

# **MDBS Data Base Restructuring System**

**- MDBS DBRS MANUAL -**

**Version 3.08**

**Micro Data Base Systems, Inc.**

**P.O. Box 248**

**Lafayette, Indiana 47902**

**USA**

**December 1985**

## **Copyright Notice**

This entire manual is provided for the use of the customer and the customer's employees. The entire contents have been copyrighted by Micro Data Base Systems, Inc., and reproduction by any means is prohibited except as permitted in a written agreement with Micro Data Base Systems, Inc.

## NEW RELEASES, VERSIONS, AND A WARNING

Any programming endeavor of the magnitude of the MDBS software will necessarily continue to evolve over time. Realizing this, Micro Data Base Systems, Inc., vows to provide its users with updates to this version for a nominal handling fee.

New versions of MDBS software will be considered as separate products. However, bona fide owners of previous versions are generally entitled to a preferential rate structure.

Finally, each copy of our software is personalized to identify the licensee. There are several levels of this personalization, some of which involve encryption methods guaranteed to be combinatorially difficult to decipher. Our products have been produced with a very substantial investment of capital and labor, to say nothing of the years of prior involvement in the data base management area by our principals. Accordingly, we are seriously concerned about any unauthorized copying of our products and will take any and all available legal action against illegal copying or distribution of our products.

## DISCLAIMER NOTICE

All rights reserved. No part of this material shall be reproduced, stored in a retrieval system, or transmitted by any means, electronic, mechanical, photocopying, recording, or otherwise, without written permission from Micro Data Base Systems, Inc.

Although care has been taken in the preparation of this material, Micro Data Base Systems, Inc. assumes neither responsibility for errors or omissions, nor for damages resulting from the use of the information contained herein. Micro Data Base Systems, Inc. does not warrant its accuracy nor guarantee the operation of the system in every instance described herein.

The reader/user assumes full liability and all risk for any damages resulting from the use of the information contained herein, and for determining whether the information contained herein is suitable for user's intended purpose.

Micro Data Base Systems, Inc. reserves the right to incorporate design improvements and new functions in its software products and software systems. Recent improvements may not always be reflected in documentation.

TABLE OF CONTENTS

	<u>Page</u>
I. OVERVIEW . . . . .	.1
A. Introduction. . . . .	.1
B. Three Stages of Data Base Restructuring . . . . .	.2
C. Organization of This Manual . . . . .	.2
II. SCHEMA ALTERATION. . . . .	.5
A. Invoking the DBRS Analyzer. . . . .	.6
B. Area Alterations. . . . .	.7
C. Record Type Alterations . . . . .	11
D. Set Alterations . . . . .	22
E. Path Specifications . . . . .	30
F. Interactive DBRS Analyzer Diagnostics . . . . .	32
G. Fatal Errors . . . . .	38
III. DECOMPOSITION. . . . .	41
A. Data Value Transformations. . . . .	41
B. Record Relationship Transformations . . . . .	43
C. Effect on Screen Dictionary . . . . .	44
D. The Decomposition Process . . . . .	44
E. Decomposition Diagnostics . . . . .	46
IV. REGENERATION . . . . .	49
A. The Regeneration Process. . . . .	49
B. Regeneration Diagnostics. . . . .	50
Figure I-1 The Three Stages of Data Base Restructuring for a Data Based Named JOBS. . . . .	.3
Figure II-1 Schema for the JOBS Data Base . . . . .	31
Table III-1. Effects of Type Changes on Data Item Values . . . . .	42

## I. OVERVIEW

## A. Introduction

The Data Base Restructuring System is designed for use with MDBS III data bases. Called DBRS, this MDBS utility system allows the schema of any existing MDBS data base to be interactively altered. The data dictionary and data base itself are then automatically revised to be consistent with the new schema and to maximize storage/processing efficiencies within the constraints imposed by the schema. DBRS can also be used by a data base administrator, independent of schema alterations, to automatically transform the physical structure of an existing data base into a different physical structure that optimizes space utilization and data access speeds.

When a schema is formally specified with the Data Description Language (DDL), some information about physical structuring is stated in addition to complete information about the data base's logical structure. This physical structuring information (page sizes and quantities) is specified in the Identification Section and Area Section. Additionally, much of the logical structuring information has physical implications. However, unlike the logical structure, a data base's physical structure is not totally dictated by the schema. The physical structure is continually changing as records (and their relationships) are created and deleted. And it is strongly influenced by the order in which creations and deletions happen. Thus, two data bases with identical schemas and identical data/relationships can physically be substantially different, to such a degree that one outperforms the other.

Restructuring performed by DBRS is based not only on schema alterations, but also on the data/relationships that currently exist in a data base. These are physically reorganized to consolidate free space fragmentation, achieve a more concentrated clustering (where applicable), reduce CALCed record overflows (where applicable), optimize access speeds through restructured indexes, and so forth. Therefore, DBRS can be used to enhance performance for a mature data base as well as to accommodate schema alterations.

The kinds of schema alterations that can be made with DBRS include:

- Adding and deleting areas.
- Changing an area's characteristics (name, pages, page size, file, etc.).
- Adding and deleting record types.
- Changing a record type's location method.
- Adding and deleting data items for a record type.
- Changing a data item's characteristics.
- Adding and deleting named relationships (sets) among record types.
- Changing a set's characteristics.

## B. Three Stages of Data Base Restructuring

For operational convenience, DBRS usage is organized into three distinct stages. See Figure I-1. The first stage is one of indicating desired schema alterations. This is accomplished by responding to prompts issued by the DBRS Analyzer. This stage should not take place in a multiuser setting. The DBRS Analyzer checks the proposed changes to assure that they do not violate standard schema conventions. It does not affect the data base. If no errors are detected, this stage results in a file containing a valid revised schema description.\*

The second stage carries out a systematic decomposition of the data base. It utilizes the revised schema description (if any) from the first stage together with a copy of the data base to produce a decomposition file. This file contains all information needed to regenerate the data base so that it is consistent with the new schema. Certain aspects of physical structuring optimization occur during the decomposition stage. The third stage uses the decomposition file to actually regenerate the data base.

After a data base has been restructured with DBRS, it is no longer compatible with the obsolete MDBS I DML commands (see Appendix A of the MDBS III DMS Manual). These obsolete commands should not be used in subsequent processing of the data base.

## C. Organization of This Manual

Detailed discussion of the three stages and how to utilize each is provided in the three chapters that follow. Diagnostics pertaining to a particular stage are described in the chapter dealing with that stage.

---

\* When revising a data base that is used with Screen Master, the Screen Master portion of the schema is discarded. To re-incorporate it, the SDL Analyzer should be used after the third stage of restructuring is finished.

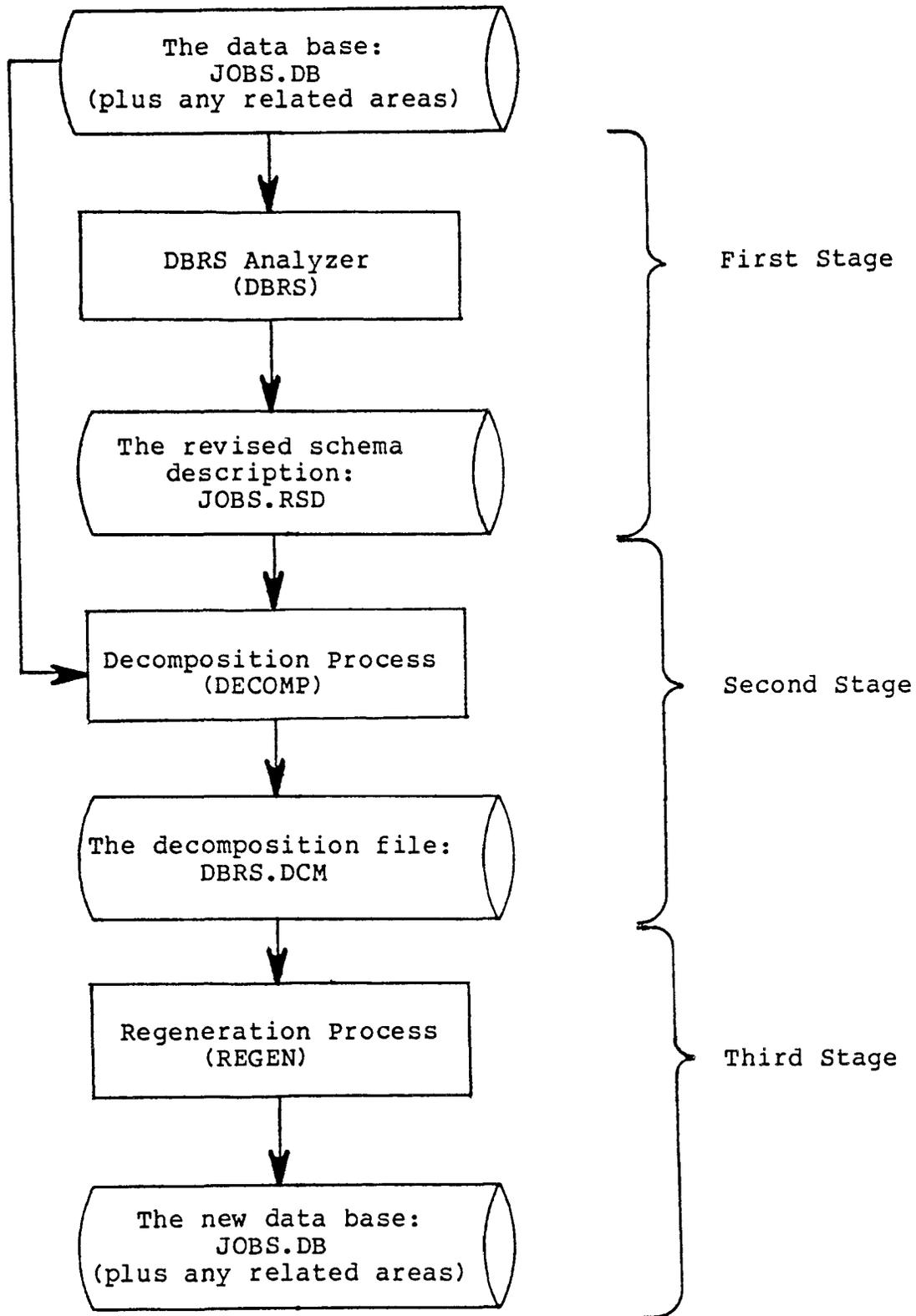


Figure I-1. The Three Stages of Data Base Restructuring for a Data Base Named JOBS

## II. SCHEMA ALTERATION

The DBRS Analyzer is an interactive program that allows its user to specify the nature of desired schema changes, analyzes proposed changes to ensure their correctness, and produces a file containing the revised schema description. Desired schema changes are specified by making menu selections and responding to prompts issued by the Analyzer. As each alteration is proposed, it is checked for conformance to standard schema definition conventions (as specified in the MDBS DDL Manual) and for consistency with the rest of the schema. If the Analyzer detects an error, a descriptive diagnostic is issued and the proposed alteration can be corrected interactively. Explanations of interactive diagnostics appear in Section F of this chapter. The DBRS Analyzer will refuse to accept incorrect alterations to a schema.

As each correct alteration is specified, the DBRS Analyzer logs it on a "selective automatic input" file. This file's name will default to be the same as the data base name, but with an SAI extension. An SAI file can later be used as input to the Analyzer to re-create the effect of the present schema alteration session. This is useful in situations where a session needs to be temporarily discontinued and later resumed. By using the SAI file as input to the later session, all of the work in the prior session is re-created automatically without requiring re-typing of the discontinued session's alterations. As it processes an SAI file in this way the DBRS Analyzer pauses at appropriate places (after the area alterations, after the record type alterations, etc.) and gives the user a chance to selectively make further revisions (to the area, record types, etc.). This continues up through the last alteration that had been logged into the SAI file. The DBRS Analyzer then reverts back to its usual interactive behavior, so the user can resume where the prior session had been discontinued.

When all desired alterations have been specified in a session, the DBRS Analyzer will make a global check of the alterations. Any errors it detects at that time are fatal. If such an error has occurred, the Analyzer will report it on the screen. The possible fatal errors are described in Section G of this chapter. The DBRS Analyzer will need to be re-run in order to correct such an error. This can be expedited by using the SAI file and making needed changes to the original alterations at the appropriate pauses.

Upon exiting from the Analyzer, a revised schema description is automatically written to a file on the default drive. This file's name will be the same as the data base name, but with an RSD extension. This file will not be usable if a fatal error was detected.

## A. Invoking the DBRS Analyzer

The DBRS Analyzer is supplied as an executable program and is normally invoked by entering

DBRS

on the operating system's command line. The user sees the DBRS Analyzer's sign-on message and is then prompted to state the name of the file containing the main area of the data base whose schema is to be altered. The file name can be qualified with a drive/directory indicator and can have an extension. Prompts are then issued for a valid user name and password. When these have been supplied, the Analyzer begins asking for schema alterations.

When DBRS is invoked, any of the following optional arguments can appear on the command line:

- Bnnnn            where nnnn is the number of bytes to be allocated for the buffer used by the DBRS Analyzer for data base interaction. If this argument is not used, approximately half of free memory is allocated for this buffer.
- Dfilename        where filename is the name of the file containing the main area of the data base whose schema is being altered. If this argument appears, the Analyzer will not prompt for data base identification.
- E                is pertinent only if the -I argument is present. When -E is used, all alterations in the SAI file are echoed to the screen as they are processed by the Analyzer.
- F                is pertinent only if the -I argument is present. When -F is used, all alterations in the SAI file are forced through the Analyzer without any pauses that allow additional selective alterations to the schema.
- Ifilename        where filename is the name of an SAI file that the Analyzer had produced in a prior session. The alterations indicated in this file will be processed automatically by the Analyzer. If the -I argument is specified without filename, then an SAI file having the same name as the data base will be processed (if it exists).
- Ofilename        where filename is an alternative name for the selective automatic input file. The alterations will be logged to this alternative file instead of the default formed from the data base name plus an SAI extension.

- Ppass where pass is the user password that is to be used by the Analyzer in attempting to open the data base. If this argument appears, the Analyzer will not prompt for a password.
- Rfilename where filename is the name of a file that will receive the revised schema description. If this argument does not appear, then the revised schema description will be saved on a file having the same name as the data base plus an RSD extension.
- Username where username identifies a valid data base user. If this argument appears, the Analyzer will not prompt for user identification.

These optional arguments can appear in any order following DBRS on the command line.

## B. Area Alterations

The first kind of schema alterations that can be made concern areas (both the main area and any extra areas). The following menu appears:

```
Area Section:
(A) Add new area
(C) Change existing area
(D) Delete existing area
(R) Rename area
(S) Schema display
(E) Exit to Record Section
```

Selection?

When any option other than E is selected, the user is prompted to supply appropriate kinds of information. When processing is completed for that option, the Area Section menu reappears. When all desired alterations to areas have been made, the E option can be selected to proceed to record type alterations.

1. If option A is selected, a series of prompts appear. These ask the user to indicate the characteristics of the new area.

```
Area name?
File name? [ ]
Size of area (pages)? [ ]
Pointers allowed? [ ]
Title?
Synonym?
Read access codes? [ ]
Write access codes? [ ]
```

Some prompts include a default response enclosed in square brackets. If the user presses the Return (Enter) key for such a prompt, then the default in brackets is used as the response. Alternatively, the user can directly type in the desired response to each prompt. The end of a response is signaled by pressing the Return key. Area name has no default response and must be explicitly entered by the user. Simply pressing Return for Title or Synonym results in no title or synonym for this area. If an invalid response is made to a prompt, the response is ignored and the prompt reappears.

After all prompts have been answered, a summary of the new area's characteristics is displayed and the Analyzer asks

OK? [yes]

If the user desires to make a change to the new area's characteristics, then the response to this question should be: no or n. As a result, the user gets an opportunity to change each of the area characteristics.

2. If option C is selected, the user is first asked to indicate which area needs to be changed.

Area name?

The user can type in the name of an existing area. If a ? or a non-existent area is specified, a list of all existing areas is displayed and the user is again prompted to enter the name of the area whose characteristics are to be changed. When an existing area name has been specified, a series of prompts appear. These ask the user to indicate the new characteristics of the area.

File name? [ ]  
Size of area (pages)? [ ]  
Page size (bytes)? [ ]  
Pointers allowed? [ ]  
Title? [ ]  
Synonym? [ ]  
Read access codes? [ ]  
Write access codes? [ ]

The present nature of each characteristic is shown in square brackets. If no change is desired, the user merely presses the Return (Enter) key. An existing title or synonym can be eliminated by pressing the space bar and then the Return key. To make a change the user explicitly types in new value. The end of this response is signaled by pressing the Return key. If an invalid response is made to a prompt, the response is ignored and the prompt reappears.

After all prompts have been answered, a summary of the area's new characteristics is displayed and the Analyzer asks

OK? [yes]

If the user desires to make a further change to the area's characteristics, then the response to this question should be: no or n. As a result, the user again gets a chance to change each of the area characteristics. Entering y or pressing the Return key causes the Area Section menu to reappear.

3. If option D is selected, the user is asked to indicate which area is to be deleted.

Area name?

The user can type in the name of an existing area. Pressing the Return key without specifying a name returns the user to the Area Section menu and no deletion occurs. If a ? or a non-existent area is specified, a list of all existing areas is displayed and the user is again prompted to enter the name of the area to be deleted. When an existing area name has been supplied, it is irrevocably deleted from the schema and the Area Section menu reappears.

4. If option R is selected, the user is asked to indicate which existing area is to be renamed.

Old name?

The user can type in the name of an existing area. Pressing the Return key without specifying a name returns the user to the Area Section menu and no renaming occurs. If a ? or non-existent area is specified, a list of all existing areas is displayed and the user is again prompted to enter an area name. When an existing area name has been specified, the user is asked to specify the new name for the area.

New name?

When a valid name that does not duplicate an existing area's name has been specified, the area has a new name and the Area Section menu reappears.

5. If option S is selected, the Schema Display menu appears and the user can select any of its options to see a desired aspect of the schema.

Schema Display:

- (A) Area display
- (I) Item display
- (R) Record display
- (S) Set display
- (W) Write DDL file
- (E) Exit to path specification

Selection?

The portion of the schema that is displayed is entirely up to date, reflecting all schema alterations that have occurred in the present DBRS Analyzer session. When processing for any selected option (other than E) is completed, the Schema Display menu automatically reappears.

- a) If option A is selected, the user is prompted for an area name. When an existing area name is entered, that area's characteristics are displayed. Alternatively, the user can simply press the Return key to have the characteristics of all areas displayed. If the user enters a ? or a non-existent area name, a list of all area names is shown and the user is again prompted for an area name.
- b) If option I is selected, the user is prompted for a record type name. If the user simply presses the Return key, a list of all existing record types and all their respective data items is displayed. By responding with a ? or non-existent record type, the user is shown a list of record types only. In either case, the user is again prompted for a record type name. When this has been furnished, the user is prompted for the name of a data item in this record type.

When an existing item name is entered, the characteristics of that item are displayed. If instead the user simply presses the Return key, characteristics of all of the record type's items are displayed. If a ? or non-existent item is specified, a list of all of the record type's items is shown and the user is again prompted for an item name.

- c) If option R is selected, the user is prompted for a record type name. When an existing record type name is entered, that record type's characteristics are displayed. Alternatively, the user can simply press the Return key to have the characteristics of all record types displayed. If the user enters a ? or a non-existent record type name, a list of all record type names is shown and the user is again prompted for a record type name.
- d) If option S is selected, the user is prompted for a set name. When an existing set name is entered, that set's characteristics are displayed. Alternatively, the user can simply press the Return key to have the characteristics of all sets displayed. If the user enters a ? or a non-existent set name, a list of all set names is shown and the user is again prompted for a set name.

- e) If option W is selected, the user is prompted to indicate the name of a file on which a DDL specification for the current schema will be written. If the user presses the return key without specifying a file name, the DDL text is displayed on the console screen.
  - f) If option E is selected, the Area Section menu reappears.
6. If option E is selected, then no further area alterations can be made in the present DBRS Analyzer session. The user can now begin making alterations to the schema's record types.

### C. Record Type Alterations

The second kind of schema alterations that can be made concern the schema's record types (including their respective data items). The following menu appears:

#### Record Section:

- (A) Add new record type
- (C) Change existing record type
- (D) Delete existing record type
- (R) Rename record type
- (S) Schema display
- (E) Exit to Set Section

Selection?

When any option other than E is selected, the user is prompted to supply appropriate kinds of information. When processing is completed for that option, the Record Section menu reappears. When all desired alterations to record types have been made, the E option can be selected to proceed to set alterations.

1. If option A is selected, the user is first asked to specify the new record type's name. If the Return key is pressed before a name is entered, the user is returned to the Record Section menu. Once a name is entered, a series of prompts then appears to determine the record type's location mode (clustered, in specific areas, CALCed). The exact prompts that appear in this series depends on how the user responds to the prompts. For instance, if clustering is selected, then prompts for area placements and CALC key specifications will not appear. The default response for any prompt is indicated in square brackets. Simply pressing the Return (Enter) key results in the default response. If some other response is desired, the user explicitly types it in and presses Return to signal that the response is complete. If an invalid response is made, the prompt reappears.

Once the new record type's location mode is specified, Title and Synonym prompts appear. If no title is desired, simply press Return for the Title prompt. When all desired synonyms have been entered, simply press the Return key for the Synonym prompt.

Now the Item Section menu appears to allow the user to specify the new record type's data items.

Item Section for record type rrrrrrrr:

- (A) Add new item
- (C) Change existing item
- (D) Deleting existing item
- (M) Move item
- (R) Rename item
- (S) Schema display
- (E) Exit

Selection?

When any option other than E is selected, the user is prompted to supply appropriate information. When processing is completed for that option, the Item Section menu reappears. When all desired items have been specified for the new record type, the E option can be selected to cause the Record Section menu to reappear.

- a) If option A is selected, a series of prompts appear. These ask the user to indicate the characteristics of the new data item.

Item name?

Item type (Unsign,Int,Char,String,Bin,Real,idEc,Time,Date)?

Item length? [ ]

Number of repetitions? [ ]

Encrypted? [ ]

Title?

Synonym?

Read access? [ ]

Write access? [ ]

Check range? [ ]

Upper range value? [ ]

Lower range value? [ ]

Some of these prompts include a default response enclosed in square brackets. A default response can be selected by simply pressing the Return (Enter) key. Alternatively, the user can directly type in the desired response for each prompt. The end of a response is signaled by pressing the Return key. Item name has no default response and must be explicitly entered by the user. Simply pressing return for title results in no title. Synonyms are specified one per Synonym prompt. When no more synonyms remain to be declared, the Return key is pressed in response to the Synonym prompt. If an invalid response is made to a prompt, the response is ignored and the prompt reappears. Depending on the field type, some range prompts may not appear.

After all item prompts have been answered, a summary of the new data item's characteristics is displayed and the Analyzer asks

OK? [yes]

A response of no gives the user a chance to change the new item's characteristics. Alternatively, pressing the Return key causes the Item Section menu to reappear.

- b) If option C is selected, the user is first asked to indicate which of the new record type's items is to be changed.

Item name?

The user can type in the name of an existing item. Pressing the Return key without entering a name returns the user to the Item Section menu and no change is made. Alternatively, if a ? or the name of a non-existent item is specified, a list of all of the record type's items is shown and the prompt reappears. When a valid entry has been made, the user is asked to indicate the item's new characteristics.

Item type (Unsign,Int,Char,String,Bin,Real,idEc,Time,Date)? [ ]  
 Item length? [ ]  
 Number of repetitions? [ ]  
 Encrypted? [ ]  
 Title? [ ]  
 Synonym? [ ]  
 Read access? [ ]  
 Write access? [ ]  
 Check range? [ ]  
 Upper range value? [ ]  
 Lower range value? [ ]

Default responses shown in square brackets indicate the item's current characteristics. A default response can be selected by simply pressing the Return (Enter) key. Alternatively, the user can type in the new value of a characteristic. If an invalid response is made to a prompt, the response is ignored and the prompt reappears. Depending on the field type, some range prompts may not appear.

- c) If option D is selected, the user is asked to indicate which of the new record type's data items is to be deleted.

Item name?

The user can specify the name of an existing item. Pressing the Return key without entering a name returns the user to the Item Section menu and no item is deleted. If a ? or non-existent item is specified, then a list of all of the new record type's existing items is shown and the user is again prompted for the name of the item to be deleted. When an existing item name has been specified, it is deleted from the record type and the Item Section menu reappears.

- d) If option M is selected, the user is asked to indicate which item is to be moved to a new position in the record type.

Item name?

Pressing the Return key without entering a name returns the user to the Item Section menu and no item movement occurs. When the response is a ? or non-existent item, a list of all of the record type's items is displayed and the user is again prompted. When an existing item name has been specified, the user is prompted to indicate which item the "moved" item should follow in the record type. To move the item prior to all others in the record type, the user would simply press the Return key in response to this prompt. In any case, the desired item repositioning occurs and the Item Section menu reappears.

- e) If option R is selected, the user is asked to indicate which of the record type's items is to be renamed.

Old name?

When the response is a ? or non-existent item, a list of all of the record type's items is displayed and the user is again prompted. When an existing item name has been specified, the user is prompted to indicate the item's new name.

New name?

If the user responds with an invalid name (e.g., a name that already exists in the record type), then the response is ignored and the prompt reappears. When an acceptable new name has been specified, the Item Section menu reappears.

- f) If option S is selected, the Schema Display menu appears allowing the user to see any desired aspect of the schema.

Schema Display:

- (A) Area display
- (I) Item display
- (R) Record display
- (S) Set display
- (W) Write DDL file
- (E) Exit

Selection?

The portion of the schema selected for display is entirely up to date, reflecting all schema alterations that have occurred in the present DBRS Analyzer session. When an option is selected, the user may be prompted for an area, item, record type, or set name. The user can respond with a ? to see a list of all existing names. When processing for any selected option (other than E) is completed, the Schema Display menu automatically reappears. To return to the item Section menu, the user selects the E option.

- g) If option E in the Item Section menu is selected, the Record Section menu reappears.
2. If option C is selected, the user is first asked to indicate which record type needs to be changed.

Record name?

The user can type in the name of an existing record type. Pressing the Return key without entering a name returns the user to the Record Section menu and no change occurs. If a ? or a non-existent record type is specified, a list of all existing record types is displayed and the user is again prompted to enter the name of a record type whose characteristics are to be changed. When an existing record type has been specified, a series of prompts appear allowing the user to change this record type's location mode. The present location mode is indicated by the default responses shown in square brackets. The user presses the Return (Enter) key for these prompts, if no location mode change is desired.

Next, title and synonym prompts appear with present values (if any) shown in square brackets. If no title change is desired simply press the Return key. To eliminate an existing title, press the space bar and then the Return key. Otherwise, type in the new title. Similarly, if a synonym does not need to be changed, press the Return key. To eliminate it, press the space bar and then the Return key. Otherwise, type in the new synonym.

Now the Item Section menu appears to allow the user to change the record type's data items.

```
Item Section for record type rrrrrrrr:
(A) Add new item
(C) Change existing item
(D) Deleting existing item
(M) Move item
(R) Rename item
(S) Schema display
(E) Exit
```

Selection?

When any option other than E is selected, the user is prompted to supply appropriate information. When processing is completed for that option, the Item Section menu reappears. When all desired items have been specified for the existing record type, the E option can be selected to cause the Record Section menu to reappear.

- a) If option A is selected, a series of prompts appear. These ask the user to indicate the characteristics of the new data item.

```
Item name?
Item type (Unsign,Int,Char,String,Bin,Real,idEc,Time,Date)?
Item length? [ ]
Number of repetitions? [ ]
Encrypted? [ ]
Title?
Synonym?
Read access? [ ]
Write access? [ ]
Check range? [ ]
Upper range value? [ ]
Lower range value? [ ]
```

Some of these prompts include a default response enclosed in square brackets. A default response can be selected by simply pressing the Return (Enter) key. Alternatively, the user can directly type in the desired response for each prompt. The end of a response is signaled by pressing the Return key. Item name has no default response and must be explicitly entered by the user. Simply pressing return for title results in no title. Synonyms are specified one per Synonym prompt. When no more synonyms remain to be declared, the Return key is pressed in response to the Synonym prompt. If an invalid response is made to a prompt, the response is ignored and the prompt reappears. Depending on the field type, some range prompts may not appear.

After all item prompts have been answered, a summary of the new data item's characteristics is displayed and the Analyzer asks

OK? [yes]

A response of no gives the user a chance to change the new item's characteristics. Alternatively, pressing the Return key causes the Item Section menu to reappear.

- b) If option C is selected, the user is first asked to indicate which of the existing record type's items is to be changed.

Item name?

The user can type in the name of an existing item. Pressing the Return key without entering a name returns the user to the Item Section menu and no change is made. Alternatively, if a ? or the name of a non-existent item is specified, a list of all of the record type's items is shown and the prompt reappears. When a valid entry has been made, the user is asked to indicate the item's new characteristics.

Item type (Unsign,Int,Char,String,Bin,Real,idEc,Time,Date)? [ ]  
 Item length? [ ]  
 Number of repetitions? [ ]  
 Encrypted? [ ]  
 Title? [ ]  
 Synonym? [ ]  
 Read access? [ ]  
 Write access? [ ]  
 Check range? [ ]  
 Upper range value? [ ]  
 Lower range value? [ ]

Default responses shown in square brackets indicate the item's current characteristics. A default response can be selected by simply pressing the Return (Enter) key. Alternatively, the user can type in the new value of a characteristic. If an invalid response is made to a prompt, the response is ignored and the prompt reappears. Depending on the field type, some range prompts may not appear.

- c) If option D is selected, the user is asked to indicate which of the new record type's data items is to be deleted.

Item name?

The user can specify the name of an existing item. Pressing the Return key without entering a name returns the user to the Item Section menu and no item is deleted. If a ? or non-existent item is specified, then a list of all of the existing record type's existing items is shown and the user is again prompted for the name of the item to be deleted. When an existing item name has been specified, it is deleted from the record type and the Item Section menu reappears.

- d) If option M is selected, the user is asked to indicate which item is to be moved to a new position in the record type.

Item name?

Pressing the Return key without entering a name returns the user to the Item Section menu and no item movement occurs. When the response is a ? or non-existent item, a list of all of the record type's items is displayed and the user is again prompted. When an existing item name has been specified, the user is prompted to indicate the which item the "moved" item should follow in the record type. To move the item prior to all others in the record type, the user would simply press the Return key in response to this prompt. In any case, the desired item repositioning occurs and the Item Section menu reappears.

- e) If option R is selected, the user is asked to indicate which of the record type's items is to be renamed.

Old name?

When the response is a ? or non-existent item, a list of all of the record type's items is displayed and the user is again prompted. When an existing item name has been specified, the user is prompted to indicate the item's new name.

New name?

If the user responds with an invalid name (e.g., a name that already exists in the record type), then the response is ignored and the prompt reappears. When an acceptable new name has been specified, the Item Section menu reappears.

- f) If option S is selected, the Schema Display menu appears allowing the user to see any desired aspect of the schema.

Schema Display:

- (A) Area display
- (I) Item display
- (R) Record display
- (S) Set display
- (W) Write DDL file
- (E) Exit

Selection?

The portion of the schema selected for display is entirely up to date, reflecting all schema alterations that have occurred in the present DBRS Analyzer session. When an option is selected, the user may be prompted for an area, item, record type, or set name. The user can respond with a ? to see a list of all existing names. When processing for any selected option (other than E) is completed, the Schema Display menu automatically reappears. To return to the Item Section menu, the user selects the E option.

- g) If option E in the Item Section menu is selected, the Record Section menu reappears.
3. If option D is selected, the user is asked to indicate which record type is to be deleted.

Record type name?

The user can type in the name of an existing record type. Pressing the Return key without specifying a name returns the user to the Record Section menu and no deletion occurs. If a ? or a non-existent record type is specified, a list of all existing record types is displayed and the user is again prompted to enter the name of the area to be deleted. When an existing record type name has been specified, it is irrevocably deleted from the schema and the Record Type Section menu reappears.

4. If option R is selected, the user is asked to indicate which existing record type is to be renamed.

Old name?

The user can type in the name of an existing record type. Pressing the Return key without specifying a name returns the user to the Record Section menu and no renaming occurs. If a ? or non-existent record type is specified, a list of all existing record types is displayed and the user is again prompted to enter a record type name. When an existing record type name has been specified, the user is asked to specify the new name for the record type.

New name?

When a valid name that does not duplicate an existing record type's name has been specified, the record type has a new name and the Record Type Section menu reappears.

5. If option S is selected, the Schema Display menu appears and the user can select any of its options to see a desired aspect of the schema.

Schema Display:

- (A) Area display
- (I) Item display
- (R) Record display
- (S) Set display
- (W) Write DDL file
- (E) Exit to path specification

Selection?

The portion of the schema that is displayed is entirely up to date, reflecting all schema alterations that have occurred in the present DBRS Analyzer session. When processing for any selected option (other than E) is completed, the schema Display menu automatically reappears.

- a) If option A is selected, the user is prompted for an area name. When an existing area name is entered, that area's characteristics are displayed. Alternatively, the user can simply press the Return key to have the characteristics of all areas displayed. If the user enters a ? or a non-existent area name, a list of all area names is shown and the user is again prompted for an area name.
- b) If option I is selected, the user is prompted for a record type name. If the user simply presses the Return key, a list of all existing record types and all their respective data items is displayed. By responding with a ? or non-existent record type, the user is shown a list of record types only. In either case, the user is again prompted for a record type name. When this has been furnished, the user is prompted for the name of a data item in this record type.

When an existing item name is entered, the characteristics of that item are displayed. If instead the user simply presses the Return key, characteristics of all of the record type's items are displayed. If a ? or non-existent item is specified, a list of all of the record type's items is shown and the user is again prompted for an item name.

- c) If option R is selected, the user is prompted for a record type name. When an existing record type name is entered, that record type's characteristics are displayed. Alternatively, the user can simply press the Return key to have the characteristics of all record types displayed. If the user enters a ? or a non-existent record type name, a list of all record type names is shown and the user is again prompted for a record type name.
- d) If option S is selected, the user is prompted for a set name. When an existing set name is entered, that set's characteristics are displayed. Alternatively, the user can simply press the Return key to have the characteristics of all sets displayed. If the user enters a ? or a non-existent set name, a list of all set names is shown and the user is again prompted for a set name.
- e) If option W is selected, the user is prompted to indicate the name of a file on which a DDL specification for the current schema will be written. If the user presses the return key without specifying a file name, the DDL text is displayed on the console screen.
- f) If option E is selected, the Record Section menu reappears.

6. If option E is selected, then no further record type alterations can be made in the present DBRS Analyzer session. At this point, the Analyzer makes various checks to assure that the altered schema's area and record type specifications are consistent with each other. If they are inconsistent, diagnostic messages (see the last section of this Chapter) are displayed and the Record Section menu reappears. If no inconsistencies are detected, the user can begin making alterations to the schema's sets.

#### D. Set Alterations

The third kind of schema alterations that can be made concern the relationships among a schema's record types. The following menu appears:

Set Section:

- (A) Add a new set
- (C) Change existing set
- (D) Delete existing set
- (R) Rename set
- (S) Schema display
- (E) Exit

Selection?

When any option other than E is selected, the user is prompted to supply appropriate kinds of information. When processing is completed for that option, the Set Section menu reappears. When all desired alterations to sets have been made, the E option can be selected to exit from the DBRS Analyzer.

1. If option A is selected, a series of prompts appear. These ask the user to indicate the characteristics of the new set.

```
Set name?  
Set type? [ ]  
Fixed retention? [ ]  
Title?  
Synonym?  
Read access? [ ]  
Write access? [ ]
```

These prompts are asking for general information about the new set. Default responses shown in square brackets can be utilized by simply pressing the Return (Enter) key. Pressing the Return key instead of entering a set name returns the user to the Set Section menu. Pressing the Return key for the Title or Synonym prompt results in no title or synonym, respectively. If an invalid response to a prompt is given, then it is ignored and the prompt reappears.

Following the general information about the new set, the user is prompted for information about the set's owner(s).

```

Enter owner record types:
Owner record?
.
.
.
Owner record?
Auto owner insertion? [ ]

```

The user must supply the name of an existing record type in response to the first Owner record prompt. If a ? or a non-existent record type is specified, a list of existing record types is displayed and the prompt reappears. When a valid owner record has been identified, the user is prompted for further owner record types. When all desired owner record types for a set have been specified, the user presses the Return key in response to the next Owner record prompt.

Further owner information prompts appear if the new set has an N:M or N:1 type. These ask for information about the owner order.

```

Owner order (Fifo,Lifo,Next,Prior,Immat,Sorted)? [ ]

```

If a sorted order is requested there are further prompts concerning the owner sort key.

```

Enter owner sort key items:
Item?
Ascending? [ ]
.
.
.
Item ?
Duplicates (Not allowed,Fifo,Lifo,Immat)? [ ]
Index width? [ ]

```

The items that constitute an owner sort key are entered one per Item prompt. Each is followed by a prompt asking whether an ascending or descending sort is to be used for that item. When all items for the sort key have been specified in the desired sequence, the user simply presses the Return key in response to the Item prompt. The Duplicates and Index width prompts then appear.

After all owner information for the new set has been supplied, the user is prompted for information about the set's member(s).

```

Enter member record types:
Member record?
.
.
.
Member record?
Auto member insertion? [ ]

```

The user must supply the name of an existing record type in response to the first Member record prompt. If a ? or a non-existent record type is specified, a list of existing record types is displayed and the prompt reappears. When a valid member record has been identified, the user is prompted for further member record types. When all desired member record types for a set have been specified, the user presses the Return key in response to the next Member record prompt.

Further member information prompts appear if the new set has a 1:N or 1:1 type. These ask for information about the member order.

Member order (Fifo,Lifo,Next,Prior,Immat,Sorted)? [ ]

If a sorted order is requested there are further prompts concerning the member sort key.

Enter member sort key items:

Item?

Ascending? [ ]

.

.

.

Item ?

Duplicates (Not allowed,Fifo,Lifo,Immat)? [ ]

Index width? [ ]

The items that constitute a member sort key are entered one per Item prompt. Each is followed by a prompt asking whether an ascending or descending sort is to be used for that item. When all items for the sort key have been specified in the desired sequence, the user simply presses the Return key in response to the Item prompt. The Duplicates and Index width prompts then appear.

2. If option C is selected, the user is first asked to indicate which set needs to be changed.

Set name?

The user can type in the name of an existing set. Pressing the Return key without entering a name returns the user to the Set Section menu and no change is made. Alternatively, if a ? or non-existent set is specified, then a list of all existing sets is displayed and the user is again prompted to enter the name of the set whose characteristics are to be changed. When an existing set has been specified, a series of prompts appear. These give the user an opportunity to alter the set's existing characteristics.

Set type? [ ]  
Fixed retention? [ ]  
Title? [ ]  
Synonym? [ ]  
Read access? [ ]  
Write access? [ ]

The present nature (if any) of each characteristic is shown in square brackets. If no change is desired to a characteristic, the user merely presses the Return (Enter) key. An existing title or synonym can be eliminated by pressing the space bar and then the Return key. To make a change to a characteristic, the user explicitly types in the new value. The end of this response is signaled by pressing the Return key. If an invalid response is made to a prompt, the response is ignored and the prompt reappears.

Following the opportunity to change general information about a set, the user gets a chance to change the set's owner characteristics.

Edit set owner? [yes]  
Owner record? [ ]  
.  
.  
.  
Owner record?  
Auto owner insertion? [ ]

If no changes are desired to the owner information, the user responds to the first prompt with: n or no. This causes processing to proceed to member information. If the response to the first prompt is yes, then a prompt appears for each of the set's owning record types. The record type name of each is enclosed in square brackets. If the user presses the Return key, the record type remains as an owner of the set. If the user presses the space bar and then the Return key, then the record type is no longer an owner of the set. The record type can be replaced as an owner of the set by typing in the name of a different record type. After prompts have been issued for all existing owners, the Owner record prompt continues to appear until the user simply presses the Return key to indicate that no additional owners are to be added to the set.

Further owner information prompts appear if the set presently has an N:M or N:1 type. These give an opportunity to change the owner order.

Owner order (Fifo,Lifo,Next,Prior,Immat,Sorted)? [ ]

If the response indicates a sorted order, then there are further prompts allowing the sort key to be altered.

```

Edit owner sort key items? [yes]
Item? [ ]
Ascending? [ ]
.
.
.
Item?
Duplicates (Not allowed,Fifo,Lifo,Immat)? [ ]
Index width? [ ]
    
```

The sort key's existing items, if any, are shown (in square brackets) one at a time. Pressing the Return key, keeps that item in the sort key. Pressing the space bar and then the Return key removes the item from the sort key. The item can be replaced by entering a new item. In cases where the item is not removed, the user is given a chance to request ascending or descending sorting for it. When all items have been specified in the desired sequence, the user simply presses the Return key in response to the Item prompt. The Duplicates and Index width prompts then appear and can be changed if desired.

After changes (if any) have been made to the set's owner information, the user is given an opportunity to change the member information.

```

Edit set member? [yes]
Member record? [ ]
.
.
.
Member record?
Auto member insertion? [ ]

```

If no changes are desired to the member information, the user responds to the first prompt with: n or no. This causes processing to proceed to member information. If the response to the first prompt is yes, then a prompt appears for each of the set's owning record types. The record type name of each is enclosed in square brackets. If the user presses the Return key, the record type remains as a member of the set. If the user presses the space bar and then the Return key, then the record type is no longer a member of the set. The record type can be replaced as a member of the set by typing in the name of a different record type. After prompts have been issued for all existing members, the Member record prompt continues to appear until the user simply presses the Return key to indicate that no additional members are to be added to the set.

Further member information prompts appear if the set presently has a 1:N or 1:1 type. These give an opportunity to change the member order.

```

Member order (Fifo,Lifo,Next,Prior,Immat,Sorted)? [ ]

Edit member sort key items? [yes]
Item? [ ]
Ascending? [ ]
.
.
.
Item?
Duplicates (Not allowed,Fifo,Lifo,Immat)? [ ]
Index width? [ ]

```

The sort key's existing items, if any, are shown (in square brackets) one at a time. Pressing the Return key keeps that item in the sort key. Pressing the space bar and then the Return key removes the item from the sort key. The item can be replaced by entering a new item. In cases where the item is not removed, the user is given a chance to request ascending or descending sorting for it. When all items have been specified in the desired sequence, the user simply presses the Return key in response to the Item prompt. The Duplicates and Index width prompts then appear and can be changed if desired.

When the set has been changed, a summary of the set's new characteristics is shown and the Analyzer asks

OK? [yes]

A response of no gives the user a chance to change the set's characteristics. Alternatively, pressing the Return key causes the Set Section menu to reappear.

3. If option D is selected, the user is asked to indicate which set is to be deleted.

Set name?

The user can type in the name of an existing set. Pressing the Return key without specifying a name returns the user to the Set Section menu and no deletion occurs. If a ? or a non-existent set is specified, a list of all existing sets is displayed and the user is again prompted to enter the name of the set to be deleted. When an existing set name has been supplied, it is irrevocably deleted from the schema and the Set Section menu reappears.

4. If option R is selected, the user is asked to indicate which existing set is to be renamed.

Old name?

The user can type in the name of an existing set. Pressing the Return key without specifying a name returns the user to the Set Section menu and no renaming occurs. If a ? or non-existent set is specified, a list of all existing sets is displayed and the user is again prompted to enter a set name. When an existing set name has been specified, the user is asked to specify the new name for the set.

New name?

When a valid name that does not duplicate an existing set's name has been specified, the set has a new name and the Set Section menu reappears.

5. If option S is selected, the Schema Display menu appears and the user can select any of its options to see a desired aspect of the schema.

Schema Display:

- (A) Area display
- (I) Item display
- (R) Record display
- (S) Set display
- (W) Write DDL file
- (E) Exit to path specification

Selection?

The portion of the schema that is displayed is entirely up to date, reflecting all schema alterations that have occurred in the present DBRS Analyzer session. When processing for any selected option (other than E) is completed, the Schema Display menu automatically reappears.

- a) If option A is selected, the user is prompted for an area name. When an existing area name is entered, that area's characteristics are displayed. Alternatively, the user can simply press the Return key to have the characteristics of all areas displayed. If the user enters a ? or a non-existent area name, a list of all area names is shown and the user is again prompted for an area name.
- b) If option I is selected, the user is prompted for a record type name. If the user simply presses the Return key, a list of all existing record types and all their respective data items is displayed. By responding with a ? or non-existent record type, the user is shown a list of record types only. In either case, the user is again prompted for a record type name. When this has been furnished, the user is prompted for the name of a data item in this record type.

When an existing item name is entered, the characteristics of that item are displayed. If instead the user simply presses the Return key, characteristics of all of the record type's items are displayed. If a ? or non-existent item is specified, a list of all of the record type's items is shown and the user is again prompted for an item name.

- c) If option R is selected, the user is prompted for a record type name. When an existing record type name is entered, that record type's characteristics are displayed. Alternatively, the user can simply press the Return key to have the characteristics of all record types displayed. If the user enters a ? or a non-existent record type name, a list of all record type names is shown and the user is again prompted for a record type name.

- d) If option S is selected, the user is prompted for a set name. When an existing set name is entered, that set's characteristics are displayed. Alternatively, the user can simply press the Return key to have the characteristics of all sets displayed. If the user enters a ? or a non-existent set name, a list of all set names is shown and the user is again prompted for a set name.
  - e) If option W is selected, the user is prompted to indicate the name of a file on which a DDL specification for the current schema will be written. If the user presses the Return key without specifying a file name, the DDL text is displayed on the console screen.
  - f) If option E is selected, the Set Section menu reappears.
6. If option E is selected, no further set alterations can be made in the present DBRS Analyzer session. At this point the Analyzer makes various checks to assure that the altered schema's area, record type and set specifications are consistent with each other. If they are inconsistent, diagnostic messages (see the last section of this chapter) are displayed and the Set Section menu reappears. If no inconsistencies are detected, the user can proceed to specify physical structuring strategies to be followed during data base regeneration.

### E. Path Specifications

One or more paths can be specified for use in restructuring. A path does not in any way affect the schema. Instead, it is stated in terms of the schema (as altered with the Set Section menu) and is used to govern the physical restructuring of the data base. As in MDBS QRS, a path is simply a sequence of set names. A path begins with a SYSTEM-owned set. Each pair of sets in a path must be mediated by a different record type. In other words, a path must be completely connected and cannot contain loops. A set's name can be prefaced with the > symbol to indicate an upstream (i.e., member to owner) path direction for that set.

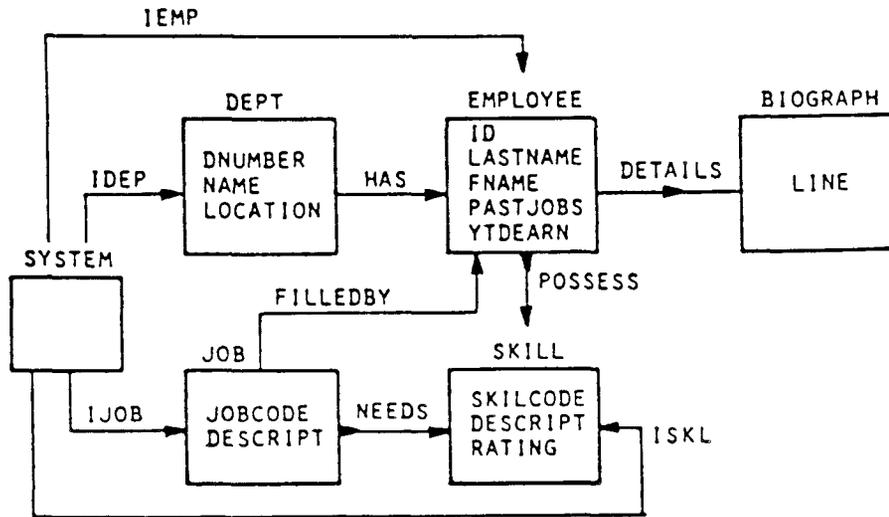


Figure II-1. Schema for the JOBS Data Base

For example,

- IDEP,HAS,POSSESS is a valid path
- IDEP,HAS,>FILLED BY,NEEDS is a valid path
- IDEP,POSSESS is invalid (it skips HAS)
- IJOB,>POSSESS,>HAS is invalid (it skips NEEDS)
- IJOB,NEEDS,>POSSESS,>FILLED BY is invalid (it loops into JOB)
- HAS,POSSESS is invalid (it does not begin with a SYSTEM-owned set)

MDBS QRS uses a path to determine the relationships that are to be used when data is retrieved. In contrast, an MDDBS DBRS path governs the subsequent physical clustering of all records existing along that specified path of relationships. During the data base restructuring, related records along a path are clustered in a depth-first fashion. For instance, specifying the path

IDEP.HAS,DETAILS

will cause each BIOGRAPH record to be clustered near its related EMPLOYEE record which is, in turn, clustered near its related DEPT record. Thus, for a given DEPT record, all of its related EMPLOYEE records are clustered about it and each of these EMPLOYEE records is accompanied by its related BIOGRAPH record.

In highly congested situations, a sub-clustering strategy is employed during the data base restructuring. For instance, if all of a department's employee records (and their related biographical records) cannot be arranged in a single cluster, they are grouped into distinct sub-clusters.

Multiple paths can be specified before leaving the DBRS Analyzer. The initial

Path?

prompt will reappear after each path has been specified. When the user has finished specifying all (if any) desired restructuring paths, the Return (Enter) key is pressed in response to the Path prompt.

The order in which multiple paths are specified is important. Path clustering will be performed for the first path, then a second, and so forth. Once a record has been clustered for a path, it is not shifted to different positions in subsequent path clusterings.

If the schema specifies that occurrences of a record type are to be CALCed, then the placement of those records' physical positions are determined by their CALC key values without regard to path clustering. For instance, if DEPT had a CALC location mode and the

IDEP,HAS,DETAILS

clustering path is requested, the DEPT records are first loaded based on their CALC key values. Later, related EMPLOYEE and BIOGRAPH records are clustered about the appropriate DEPT records.

Unlike the clustered record location mode that can be specified in a schema, path clustering controls record positioning only during the data base restructuring process. It does not govern physical record placements on an ongoing basis after the data base has been restructured. In the course of data base restructuring, all requested path clusterings take precedence over the clustered record location modes (if any) specified in the schema's record section.

#### F. Interactive DBRS Analyzer Diagnostics

These diagnostics can occur when alterations are being specified interactively to the Analyzer. Such an error must be corrected before proceeding to specify the next alteration.

\*\*\* a maximum of 255 repetitions are permitted

A repeating field cannot have more than 255 values per record occurrence.

\*\*\* a maximum of \_\_\_ areas may be defined

The indicated number of areas is the maximum that can be defined per data base in this environment.

\*\*\* area \_\_\_\_\_ already exists

An area having the indicated name already exists in this data base. Two areas cannot have the same name.

\*\*\* area \_\_\_\_\_ cannot contain record \_\_\_\_\_

The indicated record type's occurrences will be too long to allow them to fit in the specified area's pages. Record size should be diminished or area page size should be increased in order to allow the area to contain occurrences of the record type.

\*\*\* area \_\_\_\_\_ does not exist

A non-existent area cannot be altered or deleted. Check the spelling of the area name.

\*\*\* area \_\_\_\_\_ requests page size of \_\_\_\_ bytes;

\*\*\* the main data base area's page size limits this to \_\_\_\_ bytes

An excessive page size has been requested. In this environment, the main area's page size limits the size of pages in extra areas.

\*\*\* CALC key item \_\_\_\_\_ does not exist in record \_\_\_\_\_

The indicated item does not exist in the specified record type. Thus, it cannot participate in that record type's CALC key.

\*\*\* cannot display schema

Schema information will not be shown for a data base initialized for OEM usage if the DBRS Analyzer does not have the same serial number as the DDL Analyzer that performed the initialization.

\*\*\* cannot save \_\_\_\_\_ on SAI file

Due to operating system security, the schema alteration information cannot be saved on the indicated SAI file.

\*\*\* EOF encountered in SAI file: batch processing completed

There are no further alterations in the SAI file.

\*\*\* excessive date or time value

The largest time that can be specified for a time range is 255:55:55. The largest upper bound for a date range is 127 years beyond the lower bound.

**\*\*\* excessive input ignored**

A range value cannot be longer than the item's size.

**\*\*\* high date must not precede low date**

The upper bound of a date range cannot be less than the lower bound.

**\*\*\* I/O error encountered in SAI file: batch processing completed**

No further alteration information in the SAI file can be processed.

**\*\*\* identical range values**

The upper and lower range boundaries should be different.

**\*\*\* improper number of decimal digits**

The number of decimal digits specified for an idec item cannot exceed the total number of digits.

**\*\*\* integer cannot have fractional parts**

A non-integer has been entered where an integer is expected.

**\*\*\* invalid character in numeric range**

A non-numeric character cannot appear in a numeric value. A minus or decimal point cannot appear in an unsigned value. A decimal point cannot appear in an integer value.

**\*\*\* invalid date or time delimiter**

A slash (/) is a valid date delimiter. A colon is a valid time delimiter.

**\*\*\* invalid selection**

Only the listed options are eligible for selection.

**\*\*\* invalid set name: \_\_\_\_\_**

The indicated name cannot serve as a set name.

**\*\*\* item \_\_\_\_\_ already exists**

The indicated item already exists in the record type being altered. A record type cannot contain two items with the same name.

**\*\*\* item \_\_\_\_\_ does not exist**

A non-existent item cannot be changed or deleted.

**\*\*\* item size out of range**

The specified size is too large for the item's type (see the pertinent System Specific or DDL Manual).

**\*\*\* maximum index width is \_\_\_\_**

The indicated width is the largest permissible for this sort key.

**\*\*\* multiple of 256 bytes only**

In this environment, an area page size must be a multiple of 256 bytes.

**\*\*\* no member record types specified for set \_\_\_\_\_**

The indicated set must have a member record type.

**\*\*\* no owner record types specified for set \_\_\_\_\_**

The indicated set must have an owner record type.

**\*\*\* number of pages must be from 3 through \_\_\_\_**

In this environment, an area cannot have fewer than 3 pages, nor more than the indicated number of pages.

**\*\*\* numerical overflow**

The number is too large.

**\*\*\* numerical underflow**

The number is too small.

**\*\*\* page size must exceed 255**

In this environment, a page size cannot be less than 256.

**\*\*\* path not connected between sets \_\_\_\_\_ and \_\_\_\_\_**

A path must be a connected sequence of sets. The indicated sets do not have a record type in common.

**\*\*\* ranges out of order**

A range's lower bound must be less than its upper bound.

**\*\*\* record \_\_\_\_\_ already exists**

The indicated record type already exists in the schema. Two record types in the same schema cannot have the same name.

**\*\*\* record \_\_\_\_\_ does not exist**

A record type cannot be changed or deleted if it does not exist in the schema.

**\*\*\* record type \_\_\_\_\_ location requested in area of set \_\_\_\_\_:  
record type does not participate in set**

Occurrences of the indicated record type cannot be clustered for the specified set, because the record type is not an owner (member) of the set.

**\*\*\* record type \_\_\_\_\_ location requested in area of set \_\_\_\_\_:  
set has not been defined**

Occurrences of the indicated record type cannot be clustered for a non-existent set.

**\*\*\* set \_\_\_\_\_ already exists**

The indicated set already exists in the schema. A schema cannot contain two sets with the same name.

**\*\*\* set \_\_\_\_\_ does not exist**

A non-existent set cannot be changed or deleted.

**\*\*\* set \_\_\_\_\_ not SYSTEM owned**

The first set in a path must be system-owned.

**\*\*\* sort key item \_\_\_\_\_ has different replication counts**

In order to participate in a sort key, the indicated item must have the same number of replications in every member (owner) of a forked set.

**\*\*\* sort key item \_\_\_\_\_ has different sizes**

In order to participate in a sort key, the indicated item must have the same size in every member (owner) of a forked set.

**\*\*\* sort key item \_\_\_\_\_ not consistently encrypted**

In order to participate in a sort key, the indicated item cannot be encrypted for some members (owners) of a forked set and un-encrypted for others.

**\*\*\* sort key item \_\_\_\_\_ not found in all record types**

In order to participate in a sort key, the indicated item must exist in all of the forked set's member (owner) record types.

**\*\*\* sort key item \_\_\_\_\_ of inconsistent types**

In order to participate in a sort key, the indicated item must have the same type in every member (owner) of a forked set.

**\*\*\* SYSTEM may not be a member**

The SYSTEM record type cannot be a member of a set.

**\*\*\* SYSTEM may be the only owner**

The SYSTEM record type cannot be an owner in a multi-owner forked set.

**\*\*\* the SYSTEM record cannot be altered**

The SYSTEM record type cannot be changed or deleted.

**\*\*\* the main data base area cannot be deleted**

The main area of a data base cannot be deleted, because all data bases must have a main area.

**\*\*\* too many record types**

A schema cannot contain more than 255 record types.

\*\*\* value must be from 0 through 65535

A value from 0 through 65535 is expected.

\*\*\* YES or NO response expected

To make a YES response to an Analyzer question, enter Y or y or YES or yes or Yes. To make a NO response to an Analyzer question, enter N or n or NO or no or No.

### G. Fatal Errors

Fatal errors are possible only after all alterations have been specified. The resultant RSD file will be unusable for subsequent decomposition. The SAI file can be used when re-invoking the DBRS Analyzer (with the -I argument) to expedite the process of producing a valid RSD file. In cases where the fatal error resulted from an internal error in data base processing, the corresponding DMS command status error is also displayed. If it resulted from an operating system error, that error is also displayed.

dbrs: write error on aux lit table

An attempt was made to add information to the auxiliary literal table, but was not successful (e.g., there is no more virtual memory remaining for the literal table).

dbrs: cannot allocate memory

There is insufficient actual central memory to continue processing. Increase the amount of central memory beyond 128K. Make the schema alterations in smaller, separate sessions. Minimize the re-alterations within a session.

dbrs: cannot open \_\_\_\_\_

The indicated file could not be opened for processing. Check to be sure that it is on-line and that the operating system has granted write access to it.

dbrs: cannot open data base

The data base could not be opened, due to an invalid data base name, user name or password. The data base's main area must be on-line.

dbrs: cannot reallocate memory

An attempt to reallocate central memory was unsuccessful. Increase the amount of central memory beyond 128K. Make the schema alterations in smaller "chunks." Minimize the re-alterations within a session.

dbrs: filename error: \_\_\_\_\_

The indicated name is not a valid file name.

dbrs: input error on selective automatic input file

The SAI file has been corrupted or the file indicated with a -I argument is not an SAI file.

dbrs: input/output error on restructured schema definition file

The RSD file is invalid (e.g., due to bad media or insufficient storage capacity).

dbrs: input/output error on temporary file

A temporary file used by the Analyzer is invalid (e.g., due to bad media or insufficient storage capacity).

dbrs: invalid title or synonym

The original data dictionary is corrupted in such a way that titles or synonyms are unrecognizable.

dbrs: read error on aux lit table

An attempt was made to read information in the auxiliary literal table, but was not successful.

This page intentionally left blank.

## III. DECOMPOSITION

The interactive analysis stage of restructuring does not in any way affect the data base or its data dictionary. Upon successful completion of this stage, the user can at some later time proceed to the decomposition stage. The DBRS Analyzer writes the revised schema description on an RSD file. This file, together with the entire data base, must be on-line during the decomposition stage. The result of this stage is a decomposition file named DBRS.DCM. Because this is a sequential access file, it can be stored on either tape or disk. The decomposition file can be stored across multiple volumes (e.g., multiple floppy disks).

## A. Data Value Transformations

In addition to causing changes in the data dictionary, certain kinds of schema alterations can also cause transformations in the data base's data values. These transformations begin during the decomposition stage and are consummated in the course of the subsequent regeneration stage.

## 1. Adding new data items

When a new data item is added to an existing record type, each occurrence of that record type is adjusted to accommodate a value for the new data item. If the new item has a numeric type, then its value in each record is initialized to 0. New string, character and binary item's have blank initial values. The initial values for a new time data item are all 000:00:00 and a new date item has the base date (01/01/1900) for its initial values. When the number of replications of an existing data item is increased, the same initialization rules are used. When an entire new record type is being added, no occurrences are created for it by the restructuring system.

## 2. Deleting existing data items

When a data item is deleted from an existing record type, each occurrence of that record type is physically compressed to eliminate the deleted item's value. When the number of replications of an existing data item is decreased, similar compression occurs to eliminate the appropriate number of trailing replications from each record. Deletion of an entire record type causes all of its record occurrences to be deleted during the restructuring process.

3. Changing data item size

If the size of a non-numeric data item is increased, its values are appropriately padded with blank fill on the right. If the size of a data item is decreased, some of its data values may be truncated to fit into the new smaller size. A non-numeric item's values are truncated on the right, if needed. Left-truncation occurs for integer and unsigned items. Left truncation can also occur for an idec data item's values unless the number of decimal digits is decreased by at least as much as the item's total size. For real data items, right truncation occurs up to the decimal point and then left truncation. Of course, the sizes of date and time items are always fixed.

4. Changing data item type

The DBRS Analyzer provides a wide degree of latitude in changing an existing data item's type. Type changes can result in various kinds of data value transformations. these are described in Table III-1. Type changes that are not allowed by the Analyzer are indicated by NA. Type changes that do not affect data values are indicated by SAME. In cases where the new type has a smaller size than the old type's size, data value truncation (as described previously) can occur.

Table III-1. Effects of Type Changes on Data Item Values

From \ To	Character	String	Binary	Integer	Unsigned	Real	Idec	Date	Time
Character	SAME	SAME, except Control characters become blanks	SAME	SAME up to first non-integer symbol	absolute value of SAME up to first non-unsigned symbol	SAME up to first non-real symbol	SAME up to first non-idec symbol	SAME, except null if invalid date	SAME, except null if invalid time
String	SAME	SAME	SAME	SAME up to first non-integer symbol	absolute value of SAME up to first non-unsigned symbol	SAME up to first non-real symbol	SAME up to first non-idec symbol	SAME, except null if invalid date	SAME, except null if invalid time
Binary	SAME	SAME, except Control characters become blanks and high order bits is turned off	SAME	NA	NA	NA	NA	NA	NA
Integer	left justified with + symbol suppression	left justified with + symbol suppression	NA	SAME	absolute value of SAME	SAME	SAME	NA	NA
Unsigned	left justified	left justified	NA	SAME	SAME	SAME	SAME	NA	NA
Real	left justified with + symbol suppression	left justified with + symbol suppression	NA	rounded	rounded and absolute value	SAME	SAME with decimal alignment	NA	NA
Idec	left justified with + symbol suppression	left justified with + symbol suppression	NA	rounded	rounded and absolute value	SAME with decimal alignment	SAME	NA	NA
Date	left justified	left justified	NA	NA	NA	NA	NA	SAME	NA
Time	left justified	left justified	NA	NA	NA	NA	NA	NA	SAME

## 5. Changing feasibility ranges

If the upper range boundary of an item is lowered, any item value that exceeds the new upper bound is changed to be equal to the upper bound. If the lower range boundary is increased, any value that is less than the new lower bound is changed to be equal to the lower bound.

## B. Record Relationship Transformations

Certain schema alterations to existing sets can cause transformations in existing record connections, as well as changes in the data dictionary. These transformations commence during the decomposition stage and are consummated in the regeneration stage.

### 1. Changing set types

When an N:M set has been changed to 1:N or 1:1, only the first owner connection for each of the set's member occurrences is preserved during the data base restructuring. The same is true, when an N:1 set has been changed to 1:1.

When an N:M set has been changed to N:1 or 1:1, only the first member connection for each of the set's owner occurrences is preserved during the data base restructuring. The same is true, when a 1:N set has been changed to 1:1.

### 2. Removing record types from a set

When the schema has been altered in such a way that a record type that had been a set's owner is no longer the owner of that set, then all occurrences of that record type are disconnected from the set during restructuring. Similarly, if a record type is no longer a member of a set, then all of its occurrences are disconnected from the set's owner occurrences during the restructuring. If an entire set has been deleted from the schema, then all connections among its owner and member occurrences are removed.

### 3. Changing set order

When the order of an existing set has been altered in the schema, the records will be reconnected based on the new order. If the new order is sorted (or if it is sorted based on a new sort key), the records will be reconnected based on the sort key. If the new order is lifo, the existing connections are not changed unless the former order was fifo. In that case, they are reconnected in reverse fashion. Similarly, if the new order is fifo, then existing connections are preserved unless the former order was lifo. In that case, they are reconnected in reverse fashion. Reversal also occurs when prior order is changed to next order or vice versa. In all other cases, existing connections are preserved.

### C. Effect on Screen Dictionary

If the data base holds a Screen Master screen dictionary, a copy of that dictionary's SDL specification should be saved before the data base is decomposed. This is easily accomplished with ISDU. Decomposition does not preserve a screen dictionary. After restructuring is completed, the saved SDL specification can be used by the SDL Analyzer to automatically build the screen dictionary. ISDU can be used to modify the screen dictionary (as needed) to conform to the new schema.

### D. The Decomposition Process

The decomposition process is carried out by the program provided on the executable file named DCOMP. This is executed by entering

DECOMP

on the operating system command line. A copy of all data base areas must be on-line. The file produced by the DBRS Analyzer must also be on-line if schema alterations are desired. Be sure to back up the data base prior to executing DECOMP.

Any combination of the following arguments can appear (in any order) after DECOMP on the command line.

- Bnnnn            where nnnn is the number of bytes to be allocated for the buffer used by DECOMP for data base interaction. If this argument is not used, approximately half of free memory is allocated for this buffer.
- Dfilename        where filename is the name of the file containing the main area of the data base whose schema is being altered. If this argument appears, DECOMP will not prompt for data base identification.
- Fnnnn            where nnnn is an integer specifying the blocking factor to be used in building the decomposition file. The default for this argument is operating system dependent, but is typically set to yield block sizes of 512 bytes. When writing to a tape device under UNIX, it is customary to designate a blocking factor of about 20.
- Ifilename        where filename is the name of a file containing the revised schema description (if any) that is to govern the data base decomposition.

- Nnnnn where nnnn is an integer specifying the maximum number of decomposition file blocks that are to be held on a single volume. If this argument is omitted, the entire decomposition file will be stored on a single volume. The decomposition file is truncated, if that volume has insufficient capacity.
- Ofilename where filename is the name of the sequential access file that will store the decomposition information. If this argument is omitted, DBRS.DCM is assumed.
- Ppass where pass is the user password that is to be used by DECOMP in attempting to open the data base. If this argument appears, the Decomposer will not prompt for a password.
- Q to cause a "quick" decomposition in which efforts at optimizing the data base's physical structure are abandoned.
- Username where username identifies a valid data base user. If this argument appears, DECOMP will not prompt for user identification.

During the decomposition stage, information is sequentially written to the decomposition file a block at a time. A block consists of one or more operating system records (not to be confused with data base records). The number of operating system records in a block is determined by the blocking factor. The size of each operating system record depends on the operating system (e.g., 128 bytes under PC DOS and MSDOS; 512 bytes under UNIX). When the -N argument is used to place a limit on the number of blocks per volume, the user will see the following prompt as soon as a volume is filled:

Mount new decomposition file volume...Press Return when ready

The decomposition file continues to be written onto the new volume. The user should keep track of the sequence in which these volumes are filled. They will need to be used in the same sequence, during the subsequent regeneration stage.

There are five phases in the decomposition process. As each phase is completed, a message appears indicating that the phase is finished and stating how many data base records were processed in that phase. The phases are:

Phase 1: Process data dictionary (data dictionary for the new schema is saved on the decomposition file).

Phase 2: Mark records/Process CALC records (marks all records in the data base and CALCed records are located and saved on the decomposition file).

Phase 3: Process clusters (path clusters are identified and saved on the decomposition file; remaining record clusters as declared in the schema are then identified and saved).

Phase 4: Process remaining records (save remaining records on the decomposition file).

As data base records are saved in the decomposition file, they are automatically adjusted to be consistent with the new schema. Successful decomposition ends with the message:

Decomposition Complete

#### **E. Decomposition Diagnostics**

If DECOMP detects that the RSD file was generated from a different data dictionary than that of the data base being decomposed the message

**\*\*\*WARNING: data dictionaries differ\*\*\***

appears. When decomposing a data base that was used with Screen Master, this is to be expected. In other situations, it is normally undesirable.

Decomposition diagnostics that begin with "decomp:" generally indicate fatal errors. Processing halts immediately. For other diagnostics, DECOMP will continue processing. If the original data base is internally inconsistent or corrupt, DECOMP is likely to be incapable of decomposing it in a meaningful way.

**dbk[\_\_\_,\_\_\_]: invalid record list -some record occurrences may be lost**

The record having the indicated data base key is the last one on the page that is accessible. Remaining records (if any) on the page are not decomposed.

**dbk[\_\_\_\_,\_\_\_\_]: invalid record occurrence encountered**

The record having the indicated data base key does not match the internal identifier of any record type existing in the data dictionary.

**dbk[\_\_\_\_,\_\_\_\_]: owner/member removed for set \_\_\_\_\_**

The record having the indicated data base key has a reference to a non-existent owner or member. That reference is nulled as the record is decomposed.

**dbk[\_\_\_\_,\_\_\_\_]: record variable part truncated for record type \_\_\_\_\_**

The variable length portion of the record having the indicated data base key is inaccessible and is not decomposed.

**decomp: \*\*\* interrupted \*\*\***

The DECOMP execution was interrupted by the user.

**decomp: cannot find SYSTEM record**

DECOMP cannot find the data base's SYSTEM record. The original data base is corrupt.

**decomp: cannot open data base**

DECOMP is unable to open the data base. A valid data base name, user name and password must be provided.

**decomp: fatal loss of data dictionary**

DECOMP is unable to find the data base's data dictionary. The original data base is corrupt.

**decomp: incompatible restructured system definition**

The DECOMP version is not compatible with the version of the DBRS Analyzer that produced the RSD file.

**decomp: incorrect generation data base**

The DBRS Analyzer was run against a different data base than the one presently being used with DECOMP. The RSD was not generated from the same data base that is now being decomposed.

**decomp: invalid blocking factor**

The blocking factor specified with the -F argument is not workable. The blocking factor size is interdependent with the volume block count.

**decomp: invalid volume block count**

The volume block count specified with the -N argument is not workable. It is interdependent with the blocking factor.

**decomp: item conversion error**

A corrupt data item value was encountered.

**decomp: system table access error**

The data dictionary is corrupt.

**decomp: title/synonym read error**

A title or synonym is corrupt.

**decomp: value label - item name not found \_\_\_\_\_**

The item or value label information is corrupt.

**decomp: value label - record name not found \_\_\_\_\_**

The record type or value label information is corrupt.

**System table storage errors - some \_\_\_\_\_ lost**

Errors were detected in the data dictionary's system storage tables. Information of the indicated type was therefore lost during decomposition. Loss of value label or macro information is not nearly as severe as loss of schematic information.

## IV. REGENERATION

At any point after decomposition, the resultant decomposition file can be used to regenerate the data base according to the new schema and with a more optimal physical structure.

## A. The Regeneration Process

The regeneration process is carried out by the program provided on the executable file named REGEN. This is executed by entering

REGEN

on the operating system command line. The decomposition file must be on-line. If this file is on multiple volumes, the first volume must be mounted first. As each subsequent volume is needed, REGEN will issue the message:

Mount next decomposition file volume...Press RETURN when ready

These volumes must be mounted in the same order in which they were generated.

The following optional argument can appear after REGEN on the command line:

- Fnnnn            where nnnn is an integer specifying the blocking factor that was used in building the decomposition file.
- Ifilename        where filename is the name of the decomposition file. This can be a fully qualified file name. If a drive is specified, each of the file's volumes must in turn be mounted on that drive. If this argument is omitted, the file name is assumed to be DBRS.DCM and the default drive is used.

There are five phases in the regeneration process. As each phase is completed, a message appears indicating that the phase is finished. The phases are:

- Phase 1: Initialize areas
- Phase 2: Load data dictionary
- Phase 3: Load records and allocate index space
- Phase 4: Establish record map
- Phase 5: Build indexes and pointer arrays
- Phase 6: Establish set ordering map
- Phase 7: Reorder sorted sets

At the end of the seven regeneration phases, the message

Regeneration complete

appears and the modified data base is ready for use. Phase 3 is normally the most time-consuming phase. If the schema revision did not involve the introduction of a new sorted ordering for any set, then the last two phases are skipped. These two phases tend to be disk intensive and lengthy.

## B. Regeneration Diagnostics

Regeneration diagnostics that begin with "regen:" are fatal, which means that processing halts and the data base is not successfully regenerated. Other diagnostics do not halt the processing, but the resultant regenerated data base must be decomposed and regenerated again in order to be usable.

**dbk[\_\_\_,\_\_\_]: CORRUPT reference to missing record**

The record whose data base key is indicated has a reference to a non-existent record.

**dbk[\_\_\_,\_\_\_]: CORRUPT reference to missing set structure**

The record whose data base key is indicated has a reference to a non-existent index or pointer array.

**regen: cannot create record**

A decomposed record cannot be regenerated because it has corrupt data, the target area is full, or the decomposition was unsuccessful.

**regen: decomposition file format error**

The file specified as input to the regeneration is not a decomposition file.

**regen: exec failed**

In UNIX environments, an exec of a REGEN module failed.

**regen: incompatible decomposition file**

The decomposition file was generated by a version of DECOMP that is incompatible with the version of REGEN.

**regen: internal close failed**

REGEN's own internal attempt to execute DBCLS was unsuccessful.

**regen: internal open failed**

REGEN's own internal attempt to execute DBOPN was unsuccessful.

**regen: invalid blocking factor \_\_\_\_**

The blocking factor specified with the -F argument is not the same as the blocking factor that was specified for decomposition.

This page intentionally left blank.

DOCUMENTATION COMMENT FORM

MDBS Document Title: \_\_\_\_\_

We welcome and appreciate all comments and suggestions that can help us to improve our manuals and products. Use this form to express your views concerning this manual.

Please do not use this form to report system problems or to request materials, etc. System problems should be reported to MDBS by phone or telex, or in a separate letter addressed to the attention of the technical support division. Requests for published materials should be addressed to the attention of the marketing division.

Sender:

_____	_____
(name)	(position)
_____	_____
(company)	(telephone)
_____	
(address)	
_____	
(city, state, zip)	

COMMENTS:

Areas of comment are general presentation, format, organization, completeness, clarity, accuracy, etc. If a comment applies to a specific page or pages, please cite the page number(s).

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

Continue on additional pages, as needed. Thank you for your response.