file number and *r* is the record number.

### 4.1.3 Enter command

Enters new records into the database. From the displayed list, select the record type to be entered. The `Display/Edit Record` submenu is then invoked. "Using the Access Record Commands" shows the `Display/Edit Record` submenu and describes its functions in detail.

### 4.1.4 Modify command

Invokes the `Display/Edit Record` submenu, which provides a set of tools for modifying the current record, which may be selected in a variety of ways. For detailed information, about the `Display/Edit Record` submenu, see [Using the Display/Edit Record Submenu Commands](#).

### 4.1.5 Delete command

Deletes the current record from the database. The record must have been previously disconnected from all sets of which it was an owner or a member.

### 4.1.6 Recscan command

Invokes the `Scan and View Record Based on Database Address` command submenu. The commands in this submenu use the `d_recfirst`, `d_recnext`, `d_reclast` and `d_recprev` functions to locate the records, instead of the key traversal functions.

```
Scan and View Record Based on Database Address
Scan First Last Next Previous X_exit
```

Fig. 8-12. Scan and View Record Based on Database Address Submenu

### Scan

Scans records based on their database address (their order is determined by their physical appearance in a data file). A list of all record type names will be displayed, from which you should select the type to be scanned.

The records located by the scan will be paged and displayed in the same horizontal packed format as the `Keys-can` command. For more information about the display of various RDM Embedded data types, see [Editing a Data Field](#).

### First

Finds and displays the first record of the selected type in the data file.

### Last

Finds and displays the last record of the selected type in the data file.

### Next

Finds and displays the next record in the data file. When it reaches the end of the file, it displays a "no more records" message. A subsequent `Next` command will begin with the first record.

### Previous

Finds and displays the previous record in the data file. When it reaches the top of the file, it displays a "no more records" message. A subsequent `Previous` command will begin with the last record.

## 4.2 Using the Access Set Commands

Selecting the `Set` command from the `Database Access Commands` submenu, accesses the `Set Manipulation Functions` submenu, which provides a set of tools for managing sets. This submenu is shown in Figure 8-13, followed by a description of its functions.

```
Set Manipulation Functions
Scan Owner First Next Last Previous Connect Disconnect Total X_exit
```

### 13. Set Manipulation Functions Submenu

### Scan command

Scans and displays all member records connected to the current owner of a set. The set to be scanned is selected from the displayed list of set names.

The records located by the scan will be paged and displayed in a horizontal packed format. This is similar to the format that would result if the struct variable corresponding to the record were statically initialized in a C program. That is, it would be enclosed in braces, { }, with each sub-field grouping separated by commas. For more information about the display of various RDM Embedded data types, see [Editing a Data Field](#).

### Owner command

Finds the owner of the current record, through the set selected from the displayed list of set names. Note that this function serves a different purpose than the Owner function under the Display/Edit Record submenu.

### First command

Finds the first member connected to the current owner of the set selected from the displayed list of set names.

### Next command

Finds the next member of a set. The first time (or after a "no more records" message), Next displays the first member of the set.

### Last command

Finds the last member connected to the current owner of the set selected from the displayed list of set names.

### Previous command

Finds the previous member of a set. The first time (or after a "no more records" message), Previous displays the last member of the set. A Last or Next command must have preceded the call to Previous.

### Connect command

Connects the current record to the current owner of the set selected from the displayed list of set names.

### Disconnect command

Disconnects the current record from the set selected from the displayed list of set names.

### Total command

Displays a count of the members connected to the current owner of the set selected from the displayed list of set names.

## 4.3 Using the Access Currency Commands

The `Currency` command, selected from the `Database Access Commands` submenu, accesses the `Currency Table Manipulation Functions` submenu (shown in Figure 8-14 and followed by a description of its functions).

```
Currency Table Manipulation Functions
Owner Member Record Auto_set Display X_exit
```

Fig. 8-14. Currency Table Manipulation Functions Submenu

### 4.3.1 Owner command

Sets the current owner of the set selected from the displayed list of set names. This function then invokes the `Change Current Owner of Set` submenu, from which you select the record you wish to be made current. The submenu is shown in Figure 8-15, followed by a description of its functions.

```
Change Current Owner of Set
Record Owner Member Change X_exit
```

Fig. 8-15. Change Current Owner of Set Submenu

#### Record

Changes the current owner of the selected set to the current record.

#### Owner

Changes the current owner of the previously selected set to the current owner of the set selected from the displayed list of sets.

#### Member

Changes the current owner of the previously selected set to the current member of the set selected from the displayed list of sets.

#### Change

Changes the current owner of the selected set to the database address entered in response to the prompt:

```
enter database address: [f :r]
```

where *f* is the file number and *r* is the record number.

### 4.3.2 Member command

Sets the current member of the set selected from the displayed list of set names. It then invokes the `Change Current Member of Set` submenu, shown in Figure 8-16, to determine the record to be made current.

```
Change Current Member of Set
Record Owner Member Change X_exit
```

Fig. 8-16. Change Current Member of Set Submenu

#### Record

Changes the current member of the selected set to the current record.

#### Owner

Changes the current member of the previously selected set to the current owner of the set selected from the displayed list of sets.

#### Member

Changes the current member of the previously selected set to the current member of the set selected from the displayed list of sets.

#### Change

Changes the current member of the selected set to the database address entered in response to the prompt:

```
enter database address: [f:r]
```

where *f* is the file number and *r* is the record number.

### 4.3.3 Record command

Changes the current record. This selection invokes the `Change Current Record` submenu, shown in Figure 8-17, to determine the record to be made current.

```
Change Current Record
Owner Member Change X_exit
```

Fig. 8-17. Change Current Record Submenu

### Owner

Changes the current record to the current owner of the set selected from the displayed list of set names.

### Member

Changes the current record to the current member of the set selected from the displayed list of set names.

### Change

Changes the current record to the database address entered in response to the prompt:

```
enter database address: [f:r]
```

where *f* is the file number and *r* is the record number.

### 4.3.4 Auto\_set command

Toggles the automatic set connection flag within `ida`. When turned on, this flag causes newly entered records to be connected automatically to the current owners of the sets for which they are members.

### 4.3.5 Display command

Displays the currency tables showing the record type and database addresses of the current record and, for each set, the current owner and current member of that set. It also indicates whether automatic set connection is enabled.

## 4.4 Using the Access Transaction Commands

The Transaction command, selected from the Database Access Commands submenu, controls the initiation and termination of transactions. It invokes the Transaction Processing Functions submenu, shown below, from which a transaction can be started (Begin), ended (End), or aborted (Abort).

```
Transaction Processing Functions  
Begin End Abort X_exit
```

Fig. 8-19. Transaction Processing Functions Submenu

### Begin command

Starts a transaction. You are prompted to enter the transaction identifier.

### End command

Stops a transaction. It applies to the database files all changes made since the beginning of the transaction.

### Abort command

Aborts the current transaction. It discards all changes made since the beginning of the transaction.

## 4.5 Using the Access Lock Commands

The `Lock` command, selected from the `Database Access Commands` submenu, is used to lock set types, record types, and key types. It invokes the `Multiuser Set/Record Lock Functions` submenu, shown in Fig. 8-20, from which you select either record, set or key locks. You then select the record, set, or key to be locked from the displayed list of record, set, or key names defined in the database.

```
Multiuser Set/Record Lock Functions
Record Set Key Timeout Display X_exit
```

Fig. 8-20. Multiuser Set/Record Lock Functions Submenu

### Record command

Places a lock on the record type selected from the displayed list of records. A selection list of lock types will be displayed, from which you should select the desired lock type. This command can only be executed when the database is opened for shared access.

### Set command

Places a lock on the set type selected from the displayed list of sets. A selection list of lock types will be displayed, from which you should select the desired lock type. This command can only be executed when the database is opened for shared access.

### Key command

Places a lock on the key type selected from the displayed list of keys. A selection list of lock types will be displayed, from which you should select the desired lock type. This command can only be executed when the database is opened for shared access.

### Timeout command

Sets the lock timeout value. Default value is 10 seconds.

### Display command

Displays the lock status for each set, record, and key type, indicating whether it is read-locked, read-locked with keep, write-locked, exclusively locked, or free. This command can only be executed when the database is opened for shared access.

## 4.6 Using the Access Free Commands

The `Free` command, selected from the `Database Access Commands` submenu, is used to free locked record, set, and key types. `Free` invokes the `Multi-user Set/Record Free Locks Functions` submenu, shown in Figure 8-21. From here you select which locks you want freed. If you are freeing a set, record, or key lock, select the correct one from the displayed list of names defined in the database.

```
Multi-user Set/Record Free Locks Functions
Record Set Key All X_exit
```

Fig. 8-21. Multi-user Set/Record Free Locks Functions Submenu

### Record command

Frees the lock for the record type selected from the displayed list of record types. This command can only be executed when the database is opened for shared access and when no transaction is active.

### Set command

Frees the lock for the set type selected from the displayed list of set types. This command can only be executed when the database is opened for shared access and when no transaction is active.

### Key command

Frees the lock for the key type selected from the displayed list of key types. This command can only be executed when the database is opened for shared access and when no transaction is active.

### All command

Frees all set, record and key locks. It can only be executed when the database is opened for shared access and when no transaction is active.

## 4.7 Using the Access Miscellaneous Commands

The `Miscellaneous` command, selected from the `Database Access Commands` submenu, invokes the `Miscellaneous ida Functions` submenu, shown in Figure 8-22, for displaying current database file and lock statuses.

```
Miscellaneous ida Functions
Files  Locks  X_exit
```

Fig. 8-22. Miscellaneous ida Functions Submenu

### Files command

Displays status information pertaining to the database and all database files, as shown in Table 8-4.

Table 8-4. File Status Information (From the Misc Files Command)

Item Name	Description
Database name	The name of the database currently being accessed.
Access type	Exclusive, shared, or one_user access.
Transaction	The identifier of the active transaction.
FILE	The file number of the listed file.
SLOTS	The total number of allocated file slots (if data file) or pages (if key file).
TYPE	The file type: "data", "vardata", "blob" or "key".
STATUS	File status: "opened" or "closed", "locked" or "free."
RW	The total number of read locks on the file within this application. A file may be locked from different set/record locks. This value gives the current total. It is displayed only when the database is opened for shared access. A value of -1 indicates that the file is write-locked.
EX	The total number of exclusive locks on the file. A file may be exclusively locked from different set/record locks. This value gives the current total. It is displayed only when the database is opened for shared access.
NAME	The name of the file including any current path names.

### Locks command

Displays the lock status for each set, record, and key type, indicating whether it is read-locked, read-locked with keep, write-locked, exclusively locked, or free. This command can only be executed when the database is opened for shared access.

# RDM Embedded Functions Cross-Referenced to ida Commands

The `ida` utility allows you to use a menu-driven interface instead of entering RDM Embedded functions from the command line. Table 8-5 lists RDM Embedded functions and the corresponding sequence of menu commands that produce the same result. This list is helpful when learning how to use the various RDM Embedded functions. For complete information on the operation of a particular RDM Embedded C function, refer to the detailed function descriptions in the RDM Embedded Reference Manual.

In the following list, functions are grouped by type of operation. The function name is in the left column, and the menu command sequence is in the right column.

Table 8-5. RDM Embedded Functions and Corresponding Command Sequences

## Database Administration Functions

Table 8-5. RDM Embedded Functions and Corresponding Command Sequences

Function	Menu Command Sequence
<code>d_open</code>	Open
<code>d_close</code>	Close
<code>d_initialize</code>	Initialize
<code>d_initfile</code>	no corresponding sequence
<code>d_destroy</code>	no corresponding sequence

## Record Manipulation Functions

Function	Menu Command Sequence
<code>d_recread</code>	Access Record Modify
<code>d_recwrite</code>	Access Record Modify Write
<code>d_fillnew</code>	Access Record Enter Write
<code>d_makenew</code>	no corresponding sequence
<code>d_setkey</code>	no corresponding sequence
<code>d_delete</code>	Access Record Delete
<code>d_keystore</code>	Access Record Enter/Modify Store_key
<code>d_keydel</code>	Access Record Enter/Modify Delete_key

## Record Navigation Functions

Function	Menu Command Sequence
<code>d_recfrst</code>	Access Record Recscan First
<code>d_recnext</code>	Access Record Recscan Next
<code>d_recprev</code>	Access Record Recscan Previous
<code>d_reclast</code>	Access Record Recscan Last

## Set Manipulation Functions

Function	Menu Command Sequence
<code>d_connect</code>	Access Set Connect
<code>d_discon</code>	Access Set Discon
<code>d_members</code>	Access Set Total
<code>d_isowner</code>	no corresponding sequence
<code>d_ismember</code>	no corresponding sequence

## Set Navigation Functions

Function	Menu Command Sequence
<code>d_findco</code>	Access Set Owner
<code>d_findfm</code>	Access Set First
<code>d_findlm</code>	Access Set Last
<code>d_findnm</code>	Access Set Next
<code>d_findpm</code>	Access Set Prev

## Key Field Access Functions

Function	Menu Command Sequence
<code>d_keyfirst</code>	Access Record Keyscan First
<code>d_keylast</code>	Access Record Keyscan Last
<code>d_keyfind</code>	Access Record Keyscan Keyfind
<code>d_keynext</code>	Access Record Keyscan Next
<code>d_keyprev</code>	Access Record Keyscan Previous
<code>d_keyread</code>	no corresponding sequence

## Multi-User Database Control Functions

Function	Menu Command Sequence
<code>d_trbegin</code>	Access Transaction Begin
<code>d_trend</code>	Access Transaction End
<code>d_trabort</code>	Access Transaction Abort
<code>d_reclock</code>	Access Lock Record
<code>d_refree</code>	Access Free Record
<code>d_setlock</code>	Access Lock Set
<code>d_setfree</code>	Access Free Set
<code>d_keylock</code>	Access Lock Key
<code>d_keyfree</code>	Access Free Key
<code>d_freeall</code>	Access Free All

## Currency Control Functions

Function	Menu Command Sequence
<a href="#">d_setro</a>	Access Currency Record Owner
<a href="#">d_setrm</a>	Access Currency Record Member
<a href="#">d_setor</a>	Access Currency Owner Record
<a href="#">d_setom</a>	Access Currency Owner Member
<a href="#">d_setoo</a>	Access Currency Owner Owner
<a href="#">d_setmr</a>	Access Currency Member Record
<a href="#">d_setmo</a>	Access Currency Member Owner
<a href="#">d_setmm</a>	Access Currency Member Member
<a href="#">d_crget</a>	no corresponding sequence
<a href="#">d_crset</a>	Access Currency Record Change
<a href="#">d_csmget</a>	no corresponding sequence
<a href="#">d_csmset</a>	Access Currency Member Change
<a href="#">d_csoget</a>	no corresponding sequence
<a href="#">d_csoset</a>	Access Currency Owner Change

## Miscellaneous Control Functions

Function	Menu Command Sequence
<a href="#">d_crtype</a>	no corresponding sequence
<a href="#">d_cmttype</a>	no corresponding sequence
<a href="#">d_cotype</a>	no corresponding sequence