KE Texpress

Maintenance Guide

KE Software Pty Ltd

Contents

Chapter 1

Introduction

Overview

This manual describes the KE Texpress database maintenance utilities which are available to the database. They include backup and restore, Look-up table editing and machine crash recovery. Several of these maintenance procedures can be performed by appropriately privileged users, who must be a member of the Unix group, KE Texpress, and possess a database user account with Administrator menu privilege.

Using this Guide

The maintenance operations provided by KE Texpress enable an administrator to provide a reliable and efficient environment for the capture and retrieval of data.

Many maintenance facilities require exclusive access to a database. Thus it is sometimes necessary temporarily to prevent database access to all users, so that these critical operations can be performed. Chapter 2 describes a tool for *closing* a database, which can terminate users' sessions and prevent further sessions commencing. Also described are tools for *re-opening* the database and *analyzing* the use of the database by the user population.

Computers and, indeed, programs are not infallible and without adequate caution and preventative action, data will be lost. The most useful and, hopefully, most used maintenance utility is the database *backup* program. This enables the entire database or selected critical parts, to be copied to a removable, archival medium. This utility is described in Chapter 3.

A less often used utility which is of equal importance is the database *restore* program. This is described in Chapter 4. Following a catastrophe or human error of some kind, all or part of a database backup can be restored on to the computer, thus preventing major loss of data. The usefulness of this facility is totally dependent on the regularity with which backups are performed.

Chapter 5 describes a utility for editing and pre-loading *Look-up* tables. It also describes commands to make Look-up table access more efficient and to control updates to Look-up tables.

Chapter 6 describes a facility for *recovering from computer failure*. Also described is a general maintenance utility which must be executed periodically to improve the efficiency of disc usage of the database.

Chapter 7 describes facilities which can dramatically improve the performance of interactive insertions and edits, while Chapter 8 contains descriptions of (several rarely) used database repair utilities.

Maintenance Pull-Down Menu

Administrator menu privilege enables a user to invoke the Administrator menu (or Admin mode). by running the command.

    texpressadm   *dbname*

where *dbname* is the database name.

Most of the maintenance utilities can be invoked directly from the Maintenance menu, which forms one of the pull down menus of Admin. The exact appearance of this menu depends on the user's privileges. An example of the Maintenance menu appears in the following figure:

```
[Admin]  Function  Maintenance  Audit  Batch
                  <Close database...              >
                  Open database...
                  User report...          Ctrl+W

                  Database usage...
                  Backup database...      Ctrl+B
                  Restore database...     Ctrl+L
                  Rebuild/load/progress...  Ctrl+A
                  Edit Look-up tables...

                  Database recovery...    Ctrl+V
                  Database maintenance...  Ctrl+M

                  Edit start message...
```
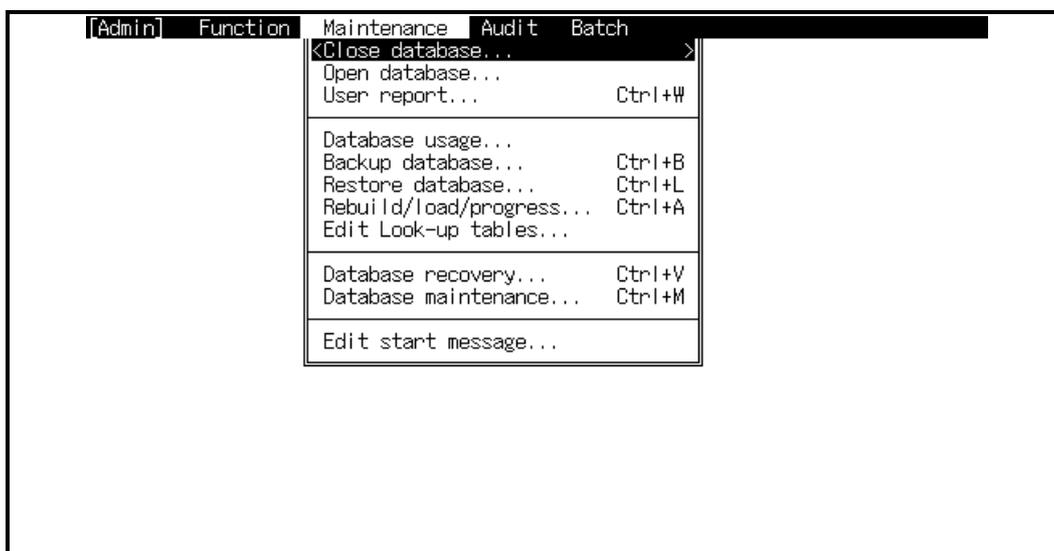
Maintenance menu

This manual assumes a basic knowledge of KE Texpress, including familiarity with the help facility, the command structure, screen layout and the database structure (refer to either the KE Texpress User Guide or the KE Texpress Design Guide).

Maintenance Warning Messages

As several of the maintenance utilities should be executed regularly, KE Texpress automatically determines when they should be performed, and issues warnings to the administrator whenever Admin is invoked. These warnings cover the following four areas:

Database backup

If the database has not been backed up for more than seven days and it has been modified in that time, a warning displays:

```
A database backup should be performed
```

This message appears only for users with **Backup database** privilege.

Batch insertion

To provide fast interactive insertions and edits, KE Texpress provides a facility which transparently delays a significant proportion of the insertion until a later time. If the database contains more than 500 such partially inserted records, warning displays to the DBA:

```
A batch insertion should be performed
```

Database maintenance

If the database maintenance utility has not been performed for more than 14 days and the database has been modified in that time, a warning displays:

```
Database maintenance should be performed
```

This message appears for all users who are privileged to invoke Admin.

Audit file flush

KE Texpress provides a facility to maintain an audit trail of operations performed on a database (refer to the KE Texpress Utilities Guide). If enabled, it maintains an audit file which contains information about each database operation performed. Should this audit file become larger than 50% of the size of the data file (the file containing all of the records stored in the database) and the audit file size is greater than 50 kilobytes, a warning displays:

```
The audit file should be flushed
```

This indicates that the size of the audit file should be reduced by removing the oldest information. This message appears only for users with **Audit control** privilege.

Data file compaction

If many insertions and edits are being performed, it is possible for the data file to be quite fragmented. Also, the free list file which contains offsets of free storage space in the data file may become quite large. This will result in a slight gradual deterioration of the time taken to perform an insert or edit. If this condition is detected, a warning message displays:

```
Data file compaction should be performed
```

This message appears only for users with **Backup database** privilege. Data file compaction is achieved by performing a database backup and compacted restore, followed by an index rebuild (refer to the following chapters and to the KE Texpress Design Guide for more details).

The parameters listed above (such as the number of days between backups, etc.) can be changed by the DBA. The warnings can be disabled completely if required (refer to the KE Texpress Interface Guide).

Command Notation

Commands run within KE Texpress can be performed from pull-down menus, or via keyboard accelerators (characters). Often, a command can be performed in several ways. <u>For uniformity and ease of use, only the pull-down menu option is described in the guide</u>.

The *Perform Query* command can be run by selecting the *Query* option on the *Function* pull-down menu in *Query* mode. The guide would describe it as:

```
Perform Query ([Query] Function ⇒ Query)
```

where the description of the command is followed by the pull-down menu option in brackets. The mode is contained in square brackets, e.g. [**Query**] and the pull-down menu name **Function** is followed by ⇒ and the command **Query** exactly as listed on the pull-down menu.

Alternatively, the Perform Query command can be performed from the keyboard, by holding down the **Control** key and typing **Y**. In the guide, the keystrokes are described as **Ctrl+Y**.

Many of the common commands may be invoked directly by a particular function key (provided the terminal supports function keys). Throughout KE Texpress (and the screen images displayed in this document), function keys are represented by the letter **F** followed by the function key number. Thus the command generated by pressing function key number one is represented as **F1**. The notation, F1 , is used in this document.

Other special symbols used in this document are:

ESC	The **escape** key.
↵	The **return** or **enter** key.
DEL	The **delete** or **interrupt** key (used to interrupt an operation).
Space	The **space bar**.
Backspace	The **backspace** key.
Tab	The **tab** key.
←	The **left arrow** key.
→	The **right arrow** key.
↑	The **up arrow** key.
↓	The **down arrow** key.

Chapter 2

Close, Open and Activity

Overview

This chapter describes the utilities used to close and re-open a database and to determine who is currently using the database. Only users with **Open/Close database** privilege can perform these operations. The chapter also describes the statistics on database usage which are automatically maintained by KE Texpress. All users with **Administrator menu** privilege can view these statistics.

Close Database
([Admin] Maintenance ⇒ Close database)

Many of the maintenance utilities require exclusive access to the database. If no users are currently accessing the database, the maintenance utilities silently close the database for the duration of the operation. However, they cannot be invoked while users are active. Hence, it is often necessary explicitly to close the database before invoking the utility.

If the database is already closed when this option is invoked, a message displays:

```
Database is closed
```

Otherwise, the user is prompted for a time interval:

```
Enter minutes till closing:
```

A time period, in minutes, may then be entered. By default a time period of five minutes is selected. Entering zero (0) minutes indicates that the database is to be closed immediately. Entering an empty value by pressing the **ESC** key aborts the command.

After the time period is established, the user is prompted:

```
Wait until database is closed? [yes] [no]
```

This gives the user the option of monitoring the database close operation or performing the operation in background. Choosing yes causes the messages to display:

```
Commencing database close program . . .
```
and
```
Closing database . . .
```

The close program notifies all current users of the impending database closure. Notification occurs at decreasing intervals until the database is closed. As the close time approaches, no further users will be permitted to enter the database.

Just before closure, those users who are still accessing the database (having ignored all previous warnings) will have their sessions forcibly terminated.

A message will display when the database is closed:

```
Database is now closed
```

If the database close operation is performed in background, the user closing it is notified when the database is closed. This allows a database close operation to be initiated several hours before the intended closure time without monopolizing the terminal.

If the Maintenance menu close option is selected and it is detected that a close operation is already running, a message display:

```
Database close pending in n minutes. Abort close?
[yes] [no]
```

where *n* is the time period before closure. In this situation there is the option of aborting the original close operation (and selecting a new time period if desired).

Only users with **Open/Close database** privilege can perform this operation.

Open Database
([Admin] Maintenance ⇒ Open database)

This option re-opens the database, allowing user access. The user is prompted:

```
Database is now open. Notify users? [yes] [no]
```

It is possible to send notification to all people who are currently logged on and are registered users of the database. Choosing yes displays a list of all notified users.

If the database is already open when this operation is selected, a message is displayed stating that fact and users may still be notified if desired.

If the database open option is selected when a close operation is already running, a message displays, and seeks confirmation:

```
Database close pending in n minutes. Abort close?
[yes] [no]
```

where *n* is the time period before closure. In this situation the user has the option of aborting the close operation (hence leaving the database open).

This operation can be performed only by users with **Open/Close database** privilege.

Database User Activity Report
([Admin] Maintenance ⇒ User report)

This option is used to determine what activity is currently occuring on the database. It can be directly invoked by typing **Ctrl**+**w**. If the database is closed a message displays:

```
Database is closed
```

is displayed. Otherwise a message displays:

```
Commencing database who program . . .
```

A list of all current database users is displayed as shown in the following figure:
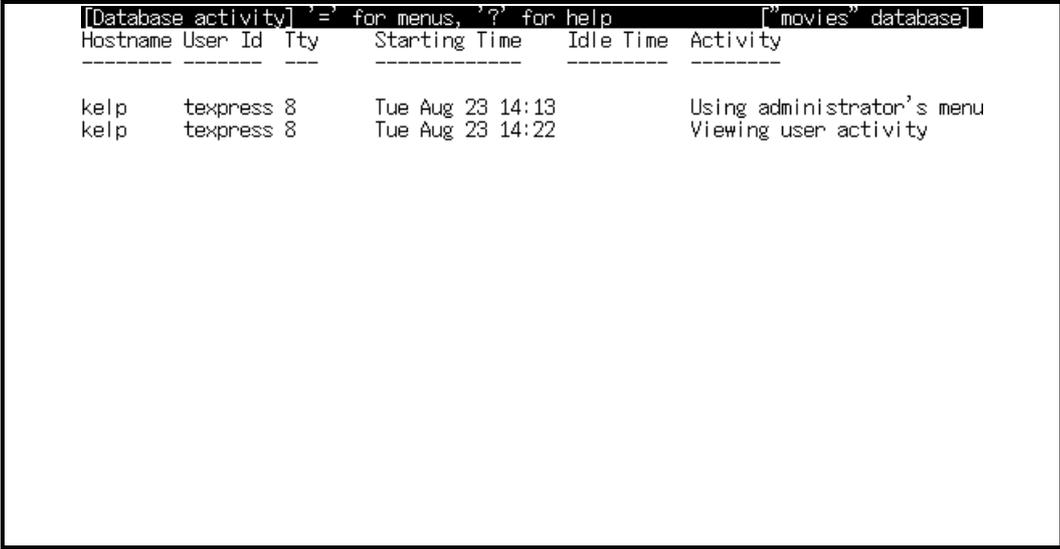
```
[Database activity] '=' for menus, '?' for help              ["movies" database]
Hostname User Id  Tty      Starting Time     Idle Time  Activity
-------- ------- ---      -------------     ---------  --------

kelp     texpress 8        Tue Aug 23 14:13             Using administrator's menu
kelp     texpress 8        Tue Aug 23 14:22             Viewing user activity
```

Database activity report

All users with **Administrator menu** privilege can perform this command.

Database Usage Statistics
([Admin] Maintenance ⇒ Database usage)

A summary of statistics indicating database usage can be displayed using this option. There is an equivalent option in the User database. The exact format of the statistics is dependent on the database usage to date.

```
[Viewing statistics] '=' for menus, '?' for help          ["movies" database]

            "movies" Database Usage

     Insertions            :       26
     Deletions             :        0
     Edits                 :        0
     Total number of records :     26
     Database utilization  :     13.0%

     Indexed terms per record :   62.0
     Segments per query    :        1.9
     Index false matches   :        0
     Range false matches   :        0
     Percentage correct    :     100.0%

     Queries               :       23
     Total matches         :       29

            Query distribution

   No.   Item              Count    %

      1.  "Title:"            2    8%
```

Database Usage Statistics

The report lists the total number of insertions, queries (via texforms), deletions and edits, the total number of records in the database, a figure representing the percentage of the database's capacity which has been used so far and, for the DBA only, the average number of indexed terms per record. Note that queries performed via KE Texql are not included in the total number of queries in this report.

The report also lists the total number of matches found for all queries, the number of false matches found (index and range) and a figure representing the number of correct matches as a percentage of the total number of matches. The percentage figure relates only to the index false matches. For the DBA, the average number of segment matches per query is also listed. Segment matches are described in more detail in Chapter 8 of the KE Texpress Design Guide.

Finally, if there has been at least one query that retrieved a record or records, then a brief break-down of the format of the queries is displayed. This shows the prompts of (up to) the sixteen most popular items used in queries with a count of the number of times each has been used and the percentage of queries in which it has been used. For an item to appear in this list, it must have been used at least once in a query. For ranged items, if an upper and a lower bound are specified in the same query, then this counts as two uses of the item.

All users with **Administrator menu** privilege can perform this operation.

Start Message
([Admin] Maintenance $\Rightarrow$ Edit start message)

A file called motd can be created in the database directory to contain a database start message. This information is displayed whenever the User or Admin datadase is called up.

The start message is edited using any Unix editor. The default editor is typically **vi**. To change the editor, use the Unix shell EDITOR environment variable.

Chapter 3

Database Backups

Overview

Database backups are crucial, to ensure the reliable, continuous use of a database. Computers and programs are not infallible and catastrophes can occur. Human error can also cause loss of data. Without adequate caution and preventative action, data will be lost.

To minimize the risk of lost data, regular, documented backups should be performed. KE Texpress provides a very simple, menu-driven backup utility which enables the entire database, or selected critical parts of it, to be copied to a removable, archival medium. Several backups of the database should exist simultaneously, with new backups replacing the oldest existing backups. Periodically, a backup should be retained as a permanent copy. All backup media should be clearly labelled and dated, and at least one recent backup should be stored off site.

KE Texpress database backups (and restores) are completely portable across machines with different architectures. A database can be backed up on one machine and restored on any other machine running KE Texpress, with complete database functionality being guaranteed.

The KE Texpress backup procedure is as follows:

(1) The particular Unix backup device to be used for the backup is selected.

(2) The appropriate backup type is chosen. (Individual files may be chosen, if desired)

(3) The operation is performed and monitored. The backup volumes are changed as necessary.

The backup utility is selected from the Maintenance pull down menu of Admin or directly invoked by typing **Ctrl B**. Only users with **Backup database** privilege can perform this operation. The backup type selection procedure commences.

Device Selection

Backups are written using a designated Unix device which is referred to as the backup device. Typically this is a magnetic cartridge or tape unit, which is the standard means of backup in most computer environments. However, it can be one of a variety of other devices and files.

The first time that a KE Texpress backup is performed on a database , no device is selected and the user must select one before the backup can be performed. KE Texpress automatically displays the Backup Device Selection box from which a selection can be made. Alternatively, if a backup has already been performed, then the previously selected backup device remains selected. A different device can be selected by choosing the *Select device* command ([-D) from the backup type selection menu backup devices being . An example of the backup device selection box is given in the following figure. A selection can be made using the Selection Box * commands described in

```
┌─────────────────────────────────────────────────────────────┐
│ [Backup] '=' for menus, '?' for help              ["movies" database]│
│                 Backup report for "movies" database            │
│                    ═══ Device Selection ═══                    │
│ Device na│─────── Device List ──────────────────────│ock size │
│ Test Disk│<Test Disk Backup                        >│  10240  │
│          │ Test Disk Backup                         │         │
│          │ Cartridge (150Mb, bs 10240)              │         │
│          │ Cartridge (150Mb, bs 20480)              │         │
│ Type     │                                          │         │
│          │                                          │         │
│ Data file│                                          │         │
│ Index fil│                                          │         │
│ Look-up t│                                          │         │
│          │                                          │         │
│          │ █                                        │         │
└─────────────────────────────────────────────────────────────┘
```
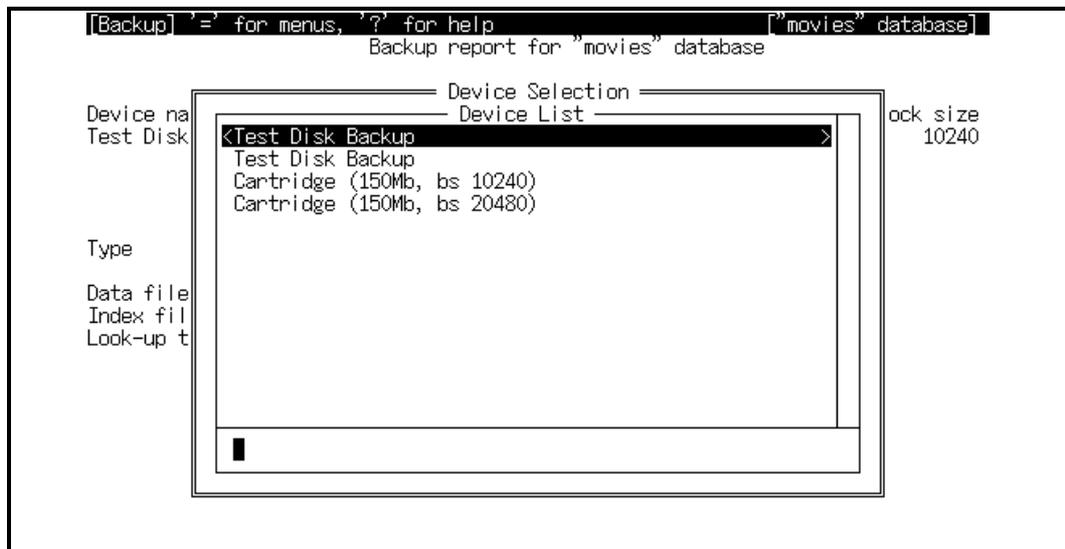
Backup Device Selection

Device selection is required only the first time a backup is performed for the database, or whenever the backup device is to be changed. If subsequent backups are to be written to the same device, this procedure is not required.

Backup Type Selection

After a device has been selected, the screen appears as in the following figure:

```
[Backup] '=' for menus, '?' for help                    ["movies" database]
                    Backup report for "movies" database


  Device name                   Device file       Device size  Block size
  Test Disk Backup              /tmp/movies               500       10240



  Type                     Selected

  Data files                   ░
  Index files                  #
  Look-up table files          #
```

Backup Type Selection

Across the top of the screen is the selected backup device. This includes the Device name is simply a text description of the type of the device. The Device file is the Unix file name that is used to access the device. The Device size is the number of Block size units that the device can store. The block size is specified in bytes, and is the blocking factor to be used for the device.

The user may then select the type of backup to be performed. The backup types are as follows:

Data files

These consist of all files critical to the database. These include the database Insertion and Query forms and the data file, containing all records stored in the database. Every database backup should include the Data files backup type. A database may be totally rebuilt from the Data files using the automatic rebuild facilities.

Index files

Index files contain the information used by the indexing method employed by KE Texpress. They can be rebuilt from the data, if required.

Form files

Form files encompass all Report forms associated with the database. If the database does not have any Report forms, this type is not displayed.

Look-up table files

These contain the Look-up tables and hierarchies of the database. If the database does not have any Look-up tables, this type is not displayed.

Interface files

If the database has explicit text, chars or noise word files installed they are encompased by this backup type.

Other files

If the database directory contains a sub-directory called **other** then all files and directories in that directory are encompassed by this backup type.

A backup type is selected if the character, #, is displayed directly to the right of the type name. Initially all backup types are selected, indicating that a complete database backup is to be performed. In the majority of cases this is sufficient and no further selection need be performed. However, if required, selective levels of backup may be specified. These are indicated by simply toggling the various selection marks for each backup type. The commands available are:

Set/Reset ([Backup] Function ⇒ Toggle)

Toggle the current type from set to reset or vice versa.

Expand ([Backup] Function ⇒ Expand)

List the individual files so that individual files may be backed up rather than all files encompassed by the particular type (described below).

Select device ([Backup] Function ⇒ Select device)

Select a new backup device.

Volumes ([Backup] Function ⇒ Volumes)

Display the number of volumes required for the current backup type and file selections.

Perform ([Backup] Function $\Rightarrow$ Perform)

Choose this command when the backup type and file selection is complete. It commences the actual backup procedure. See performing the Backup section in this chapter

Exit ([Backup] Function $\Rightarrow$ Exit)

Exit from the backup procedure.

Forward ([Backup] Move $\Rightarrow$ Forward)

Move forward to the next type.

Backward ([Backup] Move $\Rightarrow$ Backward)

Move backward to the previous type.

Individual Files Selection

If the expand command is selected during backup type selection, the screen displays the individual file names associated with the backup type. See the following figure for the data files:

```
[Files] '=' for menus, '?' for help                    ["movies" database]
                    Backup report for "movies" database


Device name                      Device file         Device size  Block size
Test Disk Backup                 /tmp/movies                 500       10240



Type                    Selected      File                        Selected

Data files                 #          Insertion form                 █
Index files                #          Query form                     #
Look-up table files        #          Data                           #
                                      Data free list                 #
                                      Temporary                      #
                                      Configuration                  #
                                      Statistics                     #
                                      User registration              #
                                      Startup Message                #
                                      Debug                          #
```

Individual File Selection

The commands for individual file selection are similar to backup type selection and are as follows:

Set/Reset ([Files] Function ⇒ Toggle)

Toggle the current file from selected to reset or vice versa.

List Volumes ([Files] Function ⇒ Volumes)

Display the number of volumes required for the current backup type and file selections.Toggle the current file from selected to reset or vice versa.

Exit ([Files] Function ⇒ Exit)

Exit from individual file selection and return to type selection. If at least one file has been selected, the # symbol appears next to the corresponding type.Toggle the current file from selected to reset or vice versa.

Forward ([Files] Move ⇒ Forward)

Move forward to the next file.Toggle the current file from selected to reset or vice versa.

Backward([Files] Move ⇒ Backward)

Move backward to the previous file.Toggle the current file from selected to reset or vice versa.

Next page ([Files] Move ⇒ Next page)

Move to the next page of files (if all files cannot be displayed in the file selection area simultaneously). If the cursor is at the last page, the terminal bell sounds.Toggle the current file from selected to reset or vice versa.

Previous page ([Files] Move ⇒ Previous page)

Move to the previous page of files. If the cursor is at the first page, the terminal bell sounds.Toggle the current file from selected to reset or vice versa.

Any combination of files may be selected. These selections are retained when control is returned to backup type selection.

Performing the Backup
([Backup] Function ⇒ Perform)

After the backup type and file selection has been completed, choosing this command commences the backup process.

The capacity of the device is determined by multiplying the device size and block size values.

If the device is a regular Unix file, and the data to be backed up exceeds the designated capacity, a message displays, and the user is returned to the Backup Type Selection screen:

```
File size too small for backup
```

If the device is a Unix special file such as those used to access tape, cartridge or floppy units, the data is written in volumes. Naturally if the data to be backed up exceeds the device capacity, several volumes are required. KE Texpress initially prompts for the first volume to be loaded with:

```
Load volume 1 now
```

If the system detects that it cannot correctly access the device, it prompts:

```
Can't write to device. Retry?
```

Choosing no aborts the operation is aborted and control returns to backup type selection. A different device may be selected if desired. Otherwise, if yes is selected a message displays:

```
Check that the device is loaded and on-line
```

This is followed by the initial prompt instructing that volume 1 is to be loaded.

If the volume is loaded correctly, the screen is updated to display the percentages of the database and device which have been written. These percentages provide a visual indication of how much of the backup has been performed, and when the next volume (if needed) is required. The backup date and current volume number are also displayed, as well as an indication of the number of volumes required.

As the backup proceeds, the percentages are dynamically updated. If a volume becomes full, a message displays:

and appropriate prompts are given to indicate that the next volume should be loaded. When the backup is complete, a message displays:

```
Database backup complete
```

and typing any key returns control to Admin.

Each backup volume should be distinctly labeled with the database name, backup type, date and volume number. KE Texpress writes this information with each volume and this is checked when the volume is used in a database restore operation.

Aborting the Backup

If the interrupt key is typed at any stage of the backup procedure, a prompt displays:

```
Abort?
```

If **yes** is selected, the operation is aborted, whereas selecting **no** resumes the operation.

Chapter 4

Restoring Database

Overview

Restoring a database from a backup is required only after exceptional events such as a disc head crash, which renders a disc useless, or the development of a bad block on a disc, which results in loss of information. Restoration can also be used to revert the database to a previous state when recent changes that have been made to a database need to be undone. In this way backups and restores can be used to protect users from their own actions.

As there are many dependencies between the files which make up a database, it is strongly recommended that only complete databases are restored. Partial restoration may have unpredictable results, where a restored file does not reflect the changes made since the backup was performed. For example, restoring the index files without restoring the data file means that records inserted or updated since the backup will not be indexed and so will not be found when querying. Should a partial restore be required, first check with your hot-line support provider before proceeding.

The **Restore database** option is selected from the Maintenance pull down menu of Admin. Only the DBA of the database can perform this command.

The restoration procedure is very similar to the backup procedure described in the previous chapter. The general approach is to select the Unix device used for the restoration, select the appropriate type (or files), then commence and monitor the operation, changing volumes (media) as necessary.

If the database does not exist when the restore facility is required (perhaps a new disc or machine is being used), then the database must first be created using the KE Texpress command, **texcreate**. Refer to the KE Texpress Design Guide for help on this command. After this, the restore facility can be invoked. In this situation the device must first be selected, so that the backup can be accessed. If the backup is to be restored into an existing database, a backup device may already exist. If the device is correct, then the device selection phase can be bypassed.

Device Selection

Before the restore type and file selection procedure can commence, KE Texpress must determine which files were originally backed up. To do this, the backup device must be selected (if not already) and the first volume loaded. If no backup device exists, then KE Texpress automatically enters backup device selection, which is as described in the previous chapter. If the selected device is a Unix special file, KE Texpress prompts with:

```
Load volume 1 now
```

and requests any key to pressed when ready. If the device cannot be accessed, the user is given another chance to load the volume correctly. If the interrupt key is pressed during this sequence, then the user is given the option of terminating the restore procedure or returning to device selection.

If the device can be accessed, KE Texpress performs several checks before the volume is accepted. The first volume must be loaded, as this contains a table of contents for the backup. If an incorrect volume is loaded, a message is displayed of the form:

```
Incorrect volume (2), volume 1 is required
```

The restore does not continue until the correct volume is loaded. Assuming the correct volume number is found, the name of the current database and that of the database from which the backup was taken are compared. If they are not equivalent, the prompt:

```
Warning! Restore "d1" database into "d2" database.
Continue? [y/n]
```

is displayed (where **d1** is the name of the database stored on the backup and **d2** is the name of the current database). This facility provides a means of restoring a database from the backup of a different database. If yes is selected, KE Texpress then determines whether any data exists in the current database. If so, the following prompt is displayed:

```
You will overwrite the current database! Abort? [y/n]
```

Selecting **no** commences the restore. Because the restore process overwrites any existing data and forms, restoring into a database with existing data should be performed with caution.

Restore Type and File Selection

KE Texpress determines which files are saved on the backup by reading the table of contents stored on the first volume. The selection of the type and individual files is then commenced. This selection procedure is as described in the previous chapter. Naturally only those files actually saved on the backup volumes can be selected for restoration.

If the database backup contains the Insertion form, Data file and Data file free list then one extra type option is available for restoration.

Compact

This type indicates that the database Data file should be compacted after it is read from the backup. The compaction removes any fragmentation in the Data file, which can occur after many record edit operations. When the Compact type is selected, the Index files type is automatically deselected. Temporary file space on the disc, equivalent to the size of the Data file, is required during the compaction. After a restoration with the Compact type selected, an index rebuild is required. The database will not be available to users until the index rebuild is performed.

Performing the Restore

When selection is complete, the Perform command (**[Restore] Function ⇒ Perform)**, is used to commence the restoration of the database files selected. KE Texpress prompts for new volumes as they are required, performing consistency checks on each new volume as it is loaded. These checks cover the database name, the date of the backup and the volume number.

As with database backup, the database and device percentages are displayed on the screen, and are dynamically updated as the database is restored. The following message is displayed when the restore has finished:

```
Database restore complete
```

Chapter 5

Look-up Table Editing

Overview

Look-up tables aid database record insertion and querying, by providing a fast method of viewing the vocabulary of an item. This vocabulary is generally created from inserted data, however the Look-up table edit facility may be used to preload a vocabulary for the item or to edit its existing vocabulary. Administrators require **Look-up table edit** privilege in order to use this facility.

It is important to remember that editing a Look-up table is simply editing an item's vocabulary and does not alter the data in existing records. Note that values inserted into the Look-up table during data insertion remain in the Look-up table, even if the record which contributed that value is deleted from the database. A value which does not occur in existing records may be added to the vocabulary in the Look-up table by editing it directly. Changes must be made to the appropriate records to ensure consistency between Look-up tables and record data. If consistency is required, the DBA can automatically rebuild a Look-up table from the record data.

Any major Look-up table editing session should be preceded (and probably followed) by a backup of the database. Refer to the chapter on Database Backups for further information.

Select Look-up Table

Look-up table editing is selected from the Maintenance pull down menu in Admin mode.

When the Look-up table edit facility is invoked, the database Insertion form is displayed. The administrator can traverse the Insertion form and select the appropriate item for editing. It is possible to traverse only to items which have the Look-up table attribute set and which are not child items in a hierarchy. To edit child hierarchy items, the parent item must first be selected and the appropriate child values expanded from the parent levels.

Edit item ([Look-up table select] Function ⇒ Select)

Edit the Look-up table (or hierarchy) associated with the item. (See the Edit Look-up table section in this chapter for more details.)

Balance Look-up table ([Look-up table select] Function ⇒ Balance)

As the sequence of insertions into Look-up tables can be somewhat random, Look-up table access performance can deteriorate over a period of time. Balancing a Look-up table restores it to its optimum configuration and improves access speed. If the selected item is the parent of a hierarchy, then it is possible to balance each level of the hierarchy in turn.

When this command is selected, KE Texpress requests confirmation by displaying the following message:

```
Balance "name" Look-up table?
```

Values for the average search depth and worst case search depth are displayed before and after the balancing procedure. The average search depth gives an indication of the average number of values which must be read before a particular value is found. The worst case search depth indicates the maximum number of values which need to be read in the search for a given value.

Note that if the Look-up table edit program is used to preload Look-up tables it is important that each table is balanced once loaded. Look-up tables should also be balanced after any major Look-up table edit session, or after large numbers of database record insertions.

Modify Status ([Look-up table select] Function $\Rightarrow$ Access)

> KE Texpress permits database Look-up tables to be designated read only. This can be used to ensure that preloaded Look-up tables maintain their integrity by disallowing new table values at record insertion time.

> When this command is selected KE Texpress displays a message indicating the current status of the Look-up table (Read or Read/Write) and requests confirmation that the status is be reversed.

> Editing of read only Look-up tables is not permitted. The table entries may be viewed using the movement commands, however all editing commands are disabled. To be edited, such a table must first have its access state modified to read/write.

Copy Look-up table ([Look-up table select] Function $\Rightarrow$ Copy)

> Copy the contents of the Look-up table (or hierarchy) to a Unix file or send the information to the printer. A copy of all Look-up table values (with traversal to lower levels of the hierarchy as appropriate) is taken.

Exit ([Look-up table select] Function $\Rightarrow$ Exit)

> Exit the Look-up table edit program and return to Admin.

Navigation

Next item ([Look-up table select] Move $\Rightarrow$ Forward)

> Move forward to the next editable item.

Previous item ([Look-up table select] Move $\Rightarrow$ Backward)

> Move backward to the previous editable item.

First item ([Look-up table select] Move $\Rightarrow$ First Look-up table)

> Move to the first editable item on the form.

Last item ([Look-up table select] Move $\Rightarrow$ Last Look-up table)

> Move to the last editable item on the form.

Find item ([Look-up table select] Move $\Rightarrow$ Find prompt)

> Find a prompt by leading letters.

Edit a Look-up Table

When a Look-up table item is selected for editing, it is highlighted, and the following message is displayed:

```
Reading "name" Look-up table . . .
```

Editing is performed on a temporary copy of the Look-up table file so if the table is very large the reading phase may take a few minutes. During this time a message will be displayed to indicate that the values of the look-up table are being extracted

and the first page of values associated with the table is displayed, one value per line.

Values may be visited and edited as desired. The actual table values are edited using the regular edit mode commands.

Insert item ([Look-up table edit] Function ⇒ Insert value)

Add a new value to the Look-up table. Values are entered and edited using the Text edit mode commands. The entered value should be terminated with (Return). If the Look-up table has User specified order, a new line is allocated directly above the current value, and the new value can be entered. Otherwise the new value is entered in a message box and then automatically positioned. If the entered value already exists in the Look-up table, the matching value is highlighted on the screen and the following message is displayed

```
Value "value" exists
```

is displayed. The value must be entered again. Entering a null (or blank) value aborts the command.

Insert item to user-specified Look-up table
([Look-up table edit] Function ⇒ Append value)

This command is available only for User specified Look-up tables. A new line is allocated directly after the current value, and the new value can be entered.

Edit item ([Look-up table edit] Function ⇒ Edit value)

Edit the text of the Look-up table value at which the cursor is currently placed. The resultant value must be unique among all Look-up table values. If the Look-up table order is not User specified the edited value is re-inserted in the appropriate place in the table. The editing window shifts (if necessary) to show the value in its new location. For User specified Look-up tables, the edited value stays in the same location. It may be moved to a new location using the position command.

Delete item ([Look-up table edit] Function ⇒ Delete value)

Delete the current value from the Look-up table. Deletions can't be undone, therefore this command should be used with caution. The prompt the following prompt will be displayed requesting confirmation that the value is to be deleted:

```
Delete "value"?
```

If **yes** is selected the value to be deleted has child values below it in the hierarchy, the additional prompt will be displayed:

```
"value" has children. Abort?
```

A **yes** response aborts the delete command. Otherwise typing **no** deletes the value and all its children. Selecting **no** to this prompt deletes the value and all its children.

Delete all items ([Look-up table edit] Function ⇒ Delete all)

This command is similar to the delete command, except that all values (and their children if applicable) at the current Look-up table level are deleted. The following prompt is displayed:

```
Delete all values (and their children) at this  level?
```

Selecting **no** aborts the command, otherwise, if **yes** is selected, all table entries at this level are deleted.

Save ([Look-up table edit] Function ⇒ Save)

As Look-up table editing is performed on a temporary copy of the original table, the write command is used to save all changes made. All modifications since editing of the table (or hierarchy) commenced, or since the last write command, are copied back to the original Look-up table (or hierarchy) file.   Before writing KE Texpress displays the following message:

```
Write changes to "name" Look-up table? [yes] [no]
```

Selecting **yes** commences the write operation. If the table is large this can take a few minutes.

Exit ([Look-up table edit] Function ⇒ Exit)

Contract to the immediate parent level of a hierarchy. If this command is selected while editing a single level Look-up table or the top-most level of a hierarchy is being edited, the status is checked. If the table has been modified since the last save command, the following message is displayed:

```
"name" Look-up table modified. Really  exit? [yes] [no]
```

Selecting **yes** result in an exit from the editing phase, with all Look-up table modifications discarded. Selecting **no** causes the exit command is aborted. If the table has not been modified, control returns immediately to the item selection procedure.

If the exit command is entered whilst editing lower levels of a hierarchy, the immediate parent level values are extracted, and editing commences at that level.

While the contraction to the parent level is performed the following message is diplayed.

```
Contracting "value" . . .
```

Navigation

Cursor movement commands may be preceded by a repetition count so that the command is performed repeatedly.

Next item ([Look-up table edit] Move ⇒ Forward a value)

Move the cursor forward (down) one value. If necessary the list of values will scroll.

Previous item ([Look-up table edit] Move ⇒ Backward a value)

Move the cursor backward (up) one value. If necessary the list of values will scroll.

Next page ([Look-up table edit] Move ⇒ Next page)

Display the next page of values.

Previous page ([Look-up table edit] Move ⇒ Previous page)

Display the previous page of values.

First item ([Look-up table edit] Move ⇒ First value)

Move the cursor to the first value in the Look-up table.

Last item ([Look-up table edit] Move ⇒ Last value)

Move the cursor to the last value in the Look-up table.

Find item ([Look-up table edit] Move ⇒ Find value)

Find a Look-up table value by leading letters. The search commences directly after the value upon which the cursor is currently residing and loops around the list of values. For very large Numeric or User specified Look-up tables this command can be slow.

View status ([Look-up table edit] Move ⇒ View status)

Status information regarding the current level of the hierarchy is obtained using this command. The status information details the Look-up table type, the number of values and the current depth level in the hierarchy (where level one is the top level). Also displayed are the parent values of that level (if appropriate).

Expand Look-up table

Expand value (Expand ⇒ Select Value)

Select the current value for expansion. All immediate child values of the current value are extracted, and editing may commence at that level. The following message is displayed while the child values are extracted:

```
Extracting "value" . . .
```

Copy values (Expand ⇒ Copy)

A copy of all Look-up table values at the current level can be written to a Unix file, or sent to the printer. If editing is currently being performed on an upper level of a hierarchy, the following prompt is displayed:

```
Include all children? [yes] [no]
```

Selecting **yes** causes all values at the current level, along with their child values, are copied. If **no** is selected, only those values at the current level are copied.

Move value (Expand ⇒ Move value)

If a Look-up table has User specified ordering, this command can be used to position the current value at a new location in the order. The mode indicator displays:

```
[Position "value"]
```

A new position can be chosen for the value, using the movement commands. The value is highlighted during the operation. The Insert or Append commands are used to insert before the current value or to append after the current value respectively. To exit from the position command without changing the order, the Exit command is used.

Merge values (Expand ⇒ Merge value)

This command merges two Look-up table values (and all of their children). Consider the case of a hierarchy of two levels, namely country and city. Suppose that both the values *Australia* and *Astralia* appear in the country Look-up table. Using this command, the incorrect value (and all its children) can be combined with the correct value. After the cursor is positioned over the incorrect value (*Astralia*) the merge command is selected. The mode indicator would display the following:

```
[Merge "Astralia"]
```

The cursor should be placed at the correct version of the value (*Australia*). The select command is entered to choose that value. The following prompt would be displayed:

```
Merge "Astralia" into "Australia"? [yes] [no]
```

Selecting **yes** causes, the values (and all children) are merged.

Chapter 6

Database Recovery and Maintenance

Overview

This chapter describes a utility used for recovering a database after hardware failure. Also described is a general maintenance utility which should be executed periodically to improve disc usage for the database.

These utilities require the database to be closed before commencement. When closing the database, if the system detects people still using the database, they will be warned of the impending closure and their sessions will be terminated. It is not necessary to close the database explicitly if there are no current database users. If the database is not closed when one of these operations is selected, it is silently closed. When the operation is completed, the database is silently re-opened.

The maintenance facilities described in this chapter can be executed by users with **Administrator menu** privilege.

Recovery after System Crash

Computer systems will fail occasionally. These failures (or system crashes) can be caused by hardware, or operating system bugs, or external causes such as a power failure. As a badly timed system crash can leave a database in an inconsistent state (such as a partially inserted record), KE Texpress provides a utility which can be used for restoring the integrity of a database after such an event.

The crash recovery command, selected from the Maintenance pull down menu of Admin, performs several recovery procedures to restore the database to a stable state. Although it cannot complete the database operation being performed at the time of the crash, it can generally restore the database to its state prior to the commencement of the interrupted operation. A message will display:

```
Commencing recovery program . . .
```

The program checks various database files to ensure their validity. Lock files which are used to provide database mutual exclusion are removed. Also the database statistics are repaired, so that the total number of records in the database is correct.

Database Maintenance

The database maintenance program is selected from the Maintenance pull down menu in Admin mode. It is used to remove any redundant information associated with the database, thus freeing up the associated disc space. It also re-organizes the data file free list which is a list of the gaps or holes in the data file left by record edit operations.

This program also compacts the temporary file which, if used heavily for the storage of records before insertion, may hold a large amount of redundant information. The temporary file is simply scanned and information compacted where possible. A percentage value is displayed indicating the compaction achieved. Temporary file compaction occurs only for databases with a Key item.

It is not necessary to perform the database maintenance procedure on uninitialized databases. However, for initialized databases it is important that it is executed periodically. A warning is automatically issued when KE Texpress considers that it should be performed. This is described in Chapter 1 of this manual.

When the maintenance option is selected, the message:

```
Commencing maintenance program . . .
```

is displayed. Typical output from the maintenance program is as displayed in the following.

```
Performing maintenance
        Removing redundant files
        Compacting temporary file (0% compaction)
        Restructuring data file free list
Done.




                        ┌─────────────────────────┐
                        │    Maintenance completed │
                        └─[ Hit any key to continue ]─┘
```

Database Maintenance

Chapter 7

Batch Insertion Mechanism

Overview

Many indexing methods are designed to provide fast retrieval of information. Unfortunately, this speed is often gained at the expense of the interactive record insertion and update speeds as these operations must perform much additional work to ensure that retrieval is efficient. KE Texpress has been specifically designed for extremely fast retrieval from massive data collections. However the interactive insertion time can deteriorate as databases become large.

When KE Texpress records are inserted or updated, the default insertion process automatically completes insertion of both the records' data, and all indexing information required by the KE Texpress retrieval technique.

An alternative fast insertion mechanism is available to improve the time taken during an interactive insertion and update in KE Texpress. A significant amount of the work to insert indexing information is postponed, to be performed during a batch insertion at a later time, such as at night or during the weekend.

The records inserted by this fast insertion method are said to be partially inserted. However, to the user the fast insertion mechanism is completely invisible. The partially inserted records are available immediately for retrieval or further editing. To achive this, the necessary information about the partially inserted records is maintained in a file, known as the overflow file, in the database directory.

A batch insertion must be performed to complete the indexing work of the insertion process for the records with entries in the overflow file, and so remove the entries for these records from that file. A regular schedule for batch insertions may be establised by the DBA. The overflow file may also be flushed at any time on demand. By default, exclusive access to the database is required during a batch insertion so no users can be active during this time. However, if the database option slowbatch is set, then batch insertions can proceed whilst users are accessing the database (see the Design Guide for more details).

The only penalty associated with the batch insertion mechanism is that if many partially inserted records exist, the query response time may be marginally compromised. The overflow file must be searched during every query, and if it contains many records, then the search time increases.

The Admin Batch Menu

The batch insertion mechanism can be enabled and scheduled from the Batch menu of Admin. The Batch menu appears as in the following. By default, the batch insertion mechanism is disabled on a newly created database.

```
[Admin]   Function   Maintenance   Audit   Batch
                                            Enable
                                           <Disable      >
                                            Flush...
                                            Batch count
                                            Schedule...
```
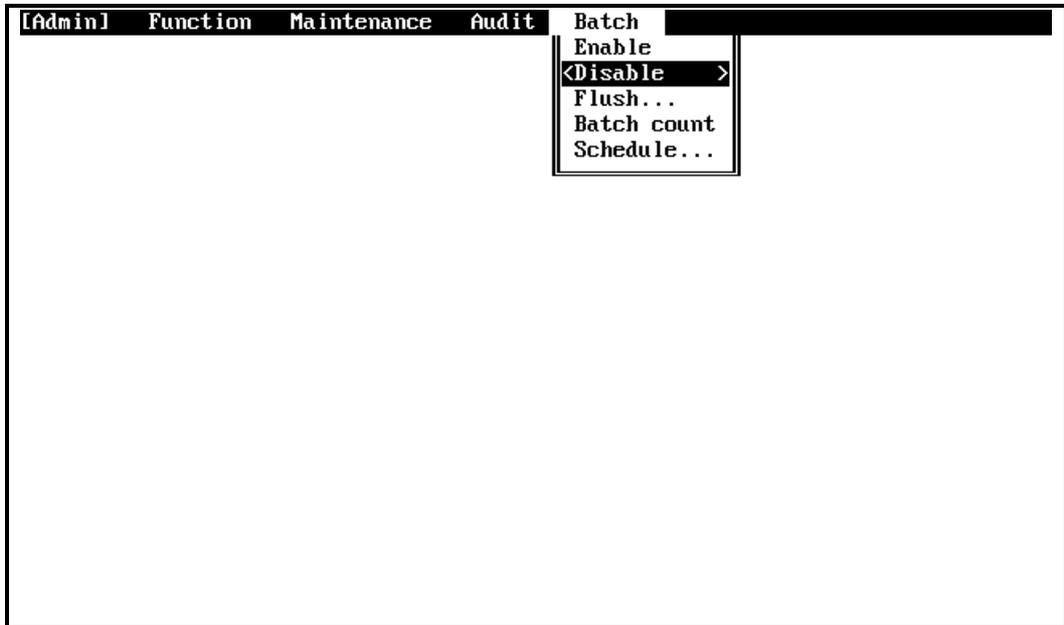
The Admin Batch Pull-down menu

Enable Batch Insertions
([Admin] Batch ⇒ Enable)

The batch insertion mechanism can be enabled only by the DBA. Enabling batch insertions mechanism can be done at any time after the database has been initialized. Texpress then displays the message:

```
Batch insertion method enabled
```

An overflow file is then created as an initially empty file in the database directory. In subsequent User sessions, all new records are partially inserted. Records updated using the edit or global edit operations are also partially inserted, except where changes are so simple that total re-insertion is equally efficient. The partially inserted records are said to be inserted in the overflow file. This means that information necssary to their retrieval is stored in the overflow file.

The configuration of the database limits the capacity of the indexing files. This limits the number of totally inserted records the database can hold. Partial insertion avoids entering indexing information in the indexing files, and the overflow file has no maximium capacity. Consequently, enabling the batch insertion mechanism has the side effect of removing the capacity limitation of the database.

It is possible to take advantage of this side effect. If a user attempts to insert a record into full database in which batch insertions are disabled, the user sees the message:

```
"dbname" Database is full. Re-entering Edit Mode
```

The user is then returned to Edit mode (At this stage the record can be saved in the temporary file if the database has a Key item). If the DBA enables the batch insertion mechanism while the user remains in Edit mode, a subsequent attempt to save(write) the record will be successful, as the record will then be partially inserted, with an entry in the overflow file.

When data is loaded into a database using the **texload** program, while the batch insertion mechanism is enabled, the records are partially inserted. Consquently more records can be loaded than dictated by the database capacity. However, when the batch insertion subsequently proceeds, it will be unable to continue once the capacity of the database is reached.

If the capacity of the database is exceeded through insertions or data loads while the batch insertion mechanism is enabled, it is then necessary to reconfigure the database and rebuild it, so that records in excess of the current capacity can be inserted completely.

Disable Batch Insertions
([Admin] Batch ⇒ Disable)

The batch insertion mechanism can be disabled only by the DBA. Disabling batch insertions can be done whenever the overflow file is empty. The following message is then displayed:

```
Batch insertion method disabled
```

If, however, the overflow file, contains entries for partially inserted records, the message displayed is:

```
Overflow file is not empty. Flush it first
```

and the command is aborted. The records must be flushed by a batch insertion. (See Flush Overflow File in this chapter).

Flush Overflow File
([Admin] Batch ⇒ Flush)

The overflow file can be flushed at any time by the DBA, provided that there are no active database users. This has the effect of completing the insertion of all partially inserted records, and thus improving the query response time. If a batch insertion can be performed, the following message is displayed:

```
Batch insertion performing in background
```

The database can be used when the batch insertion has been completed.

Each batch insertion produces a report. This report is automatically displayed to the DBA on the next entry to the database via **texadmin**, and is then removed. The format of the report is described in a following section.

Overflow Record Count
([Admin] Batch ⇒ Batch count)

The number of records in the overflow file can be determined at any time, while the batch insertion mechanism is enabled. The batch count is shown in the displayed message:

```
Overflow file contains n records
```

This option can be performed by any users with access to **Administrator mode**, but only when the batch insertion mechanism is enabled.

Schedule Batch Insertions
([Admin] Batch ⇒ Schedule)

Since periodic batch insertions are required to flush the overflow file, KE Texpress provides a facility to schedule batch insertions at regular intervals of up to one week. These can then be performed outside normal working hours, without the need for further intervention by the DBA.

If a batch insertion is attempted at a scheduled time, but cannot be completed for some reason (for example, there are users active), the batch insertion is aborted, and an appropriate error message is generated for the DBA. This does not generally present a problem, as the next scheduled batch insertion will proceed, and complete the insertion of all partially inserted records. Should the problem persist, however, it should be rectified by the DBA.

The batch insertion schedule can be viewed by all users with **Administrator menu** privilege. It can be altered only by the DBA. The screen display is like that shown in the following figure:

```
[Select times] '=' for menus, '?' for help              ["contacts" database]
                 Batch insertion schedule for "contacts" database


Current system date and time: Mon Jul 11 18:36:40 1994

   Day                         Hour                      Minute

Sunday      #        0 A.M.      #      12 Noon          0 min
Monday      #        1 A.M.              1 P.M.          5 min
Tuesday     #        2 A.M.              2 P.M.         10 min
Wednesday   #        3 A.M.              3 P.M.         15 min
Thursday    #        4 A.M.              4 P.M.         20 min
Friday      #        5 A.M.              5 P.M.         25 min
Saturday    #        6 A.M.              6 P.M.         30 min        #
                     7 A.M.              7 P.M.         35 min
                     8 A.M.              8 P.M.         40 min
                     9 A.M.              9 P.M.         45 min
                    10 A.M.             10 P.M.         50 min
                    11 A.M.             11 P.M.         55 min
```

Batch insertion schedule

The commands available to edit the schedule are listed below:

Toggle ([Select times] Function ⇒ Toggle)

Toggle the current time slot. A # next to a day, hour, or minute value indicates that the value is selected. More than one value can be selected from the same time period. For example, if two different day values are selected then a batch insertion will be performed at the selected time on both of these days. Beware, however, that selecting two different minute value

indicates that a batch insertion should occur twice in each hour selected by the rest of the schedule.

Save ([Select times] Function ⇒ Save)

Write the current schedule. The schedule is saved and becomes activated immediately. If no times are selected, then no automatic batch insertions are performed. However, the batch insertion mechanism is still enabled and the DBA can flush the overflow file as required.

If at least one time slot is selected, but there is not a value selected for every time period, then the schedule defaults to **Sunday** if the day is not selected, **0 A.M.** if the hour is not selected and **0 min.** if the minute is not selected.

Exit ([Select times] Function ⇒ Exit)

Exit from the schedule without saving it and return to the Batch insertion menu.

Forward ([Select times] Function ⇒ Forward)

Move forward to the next time slot on the screen.

Backward ([Select times] Function ⇒ Backward)

Move backward to the previous time slot.

Next ([Select times] Function ⇒ Next)

Move to the top of the column of the next sized time period. Thus the cursor moves from the days column to the hours or from the hours to the minutes, etc.

Previous ([Select times] Function ⇒ Previous)

Move to the top of the column of the previous sized time period.

When the batch insertion mechanism is first enabled, no batch insertions are scheduled. If this facility is to be used and many insertions and updates are performed on the database, the DBA should schedule a regular batch insertion. If possible, this batch insertion should not clash with the batch insertions of other databases, as the operation can be taxing on a machine's resources.

A report is produced by each batch insertion initiated by the schedule. These reports are displayed to the DBA the next time the DBA enters the database via **texadmin**. After they have been viewed they are discarded.

Initiate Scheduled Batch Insertions

Provided that the installation of the KE Texpress software was performed correctly, the program, **texupdsched**, should be running at all times. (This can be determined by using the Unix command, **ps**.) This program constantly monitors the batch insertion schedules of all databases and initiates batch insertions as necessary.

Batch Insertion Reports

Each batch insertion, either automatically or interactively initiated, produces a report containing its starting and finishing times, the elapsed time and the number of records transferred from the overflow file. The next time that Admin is invoked, this report is displayed to the DBA using the Text perusal facility. It can be copied if desired. It is automatically removed after it has been viewed.

Any errors encountered by the batch insertion are also reported in this way. A warning is produced if the capacity of the database is reached, preventing remaining records from being transferred from the overflow file. If this situation should arise, the database should be re-configured with a larger capacity so that the overflow file can be flushed completely (thus restoring optimal retrieval speed).

Overflow File Warning

If the overflow file contains more than 500 partially inserted records, whenever the DBA invokes **texadmin**, the following warning is displayed:

```
A batch insertion should be performed
```

This indicates that a batch insertion should be invoked in the near future as the overflow file is considered to be large enough to have a degrading effect on query response time.

Chapter 8

Database Repair Utilities

Overview

This chapter describes several KE Texpress utilities which can be used to repair databases when exceptional circumstances have caused inconsistencies. They are rarely, if ever, required and should be used only if the DBA has a clear understanding of their effect.

All of the utilities described in this chapter must be executed from the Unix shell or command interpreter. They can be executed only by the DBA of the database.

Next Key Value Assignment

KE Texpress has provision for automatic assignment of a Key value to a record during data entry (refer to the KE Texpress User Guide). This assignment occurs as an insertion commences, and can occur only in databases with a Key item with an incrementing numeric component. These Key items include the pre-defined KE Texpress Key types and any other user-defined Key types with an incrementing field (refer to the KE Texpress Utilities Guide). Such Key types are said to consist of a **suffix**, which is the first field to be incremented, and a **prefix**, which is all remaining fields.

To achieve automatic assignment, KE Texpress maintains a list of all distinct prefixes used by the database, together with the next suffix to be assigned with that prefix. In this way, when a prefix is identified, the next available Key value can be determined easily.

Sometimes it is necessary to reserve a block of Key values without assigning them to records. A value beyond these reserved values is to be regarded as the next Key value to be used. This can be achieved by using the KE Texpress program, **texnextkey**, which can alter the suffix for any prefixes. This program must be executed from the Unix shell by using a command of the form:

```
texnextkey dbname  [field_1 field_2 . . . field_n]
```

The first argument to this command is the name of the database. Following this are the values of each field of the Key item in its correct order. This includes the numeric suffix. The suffix is removed to determine the prefix, and the next Key value to be assigned for that prefix is then stored. The next time that the prefix is used within KE Texpress while next available Key value assignment is active, the suffix value will be assigned. Subsequent Key values will be assigned by incrementing this suffix.

If this command is invoked with appropriate parameters, it produces output of the form:

```
Old Key value:  M       823
New Key value:  M       1001
```

If no field values are specified on the command line, **texnextkey** displays a table of the prefixes used by the database together with their respective suffixes. In this case the command produces output of the form:

```
Key     1:      A       207
Key     2:      M       1001
Key     3:      PR      7
```

This command is available only for databases with a Key item.

Clearing Key Value States

As described in the KE Texpress User Guide, database Key values are always in one of four states. These states are as follows:

Unassigned

The Key value is not associated with any data.

Temporarily assigned

The Key value is reserved for a user but, so far, has no data associated with it (i.e. the user is currently entering the data).

Temporarily stored

The Key value has data associated with it, stored in the Temporary File.

Permanently assigned

The Key value is associated with a record stored in the database.

These states are mutually exclusive and are controlled entirely by KE Texpress. Thus under normal operation, they are always correct. However, again extraordinary circumstances (perhaps a system crash during a Key value state change) can result in a particular Key value assuming an inappropriate and invalid state.

The states of all Key values can be corrected by rebuilding the descriptors for the database (refer to the KE Texpress Administration Guide). But this can be time-consuming if only one Key value is in error. For this reason, KE Texpress provides a facility to repair the states of Key values selectively, by assigning to them the state, Unassigned. The program to achieve this is called **texclrkey** and can be invoked from the Unix shell by typing a command of the form:

```
texclrkey dbname field_1 field_2 . . . field_n
```

where ***dbname*** is the name of the database. The remaining arguments represent the fields of a Key value which is to be repaired. This command is available only for databases with a Key item, and can be performed only if there are no users active in the database.

An example of the output produced by this command is given in the following figure:

```
              Clearing Key values for "contacts" database
              -------------------------------------------

     Key value : 4

     State     : Permanently assigned

     Offset    : 14┌────────────────────────────────┐
                   │ Change state to unassigned?     │
                   │                                 │
                   │      ┌─────┐      ┏━━━━━┓        │
                   │      │ Yes │      ┃<No> ┃        │
                   │      └─────┘      ┗━━━━━┛        │
                   └────────────────────────────────┘
```
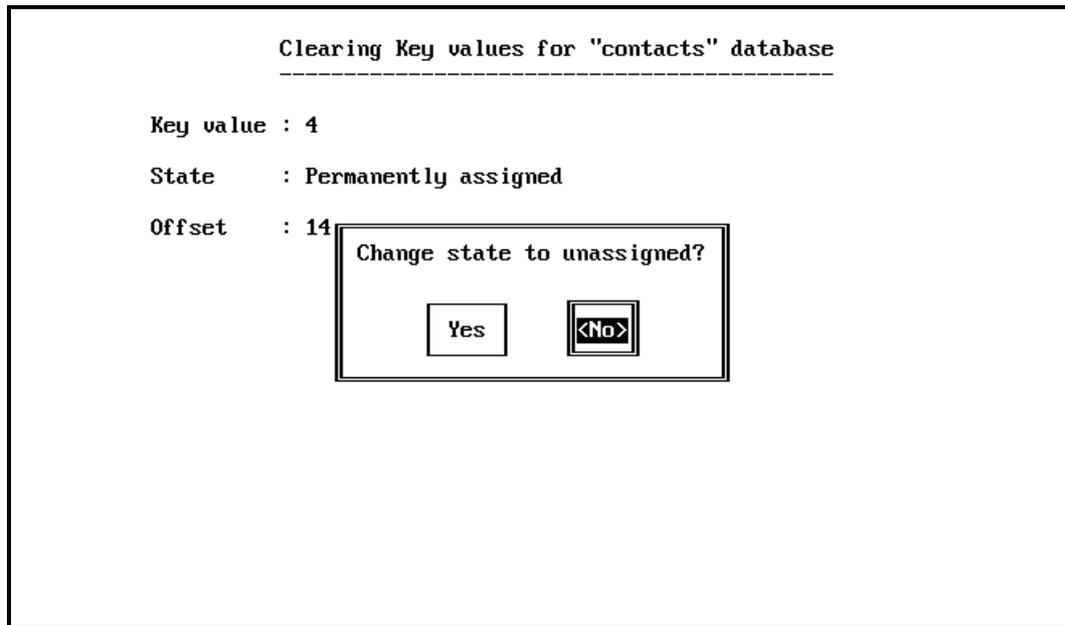
Clearing Key value states

The Key state displayed by this command is one of the states described above. For the temporarily stored and permanently assigned states, the data file offset of the record is also given.

If the Key value is unassigned, then a message displays:

```
Key value is already unassigned
```

If the Key value is any of the other three states, then the system will prompt:

```
Change state to unassigned?
```

Choosing yes results in the Key state becoming unassigned. Choosing no results in the Key state remaining unchanged. A message is displayed indicating which of these alternatives was selected and the command then terminates.

Obviously, this program is a dangerous tool. If used carelessly, it can result in the state of certain Key values being incorrect with respect to the state of the database. This, in turn, could result in a Key value being assigned to two different records. Therefore it is important that this program be used carefully and only when absolutely necessary. One condition which would warrant its use is if a Key value cannot be assigned to a record during insertion yet, when querying, no record can be found to have that value assigned to it. Again note that this condition can arise only as a result of a badly timed system crash.

If the states of some Key values appear to be incorrect, yet the DBA is uncertain of how to repair them using this program, then the descriptor files should be rebuilt (refer to the KE Texpress Design Guide). This operation repairs the states of all Key values in the database.

Clearing Record Locks

To ensure that no two users can attempt to modify the same record at the same time, KE Texpress uses the notion of a **record lock**. For each record in the database, there is one such lock which is used to indicate whether a user is currently modifying the corresponding record using the edit or global edit facilities (refer to the KE Texpress User Guide). Hence, if this lock is set, no other user can begin editing the record until it is cleared.

During normal operation of KE Texpress, this lock, when set, is always cleared at the completion of the edit or global edit commands. However, extraordinary circumstances (typically, a system crash) can result in a record lock remaining set even though there is no user currently editing the record.

For this reason, KE Texpress provides a facility to clear the record locks of all records in the database. The program to achieve this is called **texclred** and is invoked from the Unix shell by typing a command of the form:

```
texclred dbname
```

This command can be performed only if there are no active database users. It scans all records in the database clearing any locks set. It then produces a report of the form:

```
Number of record locks cleared = n
```

Every record in the database can then be edited.

Checking Database File Permissions

Although KE Texpress completely controls user access to a database, it relies on the Unix file permissions being set correctly. Under normal operation, these permissions should never become incorrect. However, other Unix utilities (in particular, Unix file restoration facilities) can cause them to be altered.

Should they become incorrect for some reason, they can be reset correctly by using the KE Texpress utility, **texperms**. This program can be invoked from the Unix shell by the DBA with a command of the form:

```
texperms dbname
```

which produces output similar to the following for each file changed:

```
Changing "/u/KE Texpress/data/dbname/data" from mode
0444 to 0660
```

This command need be executed only if KE Texpress produces error messages concerning file permissions.

Index

A

Abort
    Backup, 3-11
Access
    Look-up table, 5-4
Assignment
    Key value, 8-3
Audit file flush, 1-5

B

Backup
    Abort, 3-11
    Data files, 3-4
    Database, 3-2, 3-7
    Device Selection, 3-3
    Device selection, 3-5
    Form files, 3-5
    Index files, 3-4
    Individual file selection, 3-7
    Interface files, 3-5
    Look-up table files, 3-5
    Number of volumes, 3-5, 3-7
    Other files, 3-5
    Perform, 3-9
    Type Selection, 3-4
    Type selection, 3-4
Backup database, 3-10
Backward, 7-9
Balance Look-up table, 5-3
Batch insertion, 1-5, 7-2
    Disable, 7-5
    Enable, 7-4
    Initiation, 7-10
    Reports, 7-11

C

Clearing Key value states, 8-4
Close Database, 2-3
Close, Open and Activity, 2-2
Compact

Data Files, 4-4
    Temporary file, 6-4
Copy
    Look-up table, 5-4
Count Overflow Records, 7-7
Current database activity, 2-6

D

Data file compaction, 1-6
Database
    Activity, 2-6
    Backup, 1-5, 3-2, 3-7, 3-10
    Close, 2-3
    Maintenance, 1-5, 2-3, 2-9, 6-4
    Restore, 4-2, 4-4
    Start message, 2-9
    Statistics, 2-7
    User Activity Report, 2-6
    Utilities, 8-2
Database maintenance, 1-2
Database Recovery and Maintenance:, 6-2
Delete all items
    Look-up table, 5-6
Delete item
    Look-up table, 5-6
Device Selection, 4-3
Device selection, 3-3
    Backup, 3-5
Disable
    Batch Insertion, 7-5

E

Edit Look-up tables, 5-2
Edit start message, 2-9
Enable
    Batch Insertion, 7-4
Exit, 7-9
    Look-up table, 5-4, 5-6
Expand Look-up table, 5-9

R

S

T

U